



PROPOSTA DE UMA NOVA ESTRATÉGIA EM REDES FIELDBUS USANDO CONTROLE ADAPTATIVO POR MODELO DE REFERÊNCIA

Elenilton Teodoro Domingues

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: elenilton@bol.com.br

Mario Pinotti Junior

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: pinotti@sc.usp.br

Afonso Celso Caldas Pacheco

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: apacheco@sc.usp.br

***Resumo.** Este artigo propõe uma nova estratégia de controle em redes Fieldbus usando controle adaptativo por modelo de referência através de variáveis de estado, para resolver sistemas complexos. A nova malha de controle adaptativo consiste de um conjunto de blocos funcionais padrões existentes, conectados aos novos blocos funcionais propostos de acordo com as especificações da norma Fieldbus Foundation. Este novo algoritmo de controle proposto realiza os cálculos de maneira distribuída entre os dispositivos da rede Fieldbus, implicando em várias vantagens como: a) Perda do supervisor não implica a perda do algoritmo de controle, b) Menor tráfego de dados na rede, c) Algoritmo de controle não depende do tempo de processamento do computador.*

***Palavras chaves:** Fieldbus, blocos funcionais, controle adaptativo por modelo de referência.*

1. INTRODUÇÃO

O constante avanço tecnológico na indústria mundial vem tornando as plantas industriais unidades cada vez mais complexas, altamente integradas e que normalmente exigem o controle simultâneo de mais de uma variável do processo. Como consequência disso, pode-se encontrar mais de uma malha de controle para controlar um determinado processo (Valdman, 1998). Sistemas de controle com este aspecto são denominados sistemas de controle multivariáveis, possuindo a característica de ter múltiplas entradas e múltiplas saídas, sendo essas geralmente inter-relacionadas de maneira complicada. A ferramenta mais adequada para solucionar os problemas relacionados a estes tipos de sistemas de controle é a teoria de controle moderna que trabalha diretamente com a descrição de variáveis de estado do sistema, utiliza a notação vetorial matricial para representar sistemas dinâmicos e realiza toda a análise e projeto no domínio do tempo (Ogata, 1993). No entanto, tal técnica de controle, englobando, entre outras, técnicas como: adaptativas e preditivas exige um requisito indispensável para sua implementação: o processamento digital. Com o surgimento da tecnologia *Fieldbus* na década 1980, e o seu grande avanço na questão da padronização na década seguinte, foi possível então obter um sistema de comunicação puramente digital que suportasse tais técnicas de controle digital. No entanto, pouco esforço tem sido realizado para contribuir neste sentido. Atualmente, a maioria dos algoritmos avançados são implementados e executados no supervisor do computador central (Dultra e Veloni, 1996; Ferraz, 1999 e Kuroda et

al. 2001). Nestes sistemas, o *Fieldbus* é geralmente utilizado como uma fonte de informação de dados ao algoritmo de controle. Ferraz (1999) utilizou os dados em um algoritmo adaptativo para melhorar o controle e os resultados de processos não-lineares. Kuroda et al (2001) reduziram os defeitos no posicionamento de válvulas de controle para mudanças na distribuição de velocidade do vapor.

Algo mais significativo a esse respeito, foi colocado por Thiele (2000) que apresenta resumidamente os benefícios e as mudanças de otimização através da implementação de um algoritmo de controle de modelo preditivo como um bloco funcional em nível de controlador.

Visando suprir esta carência de tecnologia, este artigo tem por objetivo propor uma nova estratégia de controle adaptativa por modelo de referência em redes *Fieldbus* através de variáveis de estado, para resolver problemas complexos em sistemas de controle de múltiplas entradas e múltiplas saídas (MIMO). A nova malha de controle adaptativo em *Fieldbus* consistirá de um conjunto de blocos funcionais padrões existentes conectados a um conjunto de novos blocos funcionais propostos de acordo com as especificações da norma *Fieldbus Foundation* (1999).

2. REDES *FIELDBUS*

Fieldbus é uma rede constituída por um sistema de comunicação de dados digital, serial, bidirecional, multiponto, associado a protocolos de enlace e de interface com o usuário, capaz de promover a comunicação dos instrumentos de campo entre si, bem como os equipamentos hierarquicamente superiores, aproveitando ao máximo as potencialidades da geração de instrumentos de campo inteligentes.

Isto proporcionou a descentralização das tarefas onde os algoritmos de controle migram para instrumentos de campo, de forma que as malhas de controle passam a ser executadas no campo e de modo verdadeiramente distribuído entre os vários transmissores, como pode ser visto na Fig. (1).

