

---

# SISTEMA HÍBRIDO NEURO-EVOLUTIVO APLICADO AO CONTROLE DE UM PROCESSO MULTIVARIÁVEL

**Leandro dos Santos Coelho\***  
leandro.coelho@pucpr.br

**Viviana Cocco Mariani†**  
viviana.mariani@pucpr.br

\*LAS/PPGEPS - Laboratório de Automação e Sistemas, Programa de Pós-Graduação em Engenharia de Produção e Sistemas - Grupo Produtônica, Centro de Ciências Exatas e de Tecnologia, Pontifícia Universidade Católica do Paraná - Rua Imaculada Conceição, 1155, CEP 80215-030, Curitiba, PR, Brasil

†Programa de Pós-Graduação em Engenharia Mecânica - Centro de Ciências Exatas e de Tecnologia, Pontifícia Universidade Católica do Paraná - Rua Imaculada Conceição, 1155, CEP 80215-030, Curitiba, PR, Brasil

---

## ABSTRACT

This paper presents a new approach for the *PID* (proportional, integral, and derivative) multivariable controller design based on a neural network and a genetic algorithm. The multivariable *PID* controller design is divided in three steps. In the first step, a radial basis neural network is utilized for multivariable process identification. In the second step, the controller design is realized by an off-line procedure. This procedure is based on tuning of the *PID* controller gains by optimization with genetic algorithms aiming to control the model obtained by radial basis neural network. In the third step, the *PID* controller gains obtained with genetic optimization and neural model are validated in a practical process. Performance of this approach to multivariable control design is presented and discussed. The control algorithm design is validated of experimental way in a multivariable process called ball-and-plate. This process consists of a plate pivoted at its center such that the slope of the plate can be manipulated in two perpendicular directions. The basic control objective is to control the position of a free rolling ball on a plate, by applying tensions to two DC motors, according to the ball position, as measured by vision system.

**KEYWORDS:** neural network, genetic algorithm, nonlinear identification, multivariable control, intelligent control, hybrid intelligent systems.

## RESUMO

Este artigo apresenta uma nova abordagem para o projeto de um controlador *PID* (proporcional, integral e derivativo) multivariável baseado em uma rede neural e um algoritmo genético. O projeto do controlador *PID* multivariável é dividido em três etapas distintas. Na primeira etapa, uma rede neural de funções radiais de base é utilizada para identificação do processo multivariável. Na segunda etapa, o projeto do controlador é realizado, de forma *off-line*, baseado na sintonia de ganhos do controlador *PID*. Esta etapa é realizada através de uma otimização via algoritmos genéticos visando controlar o modelo do processo obtido pela rede neural de funções radiais de base. Na terceira etapa, os ganhos do controlador *PID* obtidos através de otimização genética e modelo neural são validados no processo prático. O desempenho desta proposta de projeto de controle multivariável é apresentado e discutido. O projeto do algoritmo de controle é validado de forma experimental em um processo multivariável denominado bola e chapa (*ball-and-plate*). Este processo consiste de uma chapa articulada em seu centro tal que a inclinação da chapa pode ser manipulada em duas direções perpendiculares. O objetivo básico é controlar a posição de uma bola que rola livremente sobre a chapa, aplicando-se tensões

---

Artigo submetido em 03/12/2002

1a. Revisão em 03/10/2003

2a. Revisão em 19/04/2006

Aceito sob recomendação do Ed. Assoc. Prof. Cairo Lúcio Nascimento Jr.

aos motores, estas baseadas no conhecimento da posição da bola medida pelo sistema de visão.

**PALAVRAS-CHAVE:** rede neural, algoritmo genético, identificação não-linear, controle multivariável, controle inteligente, sistemas híbridos inteligentes.

## 1 INTRODUÇÃO

As metodologias de controle avançado têm sido extensivamente aplicadas em sistemas de controle multivariável, que incluem processos mecânicos, aeroespaciais, químicos e elétricos. Entre as metodologias de controle avançado para aplicações multivariáveis destacam-se os algoritmos de controle preditivo baseado em modelo (Oliveira & Carrapico, 1998; Bordeneuve-Guibe & Vaucoret, 2001; Poncia & Bitanti, 2001), controle de estrutura variável (Jafarov, 2000; Saaj & Bandyopadhyay, 2002), controle robusto (Sana & Rao, 2001; Tan *et al.*, 2002), controle adaptativo (Haddad *et al.*, 2002; Yu & Orlov, 2002) e controle baseado em metodologias da inteligência computacional (King, 1999; Jamshidi *et al.*, 2002).

A literatura tem apresentado muitos estudos referentes ao projeto de controladores multivariáveis usando metodologias emergentes da inteligência computacional, a citar as aplicações de redes neurais (Agarwal, 1997; Chouai *et al.*, 2000; Ge *et al.*, 2001; Yu *et al.*, 2002), algoritmos evolutivos (Chipperfield & Fleming, 1996; Zhang *et al.*, 2002), sistemas nebulosos (Tanaka & Ikeda, 1998; Golea & Golea, 2002) e sistemas híbridos inteligentes (Craven *et al.*, 1999; Kawanaka *et al.*, 2000).

O controlador *PID* (proporcional + integral + derivativo) é o controlador mais popular e vastamente utilizado em malhas industriais, principalmente pela sua facilidade de implementação e por sua vasta aceitação por parte dos operadores, este um fator chave para o sucesso em um ambiente industrial.

O *PID* apresenta vantagens vinculadas ao seu baixo custo, simplicidade de implementação e, quando sintonizado adequadamente, proporciona um bom comportamento dinâmico ao processo controlado. As abordagens para a sintonia dos parâmetros de ganho proporcional, tempo integral e tempo derivativo de acordo com um período de amostragem ( $T_s$ ) do *PID*, podem ser divididas em técnicas de:

- Ziegler e Nichols (Ziegler & Nichols, 1943; Wang *et al.*, 1999);
- domínio da frequência (Iwai *et al.*, 1999);
- *autotuning* usando método do relay (Huang, 2000) e adaptativas (self-tuning control) (Åström & Wittenmark, 1995);

- otimização e heurísticas de tentativa e erro (Ulkpai & Jayasuriya, 2002);
- controle por modelo interno (Ho *et al.*, 2001);
- controle preditivo (Pomerlau *et al.*, 2000);
- escalonamento de ganhos (Jiang, 1994);
- controle robusto (Mattei, 2000);
- controle inteligente (Coelho *et al.*, 2002); e
- reconhecimento de padrões (Kraus & Miron, 1984).

Em relação ao projeto de controle *PID* multivariável, a aplicação de métodos convencionais de sintonia são eficientes para processos que podem ser modelados usando sistemas dinâmicos de baixa ordem e apresentando parâmetros fixos. Entretanto, os processos presentes no meio industrial podem apresentar características complexas, tais como processos com dinâmicas de alta ordem, presença de atraso de transporte, não-linearidades e acoplamento entre as variáveis.

Neste contexto, estudos recentes têm sido apresentados visando o projeto de controladores multivariáveis (Garces *et al.*, 1998; Adusumilli *et al.*, 1998; Ono *et al.*, 2000; Katebi *et al.*, 2000; Vásquez & Morilla, 2002). As abordagens de controle *PID* usando metodologias inteligentes também têm sido alvo de estudos na tentativa da obtenção de eficiência, robustez e flexibilidade em projetos de controle avançado (Shanley, 1995; VanDoren, 1998).

Os sistemas híbridos inteligentes são relevantes quando domínios complexos e que apresentam problemas com componentes diferentes são considerados, os quais requerem diversos tipos de processamento (ou finalidades). A concepção de que as metodologias da inteligência computacional são complementares, mais que competitivas, gerou uma ampla diversidade de configurações de sistemas híbridos inteligentes. Vale citar, por exemplo, os sistemas neuro-nebulosos, nebuloso-evolutivos, neuro-evolutivos e neuro-nebuloso-evolutivos, com diversas aplicações em identificação e controle de processos, e outras áreas afins (Khosla & Dillon, 1997; Seng *et al.*, 1999; Bonissone *et al.*, 1999; Coelho, 2001).

Este artigo apresenta uma abordagem para o projeto de controle *PID* multivariável baseado em um sistema híbrido inteligente, composto de uma rede neural (identificação do processo) e um algoritmo genético (otimização dos ganhos de um controlador *PID* multivariável). O projeto do algoritmo de controle é validado, de forma experimental, em um processo multivariável denominado bola e chapa (*ball-and-plate*).

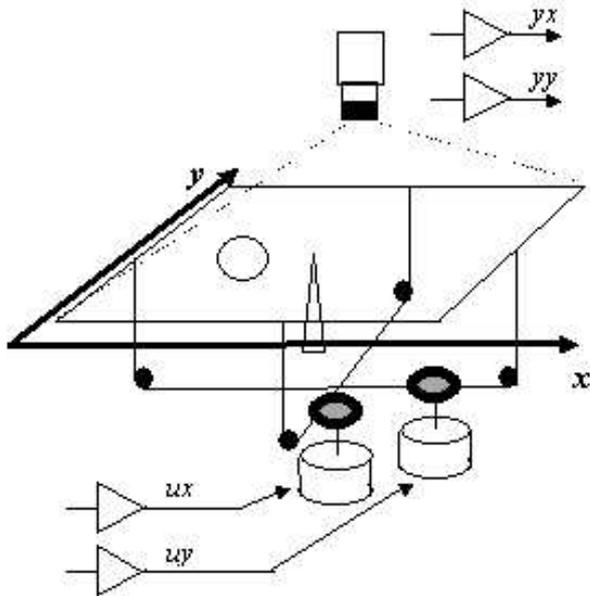


Figura 1: Processo bola e chapa.

O artigo é organizado da seguinte forma. A descrição do processo experimental bola e chapa é abordada na seção 2. Na seção 3 são apresentados os fundamentos das metodologias de inteligência computacional envolvidas e do projeto de controle *PID* multivariável. As simulações e análise dos resultados obtidos da aplicação da abordagem do projeto de controle *PID* são também apresentados na seção 3. Finalizando o artigo, a conclusão e as perspectivas de futuros trabalhos são apresentadas na seção 4, respectivamente.

## 2 DESCRIÇÃO DO PROCESSO BOLA E CHAPA

O processo “bola e chapa” é comercializado pela empresa Tecquipment em conjunto com a empresa Humusoft. O processo consiste de uma chapa (prato) articulada em seu centro tal que a inclinação da chapa pode ser manipulada em duas direções perpendiculares. Um sistema servo consistindo de dois motores *DC* é utilizado para o controle de movimento da placa e um sistema de visão é usado para medir a posição da bola. O objetivo principal é controlar a posição de uma bola que rola livremente sobre a chapa, aplicando-se tensões ao sistema de dois motores baseadas no conhecimento da posição  $(x,y)$  da bola (medida pelo sistema de visão), conforme apresentado na figura 1 (Ball & Plate Technical Manual, 1996).

