

SINTONIZAÇÃO ÓTIMA DE CONTROLADORES PID DIGITAIS

Elder M. Hemerly

Instituto Tecnológico de Aeronáutica
CTA - ITA - IEEE
12225 - São José dos Campos - SP

Resumo - Uma técnica de projeto para otimizar os parâmetros de controladores digitais do tipo Proporcional-Integral-Derivativo, dado o modelo discreto do sistema monovariável a controlar, é apresentada. Esta técnica envolve a minimização no tempo de uma função custo que incorpora o desvio da saída do processo em relação a um modelo de referência. Esta abordagem substitui os procedimentos heurísticos usualmente empregados para sintonizar os parâmetros do controlador. Um ambiente integrado para a implementação da técnica de otimização em microcomputadores é descrito. Este ambiente inclui a possibilidade de se efetuar controle em tempo real, permitindo a validação do projeto. Dois exemplos representativos são apresentados.

Abstract - An integrated environment for optimally tuning digital Proportional-Integral-Derivative controllers is proposed. A time domain optimization technique is employed to minimize a cost function involving the error between the desired step response and the actual response. This technique, when compared to usual ones like the minimization of the ISE criterion, seems to tackle more naturally performance requirements and engineering constraints issues. The integrated environment allows the user to apply real time control to the controlled process and compare the expected performance with that actually obtained. Two representative examples are presented for exhibiting the effectiveness of the optimization technique and the usefulness of the integrated environment.

1. INTRODUÇÃO

Por aproximadamente 50 anos o problema da sintonização de controladores tipo Proporcional-Integral-Derivativo tem motivado trabalhos teóricos e experimentais. Do trabalho pioneiro de Ziegler e Nichols (Ziegler e Nichols, 1942) até abordagens recentes, como aquelas baseadas em Inteligência Artificial (como em Anderson, Blankenship e Lebow, 1988), o objetivo básico é estabelecer regras para sintonizar os ganhos desse controlador, que sejam, simultaneamente, de implementação simples e eficiente para diversos tipos de processos.

A maioria das técnicas para se sintonizar controladores PID são heurísticas, como por exemplo em (Ziegler e Nichols, 1942; Shinsky, 1983; Anderson, Blankenship e Lebow, 1988) e diversas outras referências não mencionadas. Conseqüentemente, os parâmetros do controlador não são ótimos no sentido matemático formal. Com esta assertiva não se pretende negar a eventual eficiência dessas técnicas heurísticas, mas sim enfatizar que um melhor desempenho do sistema em malha fechada pode ser obtido

utilizando-se técnicas de otimização para sintonizar os parâmetros do controlador.

Alguns trabalhos anteriores já abordaram o problema da sintonização ótima de parâmetros, mas com algumas deficiências. Em trabalho recente (Al-Assadi e Al-Chalabi, 1987), por exemplo, propuseram uma técnica envolvendo a minimização do critério ISE. Contudo, conforme ressaltado por (Fu, Olbrot e Polis, 1989), a robustez do sistema de controle resultante pode ser insatisfatória. Adicionalmente, restrições de engenharia, tal como energia do controle, não são consideradas.

Neste trabalho apresentaremos uma técnica baseada em otimização no tempo para sintonizar controladores digitais do tipo PID, com dois objetivos básicos: proporcionar uma alternativa eficiente para as técnicas heurísticas e remover algumas deficiências das técnicas baseadas em otimização existentes, como aquela apresentada em (Al-Assadi e Al-Chalabi, 1987). Adicionalmente, um ambiente integrado para a implementação da técnica será descrito. O *software* correspondente foi programado na linguagem C e pode ser executado em microcomputadores do tipo IBM e compatíveis.

Inicialmente apresentaremos a técnica de otimização para sintonizar os parâmetros de controladores do tipo PID digitais. Em seguida descreveremos o ambiente integrado para gerenciar a implementação desta técnica. Por fim, dois processos representativos serão apresentados e simulados para mostrar que uma ação de controle mais efetiva pode ser obtida com a técnica proposta.

2. OTIMIZAÇÃO DE CONTROLADORES PID DIGITAIS

Como estamos interessados em sintonizar controladores PID digitais, é suposto que o sistema dinâmico contínuo a ser controlado pode ser descrito pelo seguinte modelo discreto

$$\begin{aligned} y(k) &= a_1 y(k-1) + a_2 y(k-2) + \dots + a_p y(k-p) + \\ &+ b_1 u(k-d) + b_2 u(k-d-1) + \dots + \\ &+ b_q u(k-d-q+1) \end{aligned} \quad (1)$$

ou ainda pela função de transferência discreta

$$G(z) = z^{-d} \cdot \frac{b_1 + b_2 z^{-1} + \dots + b_{q-1} z^{-q+2} + b_q z^{-q+1}}{1 - a_1 z^{-1} - a_2 z^{-2} - \dots - a_{p-1} z^{-p+1} - a_p z^{-p}} \quad (2)$$

com $d \geq 1$, onde d é o atraso de transporte do processo.