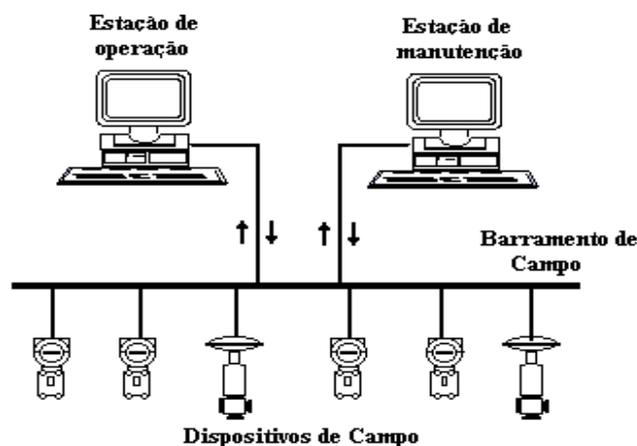


Figura 1. Sistema digital de controle *Fieldbus*

Algumas vantagens desta tecnologia são citadas a seguir:

- 1- Redução nos custos de fiação, instalação, operação e manutenção;
- 2- Maior acuidade nas medições, com diminuição das conversões A/D e D/A;
- 3- Acesso a múltiplas variáveis;
- 4- Configuração e diagnóstico remoto;
- 5- Interoperabilidade e intercambiabilidade;
- 6- Sistema aberto no que se refere a hardware e software;
- 7- Facilidade de manutenção e implementação de novas estratégias, sem alteração no arranjo físico da instalação.

A tecnologia *Fieldbus* proporciona uma produtividade com qualidade inovadora e de melhor desempenho que as tecnologias atuais.

3. BLOCOS FUNCIONAIS

Os blocos funcionais representam funções básicas de automação, desempenhadas nos dispositivos de campo, como: aquisição, controle, atuação, monitoração e alarmes. Ou seja, são modelos de aplicação que fornecem informações para a interface de operação, diagnósticos e a configuração de funções como; entradas, saídas e parâmetros internos, assim como pode ser visto na Fig. (2).

Cada bloco funcional processa parâmetros de entrada de acordo com um algoritmo específico e um conjunto interno de parâmetros, produzindo parâmetros válidos para uso no mesmo bloco funcional ou em outro bloco funcional da aplicação (*Fieldbus Foundation*, 1999).

Na configuração da estratégia de controle é especificada a escolha do bloco funcional adequado a ser utilizado e em qual dispositivo este será executado. A interligação dos blocos funcionais define a estratégia de controle e a programação do processo a ser controlado.

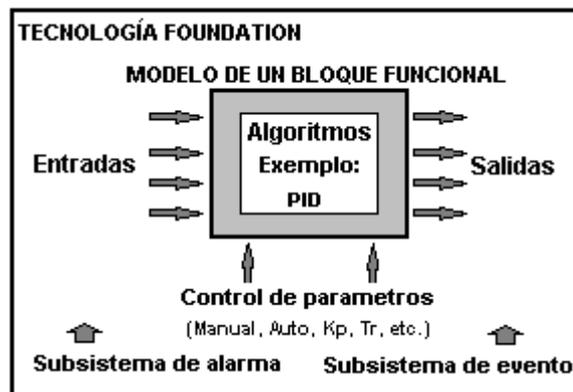


Figura 2. Diagrama geral de um bloco funcional

3.1 Limitações dos Blocos Funcionais na Aplicação das Técnicas de Controle

Atualmente as estratégias de controle em redes *Fieldbus* realizadas por blocos funcionais executam medições da variável de interesse: decidem se seu valor é aceitável e aplicam um esforço corretivo se necessário. Essas estratégias tomam, geralmente, suas decisões de acordo com o algoritmo PID (Proporcional, Integral e Derivativo) que utiliza um único sensor para a medida e um único atuador para a ação corretiva. Esta rotina de controle de única variável trabalha muito bem para uma ampla variedade de problemas de controle comutando as variáveis de processo, que podem ser manipuladas independentemente. Quando o sistema de controle é requerido para conseguir de uma só vez os objetivos que usam os atuadores múltiplos que afetam todas as variáveis do processo simultaneamente, o problema torna-se muito complicado devido ao fato de não existir nenhuma ferramenta, ou melhor, nenhum bloco funcional, que trate deste problema.

3.2 Projeto de Novos Blocos Funcionais

A implementação de novos blocos funcionais deve seguir a metodologia sugerida pela norma *Foudation*. A seguir são apresentados os passos necessários para o desenvolvimento e implementação dos novos blocos de acordo com a norma *Fieldbus Foundation* - FF890 (1999).