O processo bola e chapa possui características: (i) não-lineares, tais como limitação na taxa de variação dos sinais de controle e constante de tempo dependendo da magnitude

e frequência dos sinais de entrada; e (ii) propriedades não-modeladas, tais como atrito e defeitos na bola e/ou superfície da chapa.

Este processo consiste de um sistema dinâmico com duas entradas (tensões aplicadas a dois motores *DC*,  $u_x$  e  $u_y$ ) e duas saídas (posições  $y_x$  e  $y_y$  da bola). O sistema é projetado para ser controlado por controladores digitais. No caso deste artigo utilizou-se um algoritmo de controle *PID* multivariável na forma digital.

## 3 PROJETO DO CONTROLADOR INTELIGENTE

A seguir, na seção 3.1 são apresentados os fundamentos de inteligência computacional, redes neurais de funções radiais de base e algoritmos genéticos. Após, na seção 3.2, são apresentadas as três etapas do projeto de controle *PID* neurogenético.

### 3.1 Fundamentos das metodologias da inteligência computacional

O termo “inteligência computacional” foi proposto por Zadeh (1994) e Bezdek (1994) que atribuem esta denominação para uma coleção de metodologias que visam explorar: a tolerância a falhas, a imprecisão e a incerteza proporcionando robustez e solução de baixo custo. Os principais membros deste consórcio de metodologias denominado de inteligência computacional (uma subdivisão da inteligência artificial) incluem as seguintes abordagens:

- redes neurais artificiais;
- computação evolutiva (algoritmos genéticos, programação evolutiva, programação genética, sistemas classificadores e estratégia evolutiva);
- sistemas nebulosos (modelos: relacionais, de Mamdani, Takagi-Sugeno-Kang, Larsen e Tsukamoto),
- raciocínio probabilístico (teoria de Dempster-Shafer e redes Bayesianas); e
- sistemas híbridos inteligentes (combinações de redes neurais, sistemas nebulosos e computação evolutiva).

Os seres humanos são fascinados com a idéia da construção de máquinas inteligentes, por duas razões: imitar o comportamento inteligente e produzir soluções funcionais para tarefas complexas. A abordagem tradicional da produção de *software*, para a resolução de problemas particulares, não pode ser aplicada facilmente, em geral, nas áreas de reconhecimento de voz, visão computacional, controle de processos

não-lineares complexos e robótica móvel. Vários problemas nestas áreas podem ser resolvidos de forma apropriada com técnicas da inteligência computacional, que sejam aptas a desenvolver soluções, na forma simbólica ou numérica, para um problema particular em consideração.

A teoria de controle moderno tem sido aplicada com sucesso em sistemas bem definidos deterministicamente ou não. Entretanto, em ambientes práticos, encontram-se muitos problemas em aplicações da engenharia de controle, onde os sistemas a serem controlados exibem características complexas, tais como: (i) não-linearidades ou incertezas no modelo; (ii) entradas dos sensores e variáveis de controle multivariáveis complexas; (iii) atraso de transporte variante no tempo; (iv) medidas apresentando incertezas e inadequação; (v) necessidade de tratamento de aspectos de segurança do processo, diagnóstico de falhas e restrições ambientais; (vi) ambientes variantes no tempo e que apresentam incertezas; e (vii) mudanças nas variáveis externas não-medidas e não-controláveis (por exemplo, variações da persistência de alimentação, temperaturas externas e pressão barométrica).

O desenvolvimento da tecnologia de processamento de informação e inteligência computacional constitui-se em uma solução alternativa para estes problemas, pois apresentam aspectos relativos à incorporação de características inspiradas na natureza e na inteligência humana em sistemas de controle automático. Estes sistemas de controle, ditos “inteligentes”, tentam imitar a maneira de tomada de decisão humana e a representação de conhecimento, motivando a atenção crescente da comunidade atuante em controle de processos (Bartos, 1997). Entre as vantagens potenciais oferecidas por estas técnicas, em relação aos esquemas convencionais de controle, pode-se ressaltar: (i) algoritmos estruturados de maneira simples para a tomada de decisões; (ii) capacidade de aprendizagem; e (iii) maior grau de autonomia.

Os sistemas de controle inteligentes têm a habilidade de aprendizado, raciocínio e tomada de decisão. Entretanto, os sistemas de controle convencional apresentam certo grau de inteligência. Entretanto, as características que diferenciam os sistemas inteligentes dos convencionais são: o grau de complexidade, presença ou não de incertezas e o tipo de informação apresentado para o sistema (Shoureshi, 1993).

Em particular, os sistemas de controle inteligentes devem possuir a habilidade de tratar incertezas, aspectos qualitativos da informação, estruturas de dados complexas, dados não-estruturados e informação de sensores. Em termos de inteligência, é esperado que um sistema de controle dito “inteligente” seja capaz de replicar (mesmo que rudimentarmente) algumas habilidades humanas, entre as quais, criação, sensação, percepção, adaptação, inferência e desenvolvimento da experiência. Além disso, no atual estágio do desenvolvi-

mento tecnológico, os sistemas de controle inteligentes apresentam limitações quanto aos aspectos de autonomia em termos de aprendizado, raciocínio, planejamento e tomada de decisões (Shoureshi, 1993; Sanz *et al.*, 1999).

O termo controle inteligente refere-se tanto a abordagens de projeto de controle de processos quanto a técnicas de implementação que emulam certas características de sistemas biológicos inteligentes. Existe uma grande variedade de tópicos e áreas do conhecimento relacionadas a controle inteligente. Tradicionalmente, as abordagens de controle de processos, inteligência artificial e pesquisa operacional têm realizado contribuições relevantes na área de controle inteligente. Além disso, uma grande diversidade de áreas do conhecimento é também envolvida, tais como engenharia elétrica, ciência da computação, engenharia mecânica, engenharia industrial, engenharia química, engenharia aeronáutica, matemática e filosofia. Esta vasta variedade de tópicos relevantes e pontos de vista fazem com que as fronteiras da área de controle inteligente sejam complexas de definir (Passino & Özgüner, 1996).

A seguir são apresentados os fundamentos de redes neurais e algoritmos genéticos, conforme utilizados no projeto de um controlador inteligente.

### 3.1.1 Rede neural de funções radiais de base

As redes neurais artificiais (*RNs*) são motivadas por sistemas neurais biológicos, com intenção de simular, mesmo que rudimentarmente, a forma como o cérebro aprende, recorda e processa informações. As *RNs* são projetadas como um sistema interconectado de elementos de processamento (neurônios), cada um dos quais com um número limitado de entradas e saídas.

Entre as características que melhor explanam o potencial das redes neurais sobressaem-se devido a adaptabilidade, paralelismo, multidisciplinaridade e tolerância a falhas, o que proporciona uma ferramenta matemática potencial para aplicações práticas e complexas. As *RNs* proporcionam, usualmente, conhecimento quantitativo não-paramétrico e são adequadas para identificação de sistemas, aprendizado e adaptação de processos complexos.

Entretanto, os aspectos relativos a convergência e as propriedades de estabilidade das *RNs* ainda são assuntos de diversas pesquisas e gerador de divergências na comunidade de controle. A seguir é descrita a abordagem de uma *RN*, a *RN* de funções radiais de base, esta utilizada para a identificação de um processo multivariável.

A justificativa para o uso de uma *RN* de funções radiais de base é devido a esta apresentar estrutura topológica simples para aproximação local, precisão na aproximação de dinâ-

mica não-linear e aprendizado rápido (Peng *et al.*, 2003), sendo uma boa alternativa ao uso do *perceptron* multicamadas (Rumelhart *et al.*, 1986). A clássica *RN perceptron* multicamada com aprendizado pelo método da retropropagação do erro, que apesar de ser um aproximador global apresenta custo computacional significativo e velocidade de aprendizado baixa.

Uma função radial de base,  $\phi$ , apresenta uma saída simétrica em torno de um centro associado,  $\mu_c$ , tal que  $\phi_c(x) = \phi(\|x - \mu_c\|)$ , onde  $\|\cdot\|$  é a norma vetorial. Uma condição suficiente para uma função ser candidata à função radial de base é a de ser radialmente simétrica e a de possuir primeira derivada monotônica. A função Gaussiana é o tipo de função radial de base utilizada neste artigo.

A abordagem de determinar os componentes individuais, dada uma superposição de Gaussianas, foi explorado nos anos 60. Subseqüentemente, a utilização de funções radiais de base para interpolação numérica e aproximação de funções foi demonstrada por Powell (1985) e Poggio & Girosi (1990).

As funções Gaussianas são caracterizadas por um parâmetro de escala (ou comprimento),  $\sigma$ , e isto é verdade para muitas outras classes de funções radiais de base. Tais classes podem ser representadas como:

$$\phi_c(x) = \phi(\|x - \mu_c\|; \sigma) \quad (1)$$

Um conjunto de funções radiais de base pode servir como uma base para representação de uma diversificada classe de funções, que são expressas como combinações lineares de funções radiais de base, tal que:

$$y(x) = \sum_{j=1}^M \omega_j \phi(\|x - \mu_c\|) \quad (2)$$

Uma rede neural de funções radiais de base (*RN-RBF*), ou também denominada na literatura de *RN* de base radial, conforme apresentado na figura 2, é um agrupamento de equações similar a concepção de uma *RN feedforward* com três camadas: a(s) entrada(s), camada oculta e nó(s) de saída.

É relevante também mencionar que uma *RN-RBF* é um aproximador universal de funções contínuas com dada precisão, de forma análoga à rede neural *perceptron* multicamadas (Cybenko, 1989), conforme prova apresentada por Park & Sandberg (1991, 1993). No caso da *RN-RBF*, cada unidade oculta representa uma função radial de base simples, com comprimento (dispersão) e posição dos centros associados. As unidades ocultas são algumas vezes denominadas de centróides ou núcleos. Cada unidade de saída executa uma soma

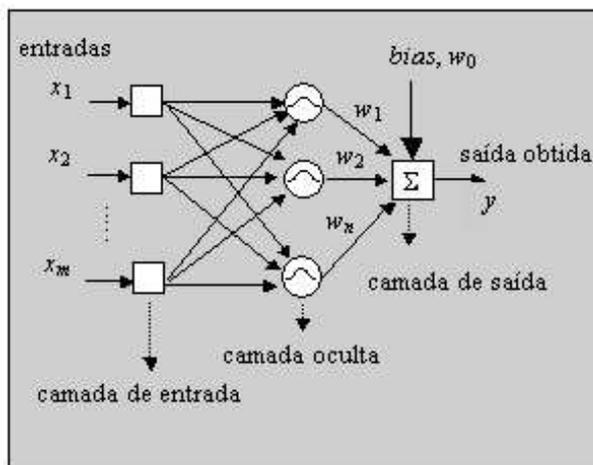


Figura 2: Uma *RN-RBF* com uma camada intermediária.

ponderada de unidades ocultas, utilizando os valores de  $\omega_j$  como pesos.