Para controlar o sistema (1), um controlador PID digital será empregado. Há diversas variantes do controlador PID digital básico. Em (Isermann, 1981) essas diversas formas são mostradas. Sem perda de generalidade, utilizaremos um PID digital com estrutura

$$u(k) = u(k-1) + k_p \{e(k) - e(k-1)\} + k_i e(k) + k_d \{e(k) - 2e(k-1) + e(k-2)\} \quad (3)$$

e o problema básico é sintonizar o controlador, isto é, determinar valores de k_p , k_i e k_d que assegurem bom desempenho do sistema de controle correspondente.

A abordagem ótima usualmente empregada para selecionar valores convenientes de k_p , k_i e k_d consiste na minimização das seguintes funções custo:

- i) ISE, $J(k_p, k_i, k_d) = \sum_{k=0}^n e^2(k)$,
- ii) IAE $J(k_p, k_i, k_d) = \sum_{k=0}^n |e(k)|$;

com $e(k) = y_{ref}(k) - y(k)$, onde $y_{ref}(k)$ é a referência, usualmente assumida ser do tipo degrau. A principal deficiência desta abordagem está no fato de que restrições de engenharia, tais como energia e taxa de variação do controle, não são explicitamente incluídas na função custo. Assim, o controlador resultante pode ser irrealista e inútil. Uma maneira convencional de se eliminar esta deficiência consiste na introdução do sinal de controle na função custo, isto é, os critérios apresentados acima são substituídos por,

$$J(k_p, k_i, k_d) = \sum_{k=0}^n (e^2(k) + \rho(\Delta u(k))^2) \text{ e}$$

$$J(k_p, k_i, k_d) = \sum_{k=0}^n (|e(k)| + \rho(\Delta u(k))^2) ,$$

respectivamente, onde $\Delta u(k) = u(k) - u(k-1)$. Infelizmente, não há modo de se estabelecer, *a priori*, qual é a influência da

ponderação do controle ρ na sobre-elevação e tempo de subida do sistema de controle resultante do processo de otimização.

Neste trabalho utilizaremos uma abordagem mais intuitiva e sensata para sintonizar, de modo ótimo, os parâmetros (k_p , k_i , k_d) do controlador. Ao invés de considerarmos o erro $e(k)$ entre a referência $y_{ref}(k)$ e a saída $y(k)$ do sistema controlado, consideraremos o erro entre $y(k)$ e a saída $y_m(k)$ de um modelo de referência. Esta abordagem encontra-se resumida na Figura 2.1, onde os valores ótimos dos parâmetros são representados por (k_p^o , k_i^o , k_d^o).

O desempenho desejado pelo projetista é especificado por um modelo de referência de segunda ordem, caracterizado pelo fator de amortecimento ξ e pela frequência natural ω_n . Mais especificamente, o projetista desejaria que o sistema controlado se comportasse como um sistema contínuo de segunda ordem descrito pela função de transferência

$$T_m(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (4)$$

Por simplicidade, ξ e ω_n são especificados no domínio s , e não no domínio z . Contudo, se (4) for amostrada com período de amostragem igual a T segundos, podemos calcular exatamente a função de transferência discreta correspondente $T_m(z)$, tal como em (Åström e Wittenmark, 1984).

Assim

$$T_m(z) = \frac{Y_m(z)}{Y_{ref}(z)} = \frac{\beta_1 z^{-1} + \beta_2 z^{-2}}{1 - \alpha_1 z^{-1} - \alpha_2 z^{-2}} \quad (5)$$

onde:

$$\alpha_1 = 2e^{-\xi\omega_n T} \cos(\sqrt{1-\xi^2}\omega_n T), \quad \alpha_2 = -e^{-2\xi\omega_n T} \quad (6)$$