Passo 1: Inicialmente deve-se modelar o funcionamento do bloco, e a partir deste modelo, identificar as variáveis que serão acessadas por procedimentos de fábrica e aquelas que serão acessadas pelo usuário com o objetivo de diagnóstico, configuração e operação. Pode-se então montar uma lista de variáveis com estas características.

Passo 2: A lista de variáveis deve ser dividida em três partes de acordo com o seu uso:

- Variáveis que descrevem o recurso, como espaço de memória e tempo mínimo de execução. Estas variáveis não são editadas pelo usuário;

- Variáveis configuradas ou verificadas apenas uma vez nas fábricas ou no lugar da instalação. Exemplo: unidade de engenharia, *offset* e *span*.

- Variáveis verificadas periodicamente pelo usuário por procedimentos de manutenção ou operação. Exemplo: alarmes, *setpoints*, sintonias e modo de operação.

Passo 3: As variáveis descrevendo os recursos são parâmetros para o bloco *resource* do transmissor, além daqueles parâmetros do bloco *resource* definidos pela norma. As variáveis definidas como sendo de configuração de fábrica ou de uso na instalação do sistema são parâmetros a serem adicionados ao bloco *transducer*.

Passo 4: Os parâmetros identificados como sendo de uso periódico, em operação ou em manutenção são os parâmetros do futuro bloco funcional.

A norma recomenda que os parâmetros dos novos blocos sejam comparados como os parâmetros dos blocos padrões, em nível de funcionalidade. Caso seja identificada alguma semelhança nos parâmetros do bloco novo com os parâmetros de um bloco padrão, então este deveria servir como base para o projeto, assim bastará adicionar os parâmetros inéditos no bloco padrão. Estes parâmetros são variáveis que devem ser mapeadas como variáveis simples, vetores ou *records*, de acordo como os tipos de dados padrões e então listados em seguida aos parâmetros padrões na definição do bloco. Os parâmetros de um bloco não padrão são classificados de acordo com a regra acima em parâmetros de *device* normalizados e parâmetros específicos do fabricante. Caso nenhum bloco padrão sirva de modelo para o novo bloco, os parâmetros deste, caso seja possível, deverão ter as mesmas características dos parâmetros padrões, como nome, função, status e tipo de operação. A *Fieldbus Foundation* prevê a possibilidade de normalização tanto de parâmetros como de blocos específicos de fabricante caso estes tenham utilidade abrangente. Na implementação computacional do bloco deve-se criar estruturas que disponibilizem para a camada de aplicação os serviços de inicialização, execução, leitura e escrita para cada parâmetro disponível. Após esta implementação, faz-se a integração destas estruturas, ou métodos, nas estruturas complementares ao *Function block application process* (FBAP): *Device Description* (DD) e ao *Object Dictionary* (OD). A DD do transmissor deve incluir a descrição do bloco e dos seus parâmetros. Na caracterização dos parâmetros tem-se: tipo de dado, permissão de acesso, *label*, texto de *help* e a classe dos parâmetros (estático, de entrada ou de saída). Já a descrição do bloco na DD inclui o fabricante, a revisão, todos os parâmetros, as *views* e os grupos de unidades relacionadas. Quando compilada, a DD fornece dois números inteiros denominados *Member ID* e *Item ID* para cada parâmetro, onde os quais são utilizados na inclusão dos parâmetros do bloco no OD, juntamente com outras informações que caracterizam o parâmetro, entre as quais são mostradas adiante nas Tabelas de Parametrização dos blocos. Informações sobre o bloco devem estar presente também no arquivo *Capability file*, utilizado pelo configurador juntamente com o arquivo de DD para uma correta identificação e configuração de cada diferente transmissor. No arquivo de *Capability* do transmissor estão as informações de tempo de execução do bloco e quantidade de memória RAM necessárias para armazenar todos os seus parâmetros, assim como o *Item ID* do bloco e o número de parâmetros deste.

4. NOVOS BLOCOS FUNCIONAIS

4.1. Bloco Funcional Modelo de Referência (MR)

O projeto do bloco Modelo de Referência (MR) de acordo com os requisitos obrigatórios da norma *Fieldbus Foundation* tem como objetivo especificar o desempenho de um sistema através de um modelo referencia que produzirá a saída desejada para uma determinada entrada.

O bloco foi estruturado para representar tanto modelos de sistemas de uma única entrada e uma única saída (SISO) como modelos de sistemas de múltiplas entradas e múltiplas saídas (MIMO), na qual pode conter um vetor de estado com até seis variáveis de estado para descrever o comportamento dinâmico do sistema.