O projeto da *RN-RBF* adotado, neste artigo, é o de treinamento em dois estágios. Esta forma de treinamento é uma alternativa atraente para configurações *off-line* de treinamento, e visa a estimação de parâmetros de forma desacoplada, em dois estágios, a fim de determinar: (i) o número de centros e os valores de  $\mu_c$  e  $\sigma_j$ , sendo que a posição dos centros é determinada através do algoritmo de clusterização *k*-médias (Moody & Darken, 1989); e (ii) os pesos das unidades de saída pela técnica da pseudo-inversa, para os centros e comprimentos obtidos no estágio (i).

Entretanto, é relevante mencionar que existem outras técnicas que podem ser utilizadas para a definição dos parâmetros de projeto e otimização da *RN-RBF*, tais como mínimos quadrados ortogonal em batelada (Chen *et al.*, 1991) e recursivo (Gomm & Yu, 2000), método de Levenberg-Marquardt (Peng *et al.*, 2003) algoritmos genéticos (Liu & Kadirkamanathan, 1999) e abordagens de clusterização (Uykan *et al.*, 2000).

### 3.1.2 Algoritmos genéticos

Os paradigmas da computação evolutiva (ou computação evolucionária) são também denominados algoritmos evolutivos (ou algoritmos evolucionários). Os algoritmos evolutivos (AEs) são sistemas computacionais para resolução de problemas baseados nos princípios da teoria evolutiva e na genética. Uma variedade de algoritmos evolutivos tem sido desenvolvida e todos dividem uma base conceitual comum, através de procedimentos de seleção, mutação e recombinação. Entre os algoritmos de busca estocástica destacam-se os algoritmos genéticos.

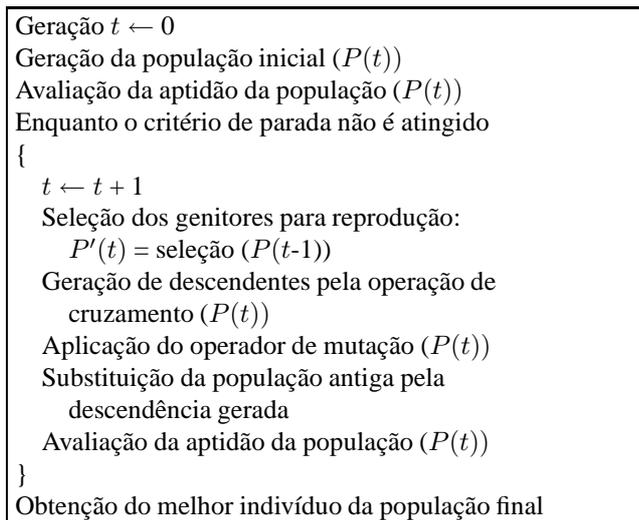


Figura 3: Pseudocódigo básico dos dados nos AGs canônicos.

Os algoritmos genéticos (AGs) são especialmente úteis para as tarefas de otimização global, onde os métodos determinísticos podem levar a mínimos locais. Os AGs são técnicas robustas e eficientes em espaços de procura irregulares, complexos e apresentando múltiplas dimensões. Um AE caracteriza-se por: (i) operar em uma população de pontos (soluções); (ii) não requerer cálculos de derivadas e informação sobre o gradiente da função objetivo; (iii) trabalhar com a codificação de seu conjunto de parâmetros, não com os próprios parâmetros (representação binária); (iv) realizar transições probabilísticas, em vez de regras determinísticas; (v) necessitar apenas da informação sobre o valor da função objetivo para cada indivíduo da população; (vi) apresentar simplicidade conceitual; (vii) ser pouco afetado, quanto à eficiência, quando descontinuidades e ruídos estão presentes nos dados do problema. As características (iii) a (v) não são comuns a todos os AEs, mas geralmente presentes nos algoritmos genéticos (Goldberg, 1989; Coelho & Coelho, 1999).

O ciclo básico dos dados num AG canônico (representação binária) é baseado nos passos apresentados nas figuras 3 e 4.

O projeto do AG utilizado para sintonia dos parâmetros do controlador PID multivariável segue a abordagem de operador de reprodução baseado em gradiente proposto por Pham & Jin (1995).

A justificativa de usar o operador de reprodução é acrescentar ao AG uma natureza combinando tanto características determinísticas quanto estocásticas, ou seja, acrescentar um mecanismo de busca local ao AG. O operador de reprodução é empregado para superar as limitações usualmente associadas ao método de seleção por roleta. Estas limitações incluem: perda do indivíduo mais apto da população, dominância ex-

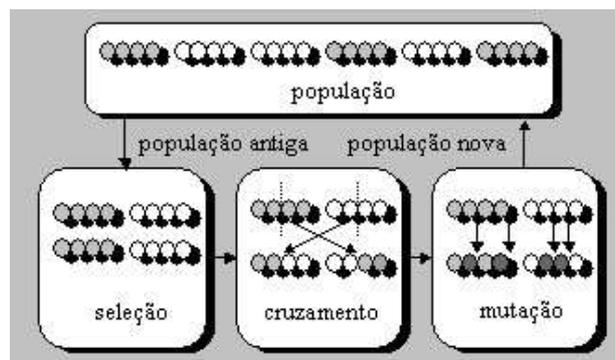


Figura 4: Ciclo básico dos dados nos AGs canônicos.

cessiva de indivíduos muito aptos (elite dominante) e a habilidade limitada de explorar novos pontos no espaço de busca. Para a implementação deste operador um novo vetor de soluções é calculado a cada geração pela equação:

$$x'(t) = x(t) + \psi_i \frac{[f_{max}(t) - f_i(t)]}{f_{max}(t)} [x_b(t) - x_i(t)], \quad (3)$$

$$i = 1, \dots, N_{ind}$$

onde  $\psi_i$  é um coeficiente positivo,  $x_b(t)$  é o vetor de parâmetros com maior aptidão,  $f_{max}(t)$ , e  $N_{ind}$  é o número de soluções (indivíduos) avaliadas a cada geração do AG. Pham & Jin (1995) recomendam  $\psi_i$  entre 0,0 e 2,0 para um comportamento adequado do operador de reprodução. Este operador pode ser utilizado para aprimorar a qualidade da solução obtida, a cada geração, após aplicar-se os operadores de cruzamento e mutação (Coelho & Coelho, 2000).

## 3.2 Controlador híbrido neuro-genético

O projeto do controlador PID multivariável é dividido em três etapas distintas. Na primeira etapa, a RN-RBF é utilizada para identificação do processo.

Na segunda etapa, o projeto do controlador é realizado baseado na sintonia de ganhos do controlador PID realizada por meio de otimização via AG visando controlar o modelo matemático do processo obtido pela RN-RBF.

Na terceira etapa, os ganhos do controlador PID multivariável obtidos com otimização via AG e modelo matemático obtido através de rede neural são validados no processo prático, mostrando a eficiência e robustez desta abordagem de projeto. Estas três etapas são detalhadas a seguir.

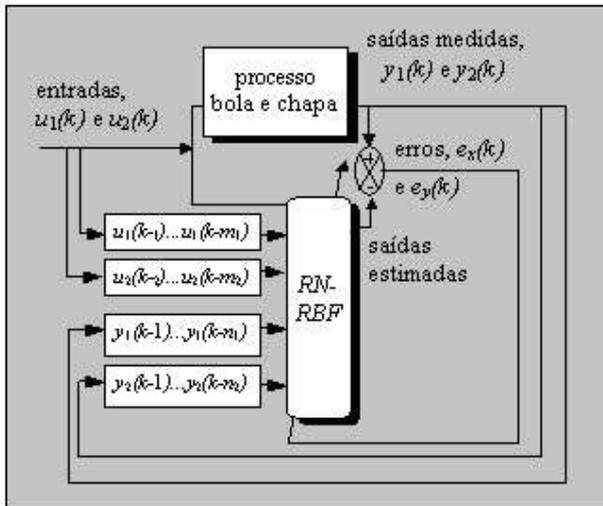


Figura 5: Configuração da identificação de processo multivariável bola e chapa através da RN-RBF.

### 3.2.1 Etapa 1: Coleta de dados e identificação do modelo usando RN-RBF

A coleta de dados das entradas (tensões aplicadas aos motores DC do processo bola e chapa),  $u_1$  (equivalente a  $u_x$  da figura 1) e  $u_2$  (equivalente a  $u_y$  da figura 1), e das saídas (posições  $x$  e  $y$  da bola sobre a chapa do processo bola e chapa)  $y_1$  (equivalente a  $y_x$  da figura 1) e  $y_2$  (equivalente a  $y_y$  da figura 1) foi realizada em um procedimento de identificação em malha aberta. O procedimento de identificação adotado nesta etapa é dividido nas seguintes fases: (i) obtenção dos dados do processo; (ii) escolha da estrutura utilizada para representar o modelo; (iii) escolha do algoritmo para tratar o modelo matemático do processo; (iv) estimação dos parâmetros do modelo; e (v) validação do modelo (Ljung, 1999).

O procedimento de identificação multivariável do processo bola e chapa através de uma RN-RBF (modelo série-paralelo) é apresentado na figura 5.

A figura 6 apresenta as entradas,  $u_1$  e  $u_2$ , (sinais binários pseudo-aleatórios de magnitude  $-0,3$  ou  $0,3$ ), utilizadas para identificação através da RN-RBF.

A RN-RBF utilizada para identificação consiste de 4 neurônios de entrada compostos dos vetores  $\{u_1(k), u_2(k), y_1(k), y_2(k), y_1(k-1), y_2(k-1)\}$  e 2 neurônios de saída,  $y_1(k+1)$  e  $y_2(k+1)$ .

Nos experimentos realizados foram utilizadas 300 amostras para a fase de estimação e 300 amostras para a fase de validação do modelo obtido pela RN-RBF (com período de amostragem  $T_s=100$  ms).

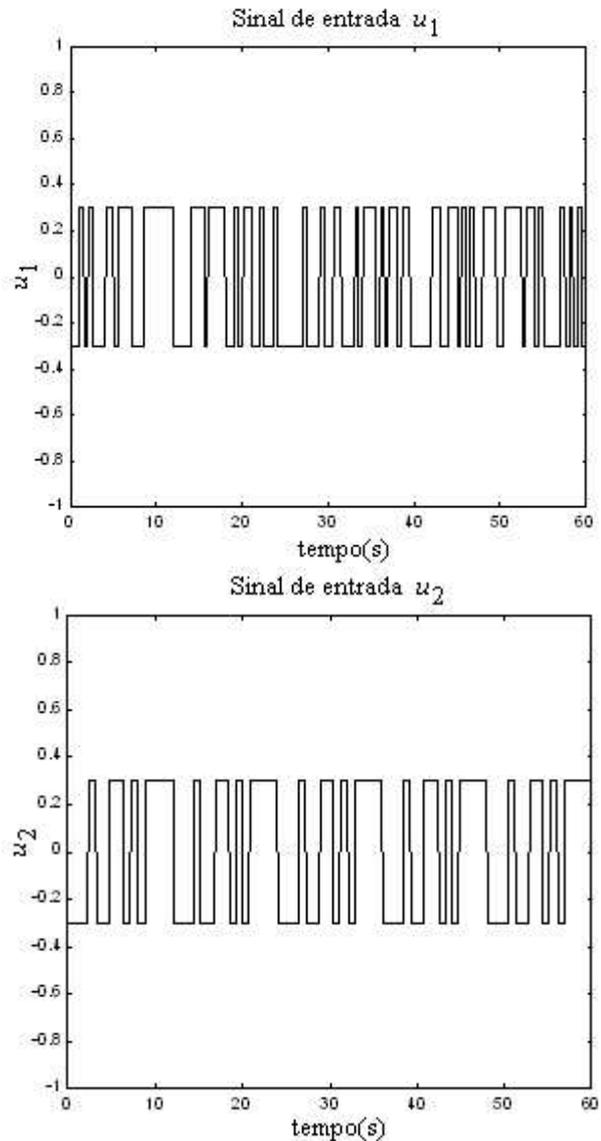


Figura 6: Dados de entrada do processo bola e chapa para o procedimento de identificação usando RN-RBF.

Foram realizados vários experimentos para determinação do número de funções Gaussianas na camada oculta da RN-RBF (de 2 até 9 funções Gaussianas). Os resultados obtidos para a soma do erro quadrático,  $ISE_i$ , são apresentados nos histogramas das figuras 7 a 10, observando-se que  $i \in [1, 2]$  representa o índice da saída  $y_i$  do processo.

Analisando-se os histogramas (figuras 7 a 10) nota-se que a RN-RBF apresentou resultados promissores na fase de estimação (interpolação) e validação (generalização) para a saída  $y_1$  com 6 a 8 funções Gaussianas na camada oculta da RN-RBF. Além disso, a RN-RBF apresentou resultados promissores na aproximação da saída  $y_2$  para 2, 3, 4, 7, 8 e 9 Gaussi-

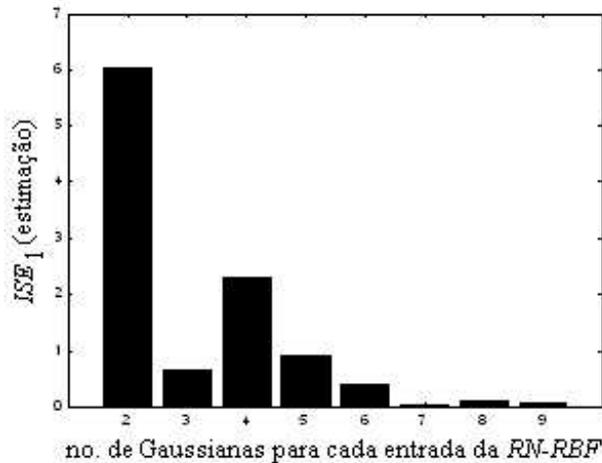


Figura 7: Histograma da soma do erro quadrático,  $ISE_1$ , da saída  $y_1$  em relação a saída da  $RN-RBF$  (fase de estimação: amostras 1 a 300).

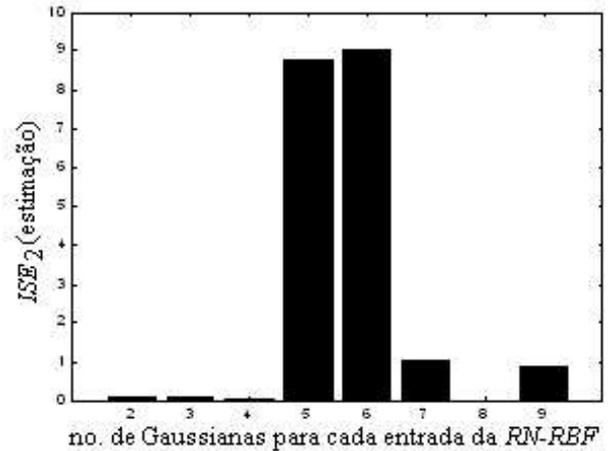


Figura 9: Histograma da soma do erro quadrático,  $ISE_2$ , da saída  $y_2$  em relação a saída da  $RN-RBF$  (fase de estimação: amostras 1 a 300).

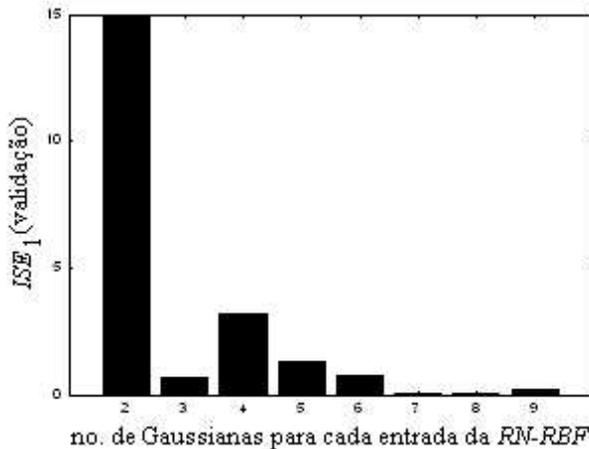


Figura 8: Histograma da soma do erro quadrático,  $ISE_1$ , da saída  $y_1$  em relação a saída da  $RN-RBF$  (fase de validação: amostras 301 a 600).

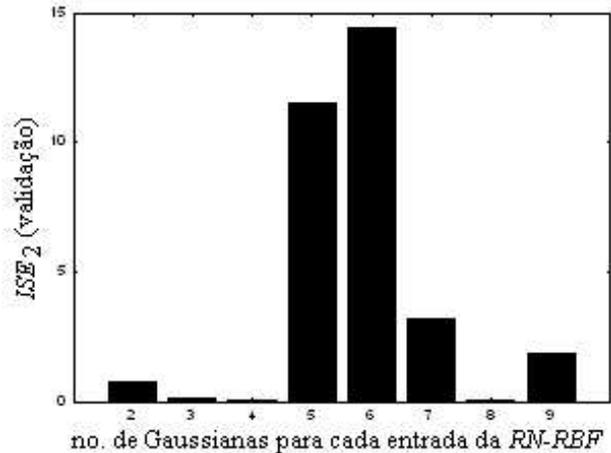


Figura 10: Histograma da soma do erro quadrático,  $ISE_2$ , da saída  $y_2$  em relação a saída da  $RN-RBF$  (fase de validação: amostras 301 a 600).

anas na camada oculta. Estes resultados motivaram a escolha do projeto da  $RN-RBF$  com 8 funções Gaussianas na camada oculta. Os resultados de 8 Gaussianas para cada entrada da  $RN-RBF$  são apresentados nas figuras 11 a 14.

Nota-se pelos resultados apresentados nas figuras 11 a 14, que a  $RN-RBF$  com 8 funções Gaussianas, na camada oculta, aproximou as saídas reais do processo com maior precisão, ou seja, as saídas estimadas sobrepuseram às duas saídas,  $y_1$  e  $y_2$ , do processo bola e chapa. A tabela 1 apresenta um resumo estatístico dos resultados obtidos com o projeto da  $RN-RBF$ .

Tabela 1:

### 3.2.2 Etapa 2: Otimização dos ganhos do controlador PID multivariável

O PID multivariável abordado no projeto tem configuração desacoplada e é regido pelas seguintes equações a diferenças:

$$u_n(k) = u_n(k-1) + q_{0,n}e_n(k) + q_{1,n}e_n(k-1) + q_{2,n}e_n(k-2) \quad (4)$$

onde

$$q_{0,n} = K_{p,n} \left( 1 + \frac{T_s}{2T_{i,n}} + \frac{T_{d,n}}{T_s} \right) \quad (5)$$

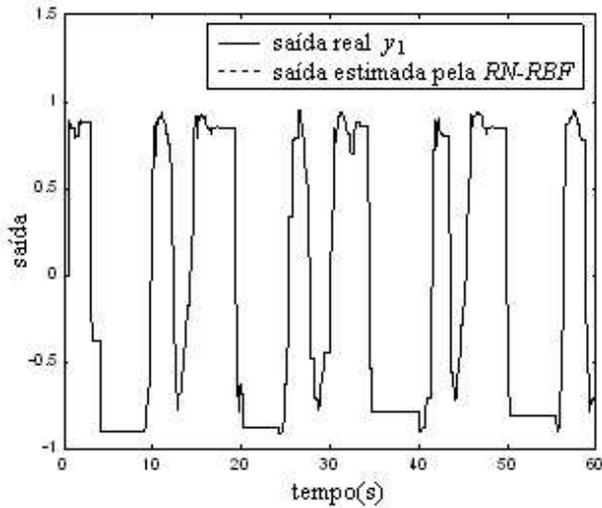


Figura 11: Saída  $y_1$  do processo e saída estimada pela *RN-RBF*.

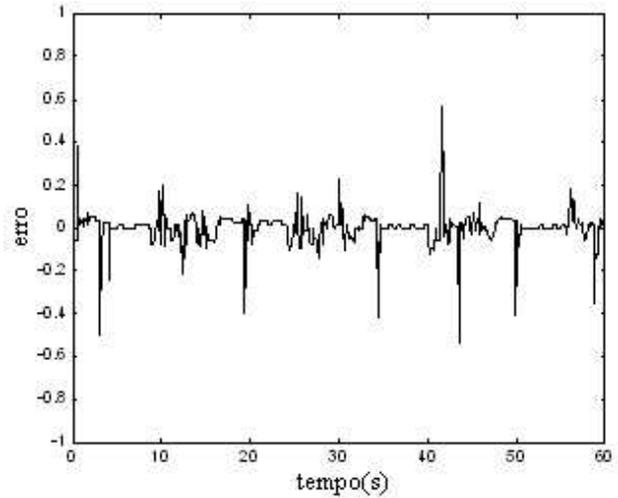


Figura 13: Erro entre a saída  $y_1$  do processo (bola e chapa) e a saída estimada pela *RN-RBF*.