4.1.1. Estruturas e Funções do Bloco MR

O algoritmo deste bloco funcional é baseado em uma equação diferencial de primeira ordem (equação de estado) dada pela Eq. (1) e por uma equação de algébrica (equação de saída) dada pela Eq.(2).

$$\dot{x}_m = A_m x_m + B_m r \quad (1)$$

$$y_m = C_m x_m \quad (2)$$

Uma representação esquemática do bloco, com seu algoritmo, entradas e saídas é apresentada a seguir na Fig. (3).

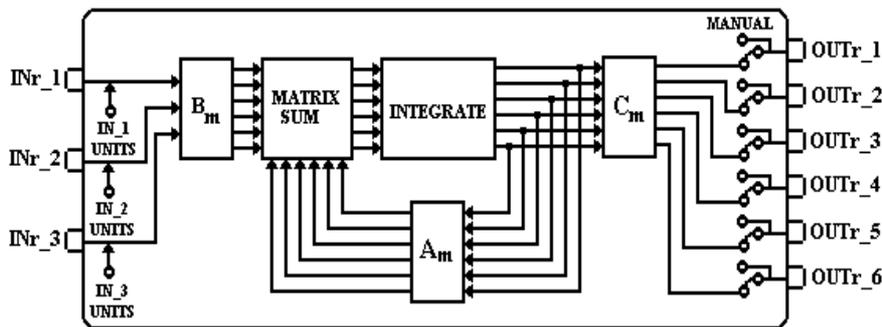


Figura 3. Bloco funcional modelo referência

Ao usuário cabe configurar, através dos parâmetros de configuração disponíveis, as matrizes do modelo de referência (A_m , B_m e C_m) e as unidades das grandezas de entrada.

Com relação aos modos de operação, o bloco suporta três. Sendo esses:

- (1) Modo OOS (*Out of Service*): o bloco fica “desligado”;
- (2) Modo Auto (Automático): a saída desejada do modelo de referência é obtida através do algoritmo definido pelo bloco;
- (3) Modo Man (Manual): as saídas desejadas são diretamente definidas pelo usuário, sendo que o *default* destas saídas em modo manual é zero.

As condições de execução do algoritmo não mudam o modo do bloco, nem com as entradas em status *Bad*. Caso ocorra uma operação indefinida, a operação do bloco não é abortada.

No caso de perda de energia no transmissor, todos os parâmetros de configuração são armazenados em memória não volátil, garantindo assim uma correta reinicialização da operação normal do bloco assim que o transmissor é novamente ligado, sem a necessidade de reconfiguração.

O sistema de alarme do Bloco MR segue os requisitos básicos da *Foundation*, portanto o bloco pode ser perfeitamente integrado a qualquer tipo de malha de controle.

4.2. Bloco Funcional Estimador de Estado (EE)

O bloco Estimador de Estado (EE) foi desenvolvido de forma que todos os seus parâmetros definidos possam ser suportados pelos dispositivos interoperáveis registrados na *Fieldbus Foundation*.

O projeto deste bloco EE é de ordem plena e foi baseado na teoria de controle moderna com a finalidade de reconstituir os estados de uma planta através das medidas das variáveis de saída e de controle. Este bloco também foi estruturado com a opção de utilizar o vetor de estado estimado na realimentação de estado para gerar o vetor de controle desejado.

4.2.1. Estruturas e Funções do Bloco EE

O algoritmo deste bloco possui um sistema de equações dinâmicas matriciais, Eq. (3) e Eq. (4), que são executadas em *loop* num processo de convergência. Enquanto que a lei de controle é definida pela Eq. (5).

$$\dot{\hat{x}} = A_p \hat{x}_p + B_p u + L (y - \hat{y}) \quad (3)$$

$$\hat{y} = C_p \hat{x}_p \quad (4)$$

$$z = -K \hat{x}_p \quad (5)$$

Uma representação esquemática do bloco EE, com seu algoritmo, entradas e saídas é apresentada a seguir na Fig.(4):

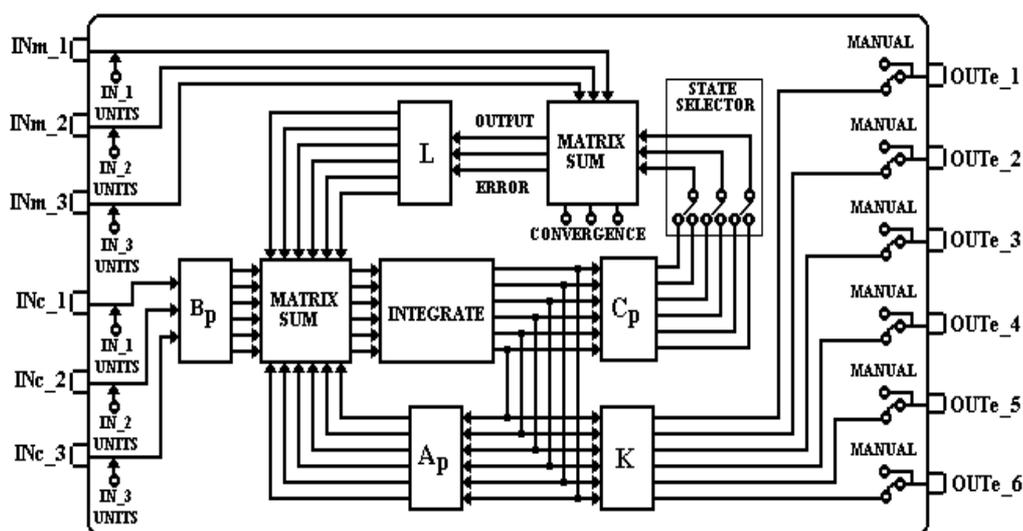


Figura 4. Bloco funcional estimador de estados

Ao usuário cabe configurar, através dos parâmetros de configuração disponíveis, as matrizes da planta (A_p , B_p e C_p), a matriz do estimador (L), as unidades das grandezas de entrada e os ganhos (K) no caso do vetor de estado estimado ser usado na realimentação de estado para gerar o vetor de controle desejado. Existem ainda parâmetros de visualização de operação do bloco como: `BLOCK_ERR`, `CONVERGENCE` e `OUTPUT_ERR`.

O bloco possui seis entradas e seis saídas. As três primeiras entradas são valores medidos da planta real e as outras três são sinais de controle recebidos de outros blocos.

As saídas deste bloco são os valores dos estados estimados da planta multiplicados pelos ganhos de realimentação ($K_{1...6}$). O valor *default* deste parâmetro $K_{1...6}$ é 1. No caso da estratégia de controle utilizar o bloco EE somente para estimar estados, com o valor do *default* de 1, as saídas do blocos corresponderam conseqüentemente os estados estimados da planta.

Os modos de operação e as condições de execução do algoritmo deste bloco são iguais ao do o bloco MR, assim como citado anteriormente.

Com relação aos *status* das entradas, estes não afetam a execução do algoritmo, sendo que os *status* de X_{p1} , X_{p2} , X_{p3} , X_{p4} , X_{p5} e X_{p6} são semelhante ao pior *status* entre as entradas. Este tipo de implementação garante o mecanismo da norma *Foundatrion* de propagação de *status* na malha, como ferramenta para garantia de segurança do sistema em caso de falhas ou emergência.

Em operação Man ou Auto, o bloco deve recuperar os valores de Pre_X_{p1} , Pre_X_{p2} , Pre_X_{p3} , Pre_X_{p4} , Pre_X_{p5} e Pre_X_{p6} da execução anterior para serem utilizados na execução seguinte.

O parâmetro CONVERGENCE indica se o sistema de equações do bloco esta respondendo da mesma forma que a planta, ou seja, se a saída do modelo convergiu ou não. Para determinar esta convergência foi utilizado uma margem de erro de 5%.

O sistema de alarme do Bloco MR também segue os requisitos básicos da *Foundation*, portanto o bloco pode ser perfeitamente integrado a qualquer tipo de malha de controle.

4.3. Bloco Funcional Adaptativo por Modelo de Referência (AMR)

O bloco funcional Adaptativo por Modelo de Referência (AMR) foi projetado dentro dos padrões da norma *Fieldbus Foundation*, baseado no algoritmo de controle adaptativo direto, que utiliza variáveis de estado para descrever o conjunto de equações do sistema adaptativo por modelo de referência.

O bloco AMR tem como objetivo encontrar a lei de controle adaptativa de realimentação que muda a estrutura e a dinâmica da planta de forma que suas propriedades são exatamente as mesmas que as de um modelo de referência, ou seja, o desempenho desejado da planta é expresso em termos de um modelo de referência que fornece a resposta desejada para um determinado sinal de comando.