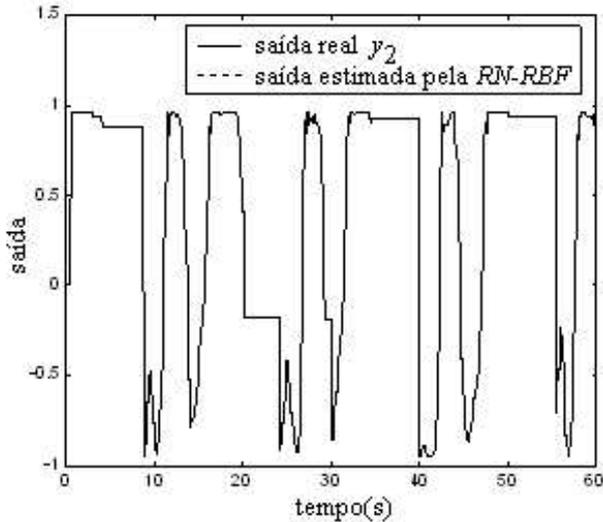


Figura 12: Saída  $y_2$  do processo e saída estimada pela *RN-RBF*.

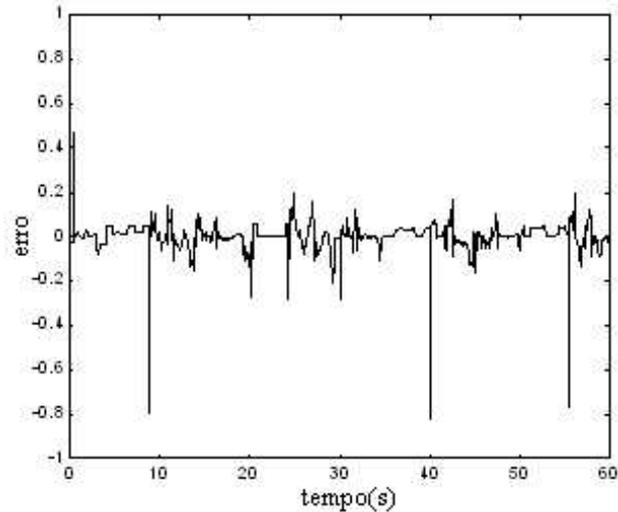


Figura 14: Erro entre a saída  $y_2$  do processo (bola e chapa) e saída estimada pela *RN-RBF*.

$$q_{1,n} = -K_{p,n} \left( 1 + \frac{2T_{d,n}}{T_s} - \frac{T_s}{2T_{i,n}} \right) \quad (6)$$

$$q_{2,n} = K_{p,n} \frac{T_{d,n}}{T_s} \quad (7)$$

onde  $n$  é o índice do controlador *PID* ( $n = 1$ , controlador *PID* para a coordenada  $x$ ;  $n = 2$ , controlador *PID* para a coordenada  $y_i$  da bola do processo bola e chapa),  $e_n(k)$  é o erro dado pela diferença entre a saída,  $y_n(k)$ , do processo e a referência desejada,  $y_{r_n}(k)$ ;  $\Delta u_n(k)$  é a variação do sinal

de controle do controlador  $n$ .

A sintonia dos parâmetros do controlador *PID* multivariável é realizada através de otimização via *AG*. O modelo matemático obtido pela *RN-RBF* (1ª etapa) é usado na otimização genética dos ganhos do controlador *PID*, conforme apresentado na figura 15. A aptidão (*fitness*),  $f(J)$ , dos indivíduos (soluções para o problema) é dada por:

$$f(J) = \frac{k_e}{(1 + J)} \quad (8)$$

Tabela 1: Resultados quantitativos obtidos da *RN-RBF* com 8 funções Gaussianas em relação as saídas reais,  $y_1$  e  $y_2$ , do processo bola e chapa.

saída $y_1$ estimada pela <i>RN-RBF</i>	fase de estimação	fase de validação
$ISE_1$	0,1156	0,0634
máximo erro	-0,5025	0,5742
média do erro	0,0011	$-8,424 \times 10^{-4}$
desvio padrão do erro	0,0678	0,0751
saída $y_2$ estimada pela <i>RN-RBF</i>	fase de estimação	fase de validação
$ISE_2$	0,0027	0,0458
máximo erro	-0,7954	-0,8230
média do erro	$-1,721 \times 10^{-4}$	$7,158 \times 10^{-4}$
desvio padrão do erro	0,0769	0,0805

onde  $k_e$  é um fator de escala. A função custo escolhida para ser utilizada na função de aptidão é

$$J = \sum_{k=1}^N \{ e_x(k)^2 + \Delta e_x(k)^2 + e_y(k)^2 + \Delta e_y(k)^2 \} \quad (9)$$

onde  $N$  é o número de amostras avaliado pelo AG.

Nas simulações de otimização *off-line* no projeto de controle *PID* digital via AGs foram utilizados os seguintes parâmetros de projeto:

- população: 30 indivíduos;
- número máximo de gerações: 100;
- experimentos realizados: 20;
- espaço de busca dos ganhos do *PID*: entre 0 e 35;
- restrição a taxa de variação da ação de controle: 1,338/s;
- restrições à ação de controle:
- coordenada  $x : u_1 \in [-1; 1]$  e coordenada  $y : u_2 \in [-1; 1]$ .
- restrições da saída do processo:
- coordenada  $x : y_1 \in [-1; 1]$ , e coordenada  $y : y_2 \in [-1; 1]$ .

Os melhores parâmetros obtidos com a otimização via AGs, após 20 experimentos, do *PID* foram:  $K_{p,1} = 0,3502$ ,  $T_{i,1} = 29,1667$ ,  $T_{d,1} = 0,9714$ ,  $K_{p,2} = 0,3496$ ,  $T_{i,2} =$

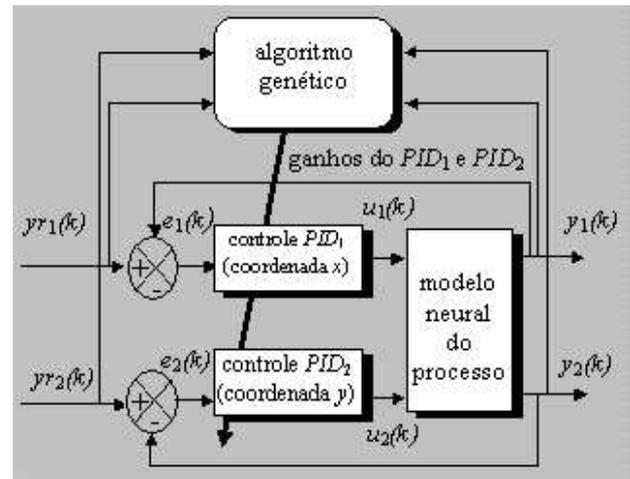


Figura 15: Otimização dos ganhos do controlador *PID* multivariável.

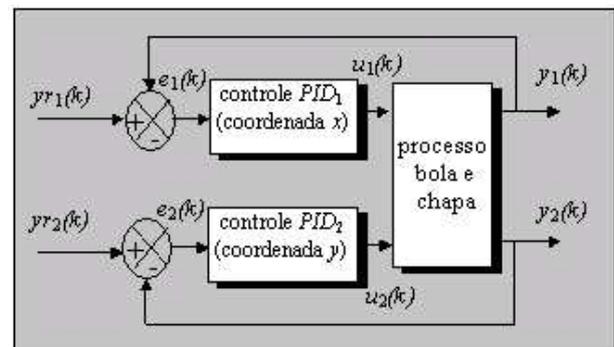


Figura 16: Diagrama de validação da sintonia por AGs do controlador *PID* no processo bola e chapa.

5,0007,  $T_{d,2} = 0,8571$ , onde foi obtida uma aptidão  $f(J)=0,8197$  para  $yr_1(k)=0,4$  e  $yr_2(k)=0,4$ , onde  $k$  varia de 1 a  $N$ . Foi usado um período de amostragem adotado de  $T_s = 100 \text{ ms}$  e fator de escala  $k_e = 15$ .

### 3.2.3 Etapa 3: Validação do projeto do controlador *PID* baseado na abordagem neuro-genética

Nesta etapa, os seis ganhos do controlador *PID* multivariável, otimizado pelo AG baseado em modelo obtido pela *RN-RBF*, ou seja,  $K_{p,1}$ ,  $T_{i,1}$ ,  $T_{d,1}$ ,  $K_{p,2}$ ,  $T_{i,2}$ ,  $T_{d,2}$ , são validados no processo prático, conforme apresentado no diagrama da figura 16.

Os resultados experimentais (validação) do projeto do controlador *PID* multivariável são apresentados para duas situações nas figuras 17 e 18, respectivamente. Na primeira situação  $yr_1(k)=0,4$  e  $yr_2(k)=-0,5$  (mesma situação abordada para otimização dos controladores *PID* por AGs com modelo

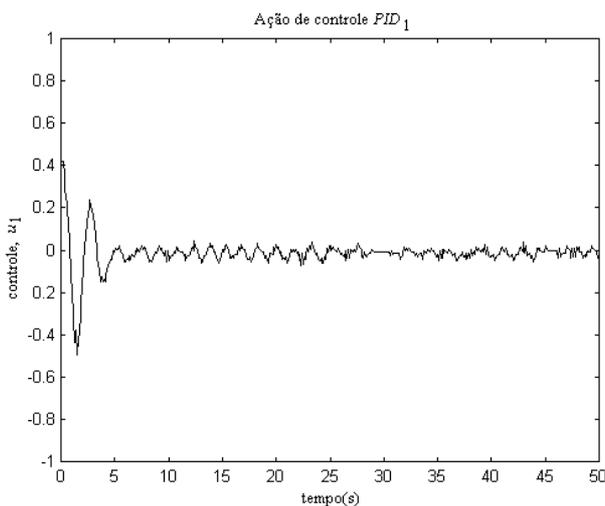
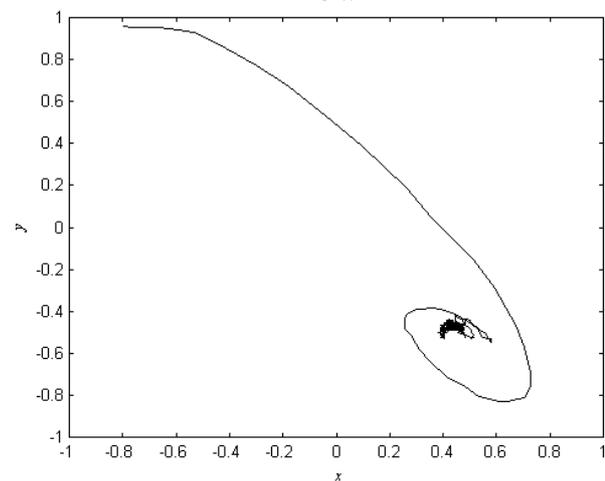
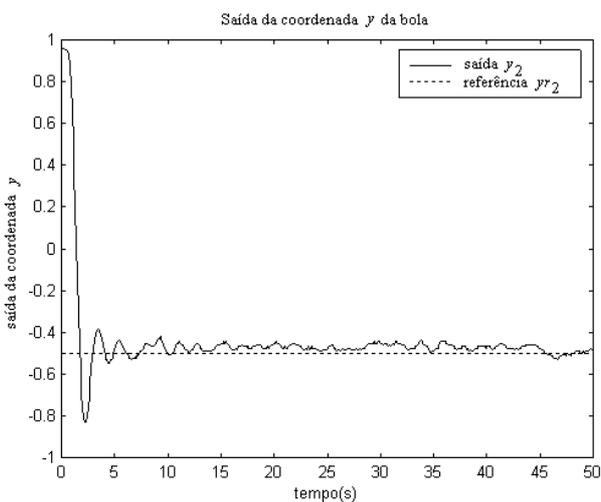
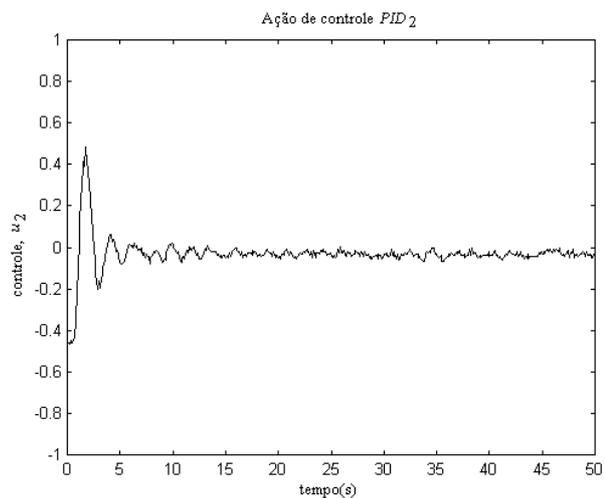
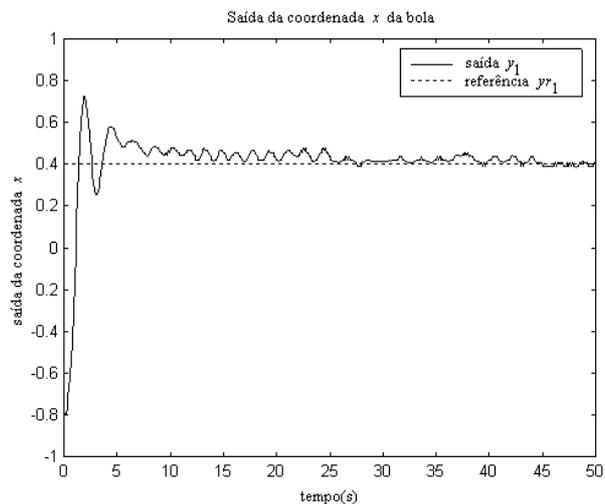


Figura 17: Resultados experimentais do controle *PID* multi-variável do processo bola e chapa (caso 1).

para  $y_{r1}(k)=0,4$  e  $y_{r2}(k)=-0,5$ , obteve-se erro médio quadrático para as saídas  $y_1$  e  $y_2$  do processo de 0,0245 e 0,0471, respectivamente. Nota-se que os controladores apresentaram resposta rápida as referências sugeridas, mas com comportamento oscilatório no caso onde as saídas do processo estavam próximas aos valores desejados. Além disso, é observado também a presença de sobre-elevação nas saídas  $y_1$  e  $y_2$  do processo nos primeiros 10 segundos de experimento; entretanto, o desempenho das saídas melhoraram sensivelmente após 25 segundos.

Na segunda situação abordada, isto é, para  $y_{r1}(k)=0,3$  e  $y_{r2}(k)=-0,2$ , o erro médio quadrático para as saídas  $y_1$  e  $y_2$  do processo são 0,0238 e 0,0102, respectivamente. Observa-se que os valores dos erros médios quadráticos foram próximos nas duas situações analisadas. Entretanto, nota-se que para cada uma das situações abordadas os controladores projetados obtiveram respostas bem distintas por se tratarem de

neural). Em uma segunda situação, adotou-se  $y_{r1}(k)=0,3$  e  $y_{r2}(k)=-0,2$ , para todas amostras, ou seja,  $k$  varia de 1 a  $N$ .

Para a primeira situação em malha fechada analisada, isto é,

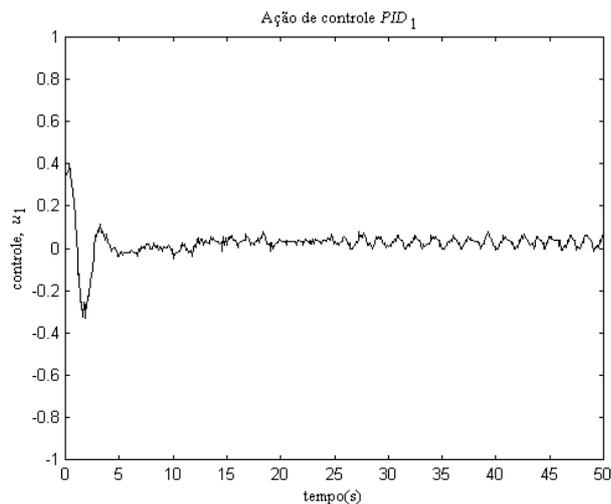
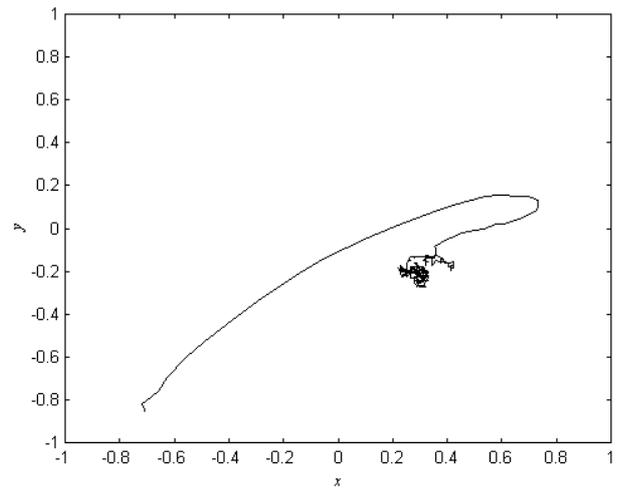
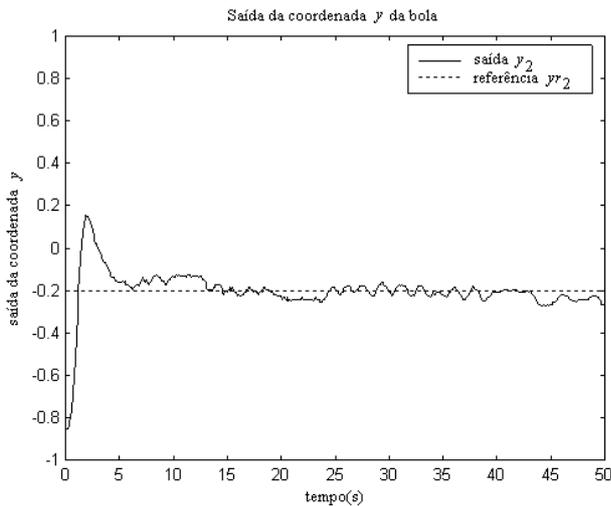
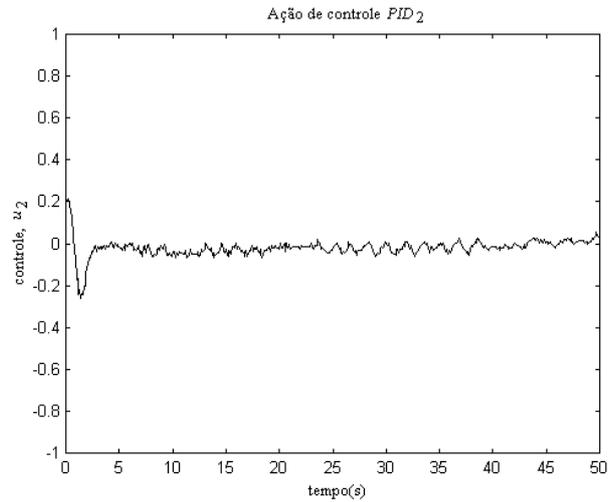
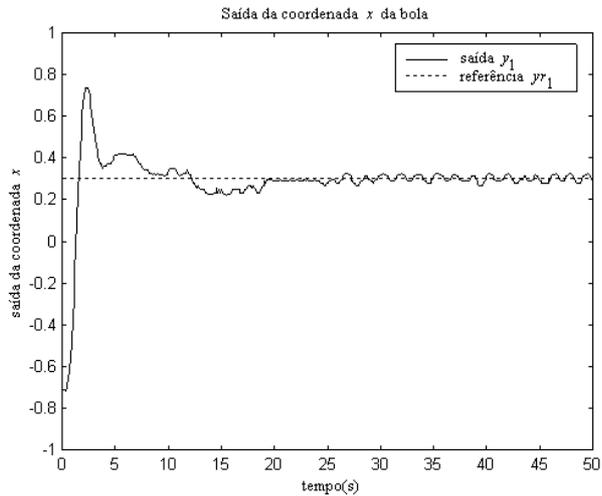


Figura 18: Resultados experimentais do controle *PID* multi-variável do processo bola e chapa (caso 2).

experimentos com condições iniciais distintas (ver as trajetórias  $x$  e  $y$  apresentadas nas figuras 17 e 18).

Baseado nos resultados obtidos, também fica evidente que os erros das saídas  $y_1$  e  $y_2$  do processo decrescem rapidamente,

mas não caracterizam um desempenho em malha fechada com erro nulo em regime permanente, nas situações analisadas. Neste contexto, nas duas situações abordadas nos experimentos, nota-se também um comportamento oscilatório da posição da bola sobre a chapa quando as saídas obtidas estão próximas dos valores das saídas desejadas, pois o processo é sensível a pequenas variações de movimentação da chapa. Este comportamento deve-se principalmente à dinâmica não-modelada do processo, pois: (i) a bola utilizada no processo é de tênis de mesa, portanto muito leve e apesar do movimento da chapa nesta situação ter sido suave, ocorre a presença de trepidação quando a bola está quase parada; (ii) a bola empregada nos experimentos possui defeitos, ou seja, não é perfeitamente esférica dificultando uma movimentação uniforme sobre a chapa; e (iii) a influência da calibração do sistema de visão para capturar de forma precisa a posição da bola sobre a chapa (a chapa é de cor preta, mas em determinadas condições gera reflexo indesejado) e enviar para

o *software* de controle o valores das coordenadas  $x$  e  $y$  do centro da bola.