4.3.1. Estruturas e Funções do Bloco AMR

O algoritmo deste bloco possui um conjunto de equações matriciais que são dadas pela Eq.(6) e Eq. (8) e executadas em *loop* para obter a lei de controle adaptativa. A Eq.(6) representa a lei de controle adaptativa, a Eq. (7) e a Eq. (8) são os termos que compõem esta lei e a Eq.(9) é a equação do erro na qual é definida pela diferença entre os estados estimados da planta real e os estados do modelo referência (Narendra e Annaswamy,1989).

$$u = -\Phi \Theta \hat{x}_p + \Phi r \quad (6)$$

$$\dot{\Theta} = -B_m^T P e \hat{x}_p^T \quad (7)$$

$$\dot{\Phi} = -\Phi B_m^T P e u^t \Phi \quad (8)$$

$$e = \hat{x}_p - x_m \quad (9)$$

Ao usuário cabe configurar, através dos parâmetros de configuração disponíveis, as matrizes (P e B_m^T), as condições iniciais das integrais e as unidades das grandezas de entrada. P é uma matriz de ordem $n \times n$, obtida através da equação *Lyapunov*:

$$PA_m + A_m^T P = -Q \quad (10)$$

O bloco possui quinze entradas e três saídas. As entradas correspondem aos pontos de operação do sistema (*setpoint*), aos estados obtidos do modelo de referência e aos estados estimados da planta, respectivamente. As três saídas compõem a lei de controle adaptativa.

Os modos de operação e as condições de execução do algoritmo deste bloco são iguais ao do o bloco MR e EE, assim como citado anteriormente.

O sistema de alarme do Bloco MR também segue os requisitos básicos da *Foundation*, portanto o bloco pode ser perfeitamente integrado a qualquer tipo de malha de controle.

Uma representação esquemática do bloco AMR, com seu algoritmo, entradas e saídas é apresentada a seguir pela Fig.(5):

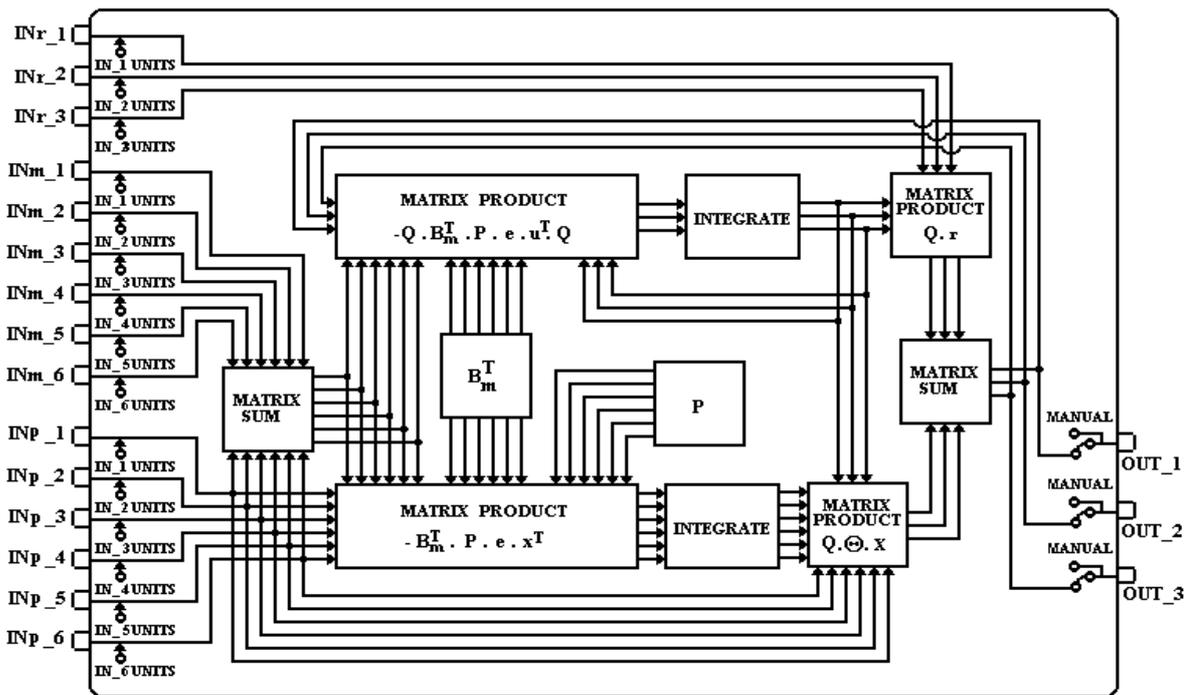


Figura 5. Bloco funcional adaptativo por modelo de referência

5. MALHAS DE CONTROLE NO *FIELDBUS*

As malhas de controle no *Fieldbus* consistem no interligamento lógico de diversos blocos funcionais implementados nos dispositivos (*devices*) da rede através de um *software* configurador. A conexão entre os blocos é definida como um *link* da rede. A Fig.(6) mostra um simples exemplo de malha de controle com os respectivos *links* interligando os blocos funcionais.