Apesar destas dificuldades, os resultados experimentais demonstram a potencialidade do projeto do *PID* multivariável desacoplado baseado na *RN-RBF* e um *AG*, no controle do processo bola e chapa que possui características não-lineares, comportamento instável e dinâmicas não-modeladas desafiadoras para o projeto de controladores.

Neste contexto, estudos futuros para obtenção de melhores controladores podem ser realizados objetivando a implementação de um procedimento de identificação em malha fechada do processo (Landau *et al.*, 2000) e o aprimoramento das metodologias de inteligência computacional visando o projeto de controladores multivariáveis que levem em consideração a interação entre as diferentes malhas de controle.

## 4 CONCLUSÃO E TRABALHOS FUTUROS

Nos últimos 50 anos, as malhas de controle *PID* (proporcional, integral e derivativo) têm sido utilizadas na maioria dos processos presentes no meio industrial. Entre os motivos da utilização de estratégias convencionais de controle pode-se mencionar: a simplicidade, a robustez, o número reduzido de parâmetros a serem configurados, o conhecimento intuitivo sobre o desempenho destas estratégias de controle e um requerimento mínimo de conhecimento matemático e técnico é geralmente necessário. No entanto, as limitações destas estratégias de controle estão relacionadas com o número reduzido de graus de liberdade que elas oferecem.

Neste artigo foi apresentada uma nova abordagem para o projeto de algoritmo de controle *PID* multivariável desacoplado baseado em *RN-RBF* e *AG*. O estudo de caso de controle de um processo multivariável não-linear bola e chapa. Este problema é um desafio em controle e pode ser muito interessante para o estudo e teste de abordagens de controle multivariável. Os resultados preliminares apresentados neste artigo foram promissores, pois o projeto de controle foi testado com sucesso em duas situações práticas de seguimento de referência.

Os resultados obtidos com o projeto de identificação do processo bola e chapa através da *RN-RBF* foram precisos e apropriados à aplicação em estudo. Esta abordagem pode ser estendida para concepções de projeto de controle baseado em modelo, tais como controle adaptativo indireto ou preditivo. Entretanto, se fazem necessários estudos mais aprofundados em relação a aspectos de: (i) conhecimento *a priori* de algumas características do processo a ser controlado, tais como ordem e atraso de transporte; (ii) configuração de algoritmos de poda (*pruning*) para a implementação da *RN-RBF*; e (iii)

implementação de algoritmos de treinamento robustos tanto para buscas globais quanto locais com a implementação de algum procedimento de otimização que minimize objetivos conflitantes (múltiplos objetivos).

Em relação a *RN-RBF*, esta foi útil para a identificação de processos multivariáveis, particularmente para processos que têm pontos de operação típicos, pois estas *RNs* mapeiam aproximações locais. O aprendizado da *RN-RBF* é particularmente simples e rápido em relação à rede neural *perceptron* multicamada, muito utilizada em aplicações de identificação não-linear.

Um aspecto negativo da *RN-RBF* é a de que podem sofrer da “maldição” da dimensionalidade, pois quando a dimensão do vetor de entrada é aumentada a dimensionalidade dos centros das funções de base tem, geralmente, aumento exponencial para representar o processo multivariável adequadamente. Além disso, a adição de centros pode também ser requerida para tratar o aumento da complexidade causada por dimensões de entrada adicionais.

Uma dificuldade na configuração de *RNs* para aplicações de identificação é o aspecto relativo ao domínio temporal. Se a ordem de um sistema é conhecida, a utilização do método *tapped-delay-line* permite a um sistema dinâmico ser modelado com uma *RN* estática. Entretanto, quando a ordem é desconhecida, muitas aplicações podem levar o projetista a adotar um número elevado de entradas. Tal procedimento pode levar a modelos matemáticos não apropriados devido ao grande número de entradas necessário.

Quanto ao projeto da otimização do *PID*, os *AGs* foram utilizados com sucesso, devido as seguintes características de: (i) tratarem adequadamente os sistemas sujeitos a restrições; (ii) não requererem as informações relativas a derivadas, estas usualmente necessárias em métodos convencionais de otimização; (iii) adequarem-se à implementação em paralelo e distribuídas; (iv) possibilitarem a utilização do conhecimento obtido *a priori* pelo projetista; e (v) tratarem com sistemas e espaços de busca complexos (Coelho & Coelho, 1999).

Contudo, algumas limitações estão presentes nos *AGs*. Os algoritmos evolutivos tratam-se de métodos estocásticos e seu desempenho varia de execução para execução (a menos que o mesmo gerador de números aleatórios com a mesma semente é utilizado).

Em futuros estudos para continuação deste projeto visa-se adotar outras metodologias de projeto para o aprimoramento dos resultados apresentados neste artigo. Entre as quais pode-se mencionar: (i) realização de experimentos em malha fechada com a utilização de um controlador proporcional, neste caso visando-se abranger toda a região de operação do processo bola e chapa e não apenas a região de interesse,

(ii) estudo comparativo de técnicas para otimização de parâmetros da *RN-RBF*, tais como mínimos quadrados ortogonal (Lin *et al.*, 2001), algoritmos evolutivos (Yen & Lu, 2002), decomposição em valores singulares (McLoone *et al.*, 1998) e técnicas de regularização de Tikhonov (Hansen, 1998), (iii) projeto desacoplado e/ou hierárquico da *RN-RBF* para identificação e controle de processos multivariáveis, e (iv) realização de estudos comparativos de controle *PID* convencional e controle *PID* nebuloso.

As estratégias de controle avançadas, no qual um sistema neuro-evolutivo se enquadra, permitem o aprimoramento do desempenho dos sistemas de controle se comparadas com as estratégias convencionais. Entretanto, na maioria das vezes, o projetista necessita configurar um número significativo de parâmetros que, em alguns casos, pode dificultar o domínio do conhecimento pelos usuários, que não sejam especialistas na utilização destas técnicas de controle. Muitas vezes, também, é necessária a utilização de diferentes pacotes (ou mesmo o desenvolvimento de um sistema) para a implementação de diferentes estratégias de controle.

Em síntese, o projeto adequado de metodologias de controle avançadas, considerando um compromisso entre desempenho e complexidade, pode oferecer uma ferramenta eficiente para a comunidade de controle atuante na academia e meio industrial.

## AGRADECIMENTOS

Agradecimentos às valiosas e pertinentes contribuições dos revisores para o aprimoramento da versão atual deste artigo.

## REFERÊNCIAS

- Adusumilli, S.; Rivera, D.E.; Dash, S.; Tsakalis, K. (1998). Integrated MIMO identification and robust PID controller design through loop shaping, *Proceedings of the American Control Conference*, Vol. 2, Philadelphia, USA, pp. 1230-1234.
- Agarwal, M. (1997). A systematic classification of neural-network-based control, *IEEE Control Systems*, Vol. 17, No. 2, pp. 75-93.
- Åström, K. J., Wittenmark, B. (1995). *Adaptive control*, 2nd edition, 2nd ed., Reading: Addison-Wesley.
- Ball & Plate Technical Manual (1996). *User's manual of CE151 ball & plate apparatus*, Tecquipment and Humusoft.
- Bartos, F. J. (1997). Artificial intelligence: smart thinking for complex control, *Control Engineering International*, Vol. 44, No. 10, pp. 44-52.
- Bezdek, J. (1994). What is computational intelligence, in J. M. Zurada, R. J.; Marks II, C.; Robinson, J. (eds) *Computational intelligence: imitating life*, IEEE Press, New York, pp. 1-12.
- Bonissone, P. P.; Chen, Y.-T.; Goeble, K.; Khedkar, P. S. (1999). Hybrid soft computing systems: industrial and commercial applications, *Proceedings of the IEEE*, Vol. 87, No. 9, pp. 1641-1667.
- Bordeneuve-Guibe, J.; Vaucoret, C. (2001). Robust multivariable predictive control: an application to an industrial test stand, *IEEE Control Systems Magazine*, Vol. 21, No. 2, pp. 54-65.
- Chen, S.; Cowan, C. F. N.; Grant, P. M. (1991). Orthogonal least squares learning algorithm for radial basis functions networks, *IEEE Transactions on Neural Networks*, Vol. 2, No. 2, pp. 302-309.
- Chipperfield, A.; Fleming, P. (1996). Multiobjective gas turbine engine controller design using genetic algorithms, *IEEE Transactions on Industrial Electronics*, Vol. 43, No. 5, pp. 583-587.
- Chouai, A.; Cabassud, M.; Le Lann, M. V.; Gourdon, C.; Casamatta, G. (2000). Multivariable control of a pulsed liquid-liquid extraction column by neural networks, *Neural Computing & Applications*, Vol. 9, pp. 181-189.
- Coelho, L. S. (2001). Metodologias de inteligência artificial aplicadas ao projeto de sistemas de controle e automação industrial. *Anais do ISA Show Brasil*, São Paulo, SP.
- Coelho, L. S., Coelho, A. A. R. (1999). Algoritmos evolutivos em identificação e controle de processos: uma visão integrada e perspectivas. *Revista SBA Controle & Automação*, São Paulo, SP, Vol. 10, No. 1, pp. 13-30.
- Coelho, L. S., Coelho, A. A. R. (2000). Estudo comparativo de configurações de algoritmos genéticos aplicados à identificação de um processo multivariável, *XIII Congresso Brasileiro de Automática*, Florianópolis, SC, pp. 1197-1202.
- Coelho, L. S.; Coelho, A. A. R.; Krohling, R. A. (2002). Parameters tuning of multivariable controllers based on memetic algorithm: fundamentals and application, *IEEE International Symposium on Intelligent Control*, Vancouver, Canada.
- Craven, P. J.; Sutton, R.; Burns, R. S. (1999). Multivariable neurofuzzy control of an autonomous underwater vehicle, *Integrated Computer-Aided Engineering*, Vol. 6, No. 4, pp. 275-288.

- Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function, *Mathematics of Control Signals and Systems*, Vol. 2, pp. 303-314.
- Garces, F.; K. Warwick; Craddock, R. (1998). Multiple PID mapping using neural networks in a MIMO steam generator system, *Control'98, International Conference on UKACC*, Swansea, UK, Vol. 1, pp. 503-508.
- Ge, S. S.; Wang, C.; Tan, Y. H. (2001). Adaptive control of partially known nonlinear multivariable systems using neural networks, *Proceedings of the IEEE International Symposium on Intelligent Control*, Mexico City, Mexico, pp. 292-297.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading: Addison-Wesley.
- Golea, N.; Golea, A. (2002). Fuzzy adaptive control of multivariable nonlinear systems, *Proceedings of the IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2002*, Vol. 1, Honolulu, HI, USA, pp. 330-334.
- Gomm, J. B.; Yu, D. L. (2000). Selecting radial basis function network centers with recursive orthogonal least squares training, *IEEE Transactions on Neural Networks*, Vol. 11, No. 2, pp. 306-314.
- Haddad, W. M.; Hayakawa, T.; Leonessa, A. (2002). Direct adaptive control for discrete-time nonlinear uncertain dynamical systems, *Proceedings of the American Control Conference*, Anchorage, AK, USA, Vol. 3, pp. 1773-1778.
- Hansen, P. C. (1998). *Rank-Deficient and Discrete Ill-Posed Problems*, Philadelphia, PA, USA: SIAM.
- Ho, W. K.; Lee, T. H.; Han, H. P.; Hong, Y. (2001). Self-tuning IMC-PID control with interval gain and phase margins assignment, *IEEE Transactions on Control Systems Technology*, Vol. 9, No. 3, pp. 535-541.
- Huang, J. -J. (2000). Automatic tuning of the PID controller for servo systems based on relay feedback, *Proceedings of the 26th Annual Conference of the IEEE Industrial Electronics Society, IECON 2000*, Nagoya, Japan, Vol. 2, pp. 1445-1450.
- Huzmezan, M.; Gough, B.; Kovac, S. (2002). Advanced control of batch reactor temperature, *Proceedings of the American Control Conference*, Anchorage, AK, USA Vol. 2, pp. 1156-1161.
- Iwai, Z.; Shimada, Y.; Mizumoto, I.; Deng, M. (1999). Design of multivariable PID controllers on frequency domain based on particle model matching, *Proceedings of the 14th IFAC World Congress*, Beijing, China, pp. 295-300.
- Jafarov, E. M. (2000). Design of sliding mode control for multi-input systems with multiple state delays, *Proceedings of the American Control Conference*, Chicago, IL, USA, Vol. 2, pp. 1139-1143.
- Jamshidi, M.; Krohling, R. A.; Coelho, L. S.; Fleming, P. (2002). *Robust control with genetic algorithms*, Boca Raton, FL: CRC Press.
- Jiang, J. (1994). Optimal gain scheduling controller for a diesel engine, *IEEE Control Systems Magazine*, Vol. 14, No. 4, pp. 42-48.
- Katebi, M. R., Moradi, M. H.; Johnson, M. A. (2000). A comparison of stability and performance robustness of multivariable PID tuning methods, *IFAC Workshop of Digital Control: Past, Present and Future of PID Control*, Terrassa, Spain, pp. 529-534.
- Kawanaka, H.; Yoshikawa, T.; Tsuruoka, S. (2000). Acquisition of fuzzy control rules for a mobile robot using genetic algorithm, *Proceedings of the 6th International Workshop on Advanced Motion Control*, Nagoya, Japan, pp. 507-512.
- Khosla, R.; Dillon, T. (1997). *Engineering intelligent hybrid multi-agent systems*, Kluwer Academic Publishers: Boston.
- King, R. E. (1999). *Computational intelligence in control engineering*. New York: M. Dekker.
- Kraus, T. W.; Myron, T. J. (1984). Self-tuning PID controller uses pattern recognition approach, *Control Engineering International*, Vol. 31, No. 6, pp. 106-111.
- Krishnapura, V. G. & Jutan, A. (2000). A neural adaptive controller, *Chemical Engineering Science*, Vol. 55, pp. 3803-3812.
- Landau, I. D.; Constantinescu, A.; Karimi, A. (2000). Direct controller order reduction by identification in closed loop applied to an active suspension, *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, pp. 4479-4484.
- Levine, W. S. (ed.) (1996). *The control handbook*, CRC Press.
- Lin, W. -M.; Yang, C. -D.; Lin, J. -H.; Tsay, M. -T. (2001). A fault classification method by RBF neural network with OLS learning procedure, *IEEE Transactions on Power Delivery*, Vol. 16, No. 4, pp. 473-477.
- Liu, G. P.; Kadirkamanathan, V. (1999). Multiobjective criteria for neural network structure selection and identification of nonlinear systems using genetic algorithms, *IEE Proc.-Control Theory and Applications*, Vol. 146, No. 5, pp. 373-382.

- Ljung, L. (1999). *System identification: theory for the user*, 2nd ed., Upper Saddle River: Prentice Hall PTR.
- Mattei, M. (2000). When quadratic stability meets PID control, *Proceedings of the 3rd IFAC Symposium on Robust Control Design*, Prague, Czech Republic.
- McLoone, S.; Brown, M. D.; Irwin, G.; Lightbody, A. (1998). A hybrid linear/nonlinear training algorithm for feedforward neural networks, *IEEE Transactions on Neural Networks*, Vol. 9, No. 4, pp. 669-684.
- Moddy, J.; Darken, C. J. (1989). Fast learning in networks of locally-tuned processing units, *Neural Computation*, Vol. 1, No. 2, pp. 281-294.
- Oliveira, L. L., Carrapico, O. L. (1998). Multivariable control increases coke unit, refinery throughput, *Hydrocarbon Processing*, Vol. 77, No. 7.
- Ono, H.; Sonoda, T.; Maekawa, A. (2000). Automatic tuning of PID controllers for MIMO processes, *IFAC Workshop of Digital Control: Past, Present and Future of PID Control*, Terrassa, Spain, pp. 79-84.
- Park, J.; Sandberg, I. W. (1991). Universal approximation using radial-basis-function networks, *Neural Computation*, Vol. 3, pp. 246-257.
- Park, J.; Sandberg, I. W. (1993). Approximation and radial basis-function networks, *Neural Computation*, Vol. 5, pp. 305- 316.
- Passino, K. M.; Özgüner, Ü. (1996). Intelligent control: from theory to application, *IEEE Expert Intelligent Systems & Their Applications*, Vol. 11, No. 2, pp. 28-30.
- Peng, H.; Ozaki, T.; Haggan-Ozaki, V.; Toyoda, Y. (2003). A parameter optimization method for radial basis function type models, *IEEE Transactions on Neural Networks*, Vol. 14, No. 2, pp. 432-438.
- Pham, D. T.; Jin, G. (1995). Genetic algorithm using gradient-like reproduction operator, *IEE Electronics Letters*, Vol. 31, No. 18, pp. 1558-1559.
- Poggio, T.; Girosi, F. (1990). Networks for approximation and learning, *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1481-1497.
- Pomerleau, A.; Hodouin, D.; Desbiens, A.; Gagnon, É. (2000). A survey of grinding circuit control methods: from decentralized PID controllers to multivariable predictive controllers, *Powder Technology*, Vol. 108, pp. 103-115.
- Poncia, G.; Bittanti, S. (2001). Multivariable model predictive control of a thermal power plant with built-in classical regulation, *International Journal of Control*, Vol. 74, No. 11, pp. 1118-1130.
- Powell, M. J. D. (1985). Radial basis functions for multivariable interpolation: a review, *IMA Conference on Algorithms for the Approximation of Functions and Data*, pp. 143-167.
- Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. (1986). Learning internal representations by error propagation, in Rumelhart, D. E.; McClelland, J. (eds.), *Parallel distributed processing: explorations in the microstructure of cognition*, Vol. 1, Cambridge, MA: MIT Press, pp. 318-362.
- Saaj, C. M.; Bandyopadhyay, B. (2002). Output feedback sliding mode control for MIMO discrete time systems, *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain.
- Sana, S.; Rao, V. S. (2001). Robust control of input limited smart structural systems, *IEEE Transactions on Control Systems Technology*, Vol. 9, No. 1, pp. 60-68.
- Seng, T. L.; Khalid, M. B.; Yusof, R. (1999). Tuning of a neuro-fuzzy controller by genetic algorithm, *IEEE Transactions on Systems, Man, and Cybernetics ? Part B: Cybernetics*, Vol. 29, No. 2, pp. 226-236.
- Shanley, A. (1995). Advanced process control, *Chemical Engineering*, Vol. 102, No. 5, pp. 28-31.
- Shoureshi, R. A. (1993). Intelligent control systems: are they for real?, *Transactions of the ASME, Journal of Dynamic Systems, Measurement, and Control*, Vol. 115, pp. 392-401.
- Tan; W.; Marquez, H. J.; Chen, T. (2002). Multivariable robust controller design for a boiler system, *IEEE Transactions on Control Systems Technology*, Vol. 10, No. 5, pp. 735 -742.
- Tanaka, K.; Ikeda, T. (1998). Absolute stability conditions in a fuzzy phase-lead compensation and their extension to MIMO systems, *IEEE Transactions on Industrial Electronics*, Vol. 45, No. 2, pp. 333-340.
- Uykan, Z.; Guzelis, C.; Celebi, M. E.; Koivo, H. N. (2000). Analysis of input-output clustering for determining centers of RBFN, *IEEE Transactions on Neural Networks*, Vol. 11, No. 4, pp. 851-858.
- VanDoren, V. J. (1998). Advanced control software goes beyond PID, *Control Engineering International*, January, pp. 57-60.

- Vázquez, F.; Morilla, F. (2002). Tuning decentralized PID controllers for MIMO systems with decouplers, *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain.
- Wang, Q. -G.; Lee, T. -H.; Zhang, Y. (1999). Multi-loop version of the modified Ziegler-Nichols method, *Proceedings of the 14th IFAC World Congress*, Beijing, China.
- Yen, G. G., Lu, H. (2002). Hierarchical rank density genetic algorithm for radial-basis function neural network design, *Proceedings of the Congress on Evolutionary Computation*, Hawaii, USA, Vol. 1, pp. 25-30.
- Yu, A. G. A.; Orlov, F. (2002). Frequency adaptive control of multivariable plants, *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain.
- Yu, W.; Poznyak, A. S.; Alvarez, J. (1999). Neuro control for multicomponent column, *Proceedings of the 14th IFAC World Congress*, Beijing, China.
- Yu; D. L.; Yu, D. W.; Gomm, J. B. & D. Williams (2002). Model predictive control of a chemical process based on an adaptive neural network, *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain.
- Zadeh, L. A. (1994). Soft computing and fuzzy logic, *IEEE Software*, Vol. 11, No. 6, pp. 48-56.
- Zhang, M.; Yeung, L.; Jiang, Y. (2002). Genetic algorithm for input/output selection in MIMO systems based on controllability and observability indices, *Electronics Letters*, Vol. 38, No. 19, pp. 1150-1151.
- Zhao, H.; Guiver, J.; Sentoni, G. (1998). An identification approach to nonlinear state space model for industrial multivariable model predictive control, *Proceedings of the American Control Conference*, Philadelphia, USA, Vol. 2, pp. 796-800.
- Ziegler, J. G.; Nichols, N. B. (1943). Optimum settings for automatic controllers, *Transactions on ASME*, Vol. 65, pp. 433-444.