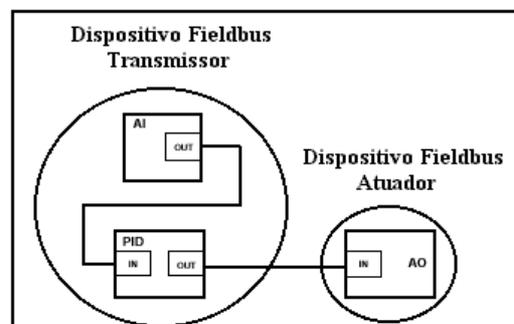


Figura 6. Exemplo de uma malha de controle

5.1. Nova malha de controle com realimentação de estado

A malha de controle por realimentação de estado consiste: O sinal de referência (*Setpoint*) é gerado pelo bloco funcional “Constante”(CT). O bloco Estimador de Estados (EE) recebe as variáveis de entrada e a de saída do sistema real através do bloco de Entrada Analógica (AI) e do bloco Saída Analógica (AO), respectivamente. Esse bloco estima os estados da planta real, multiplica pelos ganhos de realimentação e fornece os resultados ao bloco Aritmético (ARTH).

O bloco Aritmético (ARTH) tem a função de somar os estados multiplicados pelos ganhos de realimentação e comparar com o valor de referência fornecido pelo bloco constante (CT), determinando o erro. Este bloco envia o erro ao bloco saída analógica (AO) para fazer as correções necessárias e atuar na planta real. O diagrama desta estratégia é mostrado a seguir pela Fig.(7).

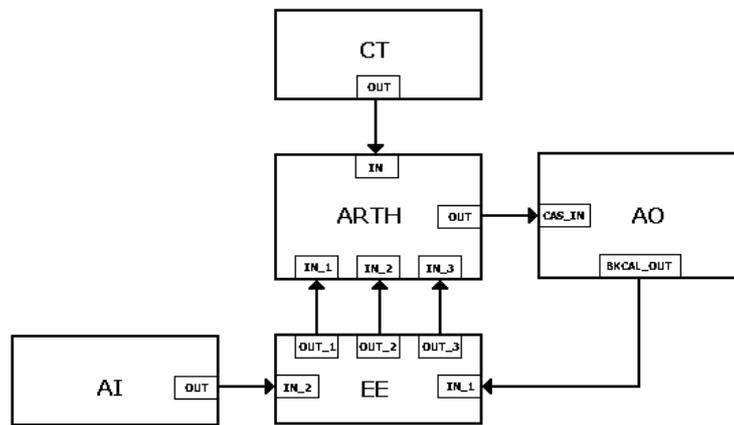


Figura 7. Malha de controle com realimentação de estado

5.2 Malha de controle adaptativo por modelo de referência

A malha de controle adaptativa por modelo de referência consiste nos seguintes blocos: O bloco modelo referência que o modelo de referência do sistema através de equações de estado, recebe o valor desejado de operação (*setpoint*) do bloco constante e produz a saída desejada. O bloco estimador de estado, que recebe os valores das variáveis de entrada e saída do sistema real através dos blocos entrada analógica e saída analógica, estima os estados e fornece a saída estimada da planta. O bloco controle adaptativo por modelo de referência, que recebe a saída desejada do sistema do bloco modelo referência, a saída estimada da planta do bloco estimador de estado e o *setpoint* do bloco constante, compara as saídas (desejada e estimada) determinando o erro e, através de um mecanismo de adaptação, produz a correção e a envia ao bloco ao para atuar na planta real. Na Fig.(8) é mostrada a nova malha de controle proposta

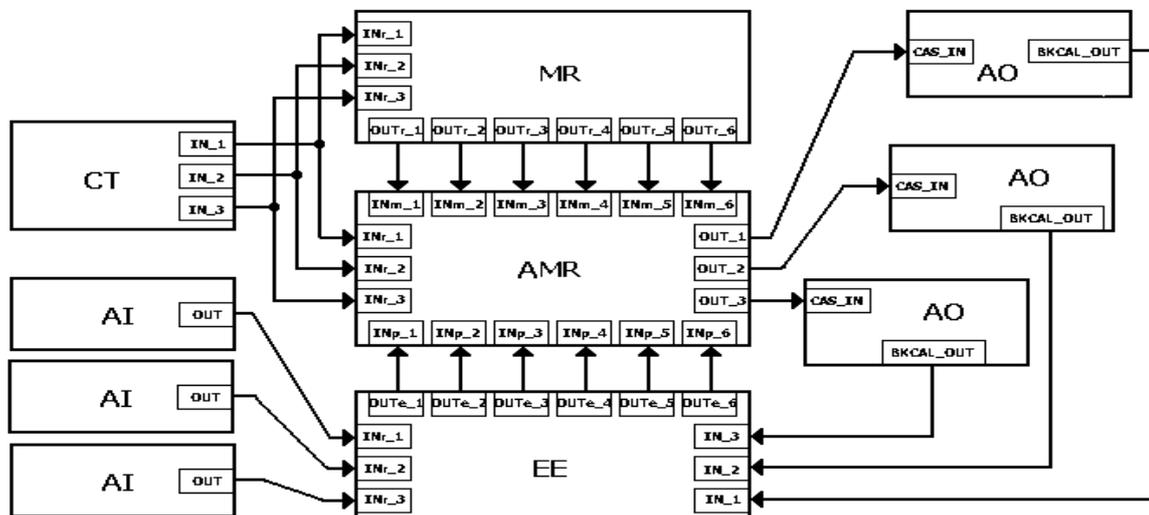


Figura 8. Malha de controle adaptativa por modelo de referência

6. CONCLUSÃO

Foi apresentada neste artigo uma nova estratégia de controle adaptativa por modelo de referência em redes Fieldbus. Esta nova estratégia traz uma maior facilidade em controlar plantas com múltiplas entradas e múltiplas saídas em *Fieldbus*, devido ao fato de trabalhar com variáveis de estado. Isso traz uma contribuição na área do desenvolvimento de controle de processos contínuos com processamento distribuído em rede; caracterizando como uma aplicação essencialmente do protocolo *Fieldbus Foundation*.

Neste aspecto, aplicar técnicas de controle moderno, usando a tecnologia *Fieldbus*, representa muito mais do que apenas uma solução alternativa aos sistemas de controles atuais. Optando por esta nova solução, várias vantagens podem ser obtidas como: 1) Sistema de controle mais eficiente; 2) Implementação de sistemas de segurança mais elaborados; 3) Controle com mais de uma variável simultaneamente; 4) Integração do sistema de controle, monitoração, segurança, manutenção e supervisão da planta; 5) Perda do supervisor não implica a perda do algoritmo de controle; 6) Menor tráfego de dados na rede, 7) Algoritmo de controle não depende do tempo do processamento do computador.

7. REFERÊNCIAS

- Dultra, M.L. e Veloni, V., 1996, "Boiler control with Fieldbus", FieldComms UK. The Industrial Networking Show and Conference. GGH Marketing Commun, Titchfield, UK; pp. 236, p.85-90.
- Ferraz, W.D., 1999, "Adaptive Control Using Fieldbus Resources" In: Conferencia Internacional ISA SHOW 99.
- Fieldbus Foundation., 1999, "Foundation Specification. Function block Application Process –Parte- 1, 2 e 3, F-890-1.4, FF-891-1.4 e FF-892-1.4, Final Specification".
- Kuroda, M., Kaseda, C. e Hidaka, T., 2001, "Control Valve Diagnostic Using Fieldbus", In: Conferencia Internacional ISA SHOW 2000.
- Narendra, K.S. e Annaswamy A.M., "Stable adaptive systems", Ed Prentice-Hall, 1989.
- Ogata, K., 1993, "Engenharia de controle moderno", Ed Prentice-Hall, Rio de Janeiro, Brazil, 47p.
- Thiele, D., 2000, "Benefits and challenges of implementing model predictive control as a function block", In: Conferencia Internacional ISA SHOW 2000.
- Valdman, B., 1998, "Dinâmica e Controle de Processos", Ed. Tórculo artes Gráficas, S.A.L, Rosalía de Castro 45, Santiago, Chile, 172p.

PROPOSAL OF A NEW STRATEGY IN FIELDBUS NETWORKS USING MODEL REFERENCE ADAPTIVE CONTROL

Elenilton Teodoro Domingues

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: elenilton@bol.com.br

Mario Pinotti Junior

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: pinotti@sc.usp.br

Afonso Celso Caldas Pacheco

Universidade de São Paulo (USP) - Departamento de Engenharia Mecânica, Caixa postal 359
Fone: (016) 273-9435, Cep: 13560-250, São Carlos – SP, Brasil, E-mail: apacheco@sc.usp.br

Abstract. *This paper presents a new strategy of control in Fieldbus networks using model reference adaptive control through state variables to perform control of complex systems. The new adaptive control strategy consists of a set of standards existing function block, connected to the new function blocks proposed in accordance with the specifications of Fieldbus Foundation. This new algorithm of considered control carries the calculations in a distributed way among the devices of the Fieldbus network, implying in some advantages as: Loss of the supervisory does not imply on the loss of control algorithm, b) Less traffic of data in the network, c) Control algorithm does not depend on the processing time of the computer.*

Keywords: *Fieldbus, function block, model reference adaptive control.*