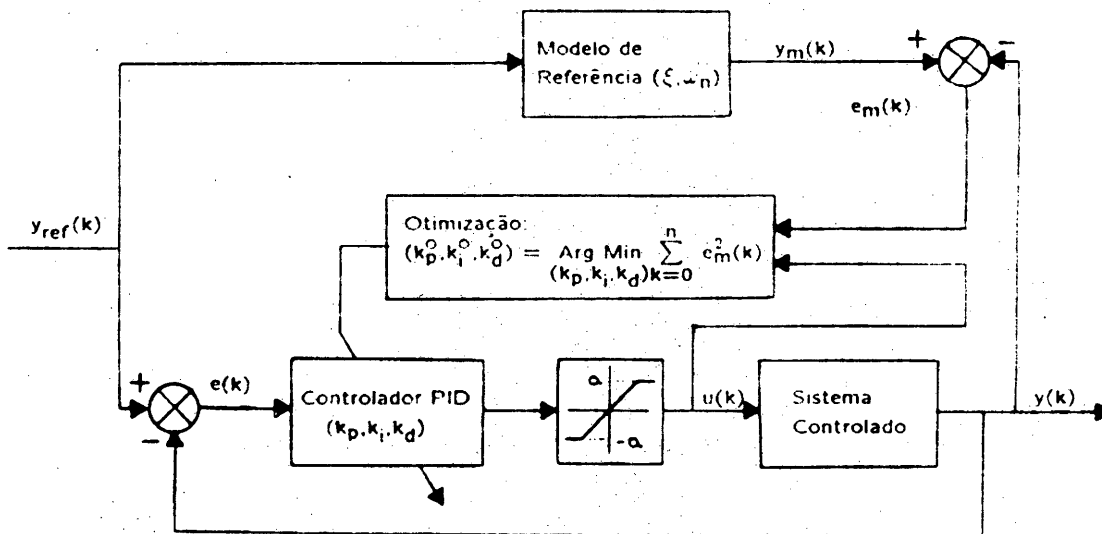


Figura 2.1. Diagrama de blocos para sintonização ótima de controladores PID digitais.

$$\beta_1 = 1 - e^{-\xi \omega_n T} (\cos(\sqrt{1-\xi^2} \omega_n T) + \frac{\xi}{\sqrt{1-\xi^2}} \sin(\sqrt{1-\xi^2} \omega_n T)) \quad (7)$$

$$\beta_2 = e^{-\xi \omega_n T} (e^{-\xi \omega_n T} (-\cos(\sqrt{1-\xi^2} \omega_n T) + \frac{\xi}{\sqrt{1-\xi^2}} \sin(\sqrt{1-\xi^2} \omega_n T))) \quad (8)$$

Considerando-se que a percentagem de sobre-elevação máxima (M) e o tempo de pico (t_p) da resposta do modelo de referência (4) ao degrau são dados por:

$$M = 100e^{-\pi\xi / \sqrt{1-\xi^2}} \quad t_p = \frac{\pi}{\omega_n \sqrt{1-\xi^2}} \quad (9)$$

percebe-se facilmente que os parâmetros ξ e ω_n especificados pelo projetista estão diretamente relacionados ao amortecimento e tempo de subida da resposta ao degrau.

Suponhamos agora que valores iniciais arbitrários de (k_p, k_i, k_d) sejam especificados para o controlador PID da Figura 2.1. e que uma entrada degrau seja aplicada, isto é, $y_{ref}(k) = 1$, for $k = 0, 1, \dots, n$. Então, a saída do sistema controlado será $\{y(k), k = 0, 1, \dots, n\}$ e a saída do modelo de referência $y_m(k)$ pode ser calculada com base em (5). Podemos assim calcular o erro $e_m(k)$ e avaliar a função custo.

$$J(k_p, k_i, k_d) = \sum_{k=0}^n e_m^2(k) \quad (10)$$

Obviamente, poderíamos também utilizar $|e_m(k)|$ ao invés de $e_m^2(k)$ em (10), o que usualmente origina uma resposta menos oscilatória.

A principal conveniência de se utilizar $e_m(k)$, e não $e(k)$, na função custo está no fato de que restrições de engenharia poderão ser consideradas com naturalidade: ao invés de arbitrar uma ponderação ρ para o controle, que não possui relação direta com o desempenho do sistema de controle, o projetista especifica diretamente o desempenho desejado. Isto será ilustrado quando apresentarmos os exemplos de simulação, na seção 4. No momento, comentaremos apenas um cenário típico: o projetista inicia especificando valores de ξ e ω_n que julga convenientes. Se alguma restrição for violada, ele decrece o valor de ω_n . Caso contrário, ele aumenta ω_n .

Neste ponto a sintonização ótima dos parâmetros do controlador PID se resume na minimização da função custo (10), isto é, na determinação de

$$(k_p^o, k_i^o, k_d^o) = \underset{(k_p, k_i, k_d)}{\text{Arg Min}} \sum_{k=0}^n e_m^2(k) \quad (11)$$

Neste trabalho empregaremos o método de Powell (Powell, 1964; Zangwill, 1967) para resolver o problema de minimização (11). Este método será utilizado porque ele evita o cálculo de derivadas, que é uma tarefa exigente em termos computacionais e susceptível a erros numéricos. O método requer apenas a avaliação da função custo (10), que é uma tarefa extremamente simples.

3. PRINCIPAIS CARACTERÍSTICAS DO AMBIENTE INTEGRADO

Para ser eficiente, a técnica de otimização apresentada na seção 2 deve ser programada em um ambiente integrado que permita ao projetista alterar parâmetros de projeto, tais como ξ e ω_n , e verificar rapidamente as consequências. Seria também conveniente que o projetista dispusesse de recursos gráficos, de modo a verificar se restrições foram violadas, etc. Adicionalmente, considerando-se que o projeto é feito utilizando-se um modelo do sistema controlado, seria importante introduzir recursos que permitissem ao projetista verificar a robustez do sistema de controle, variando os parâmetros do modelo, inserindo ruído, perturbações, etc. Finalmente, seria extremamente conveniente que o projeto pudesse ser validado, isto é,

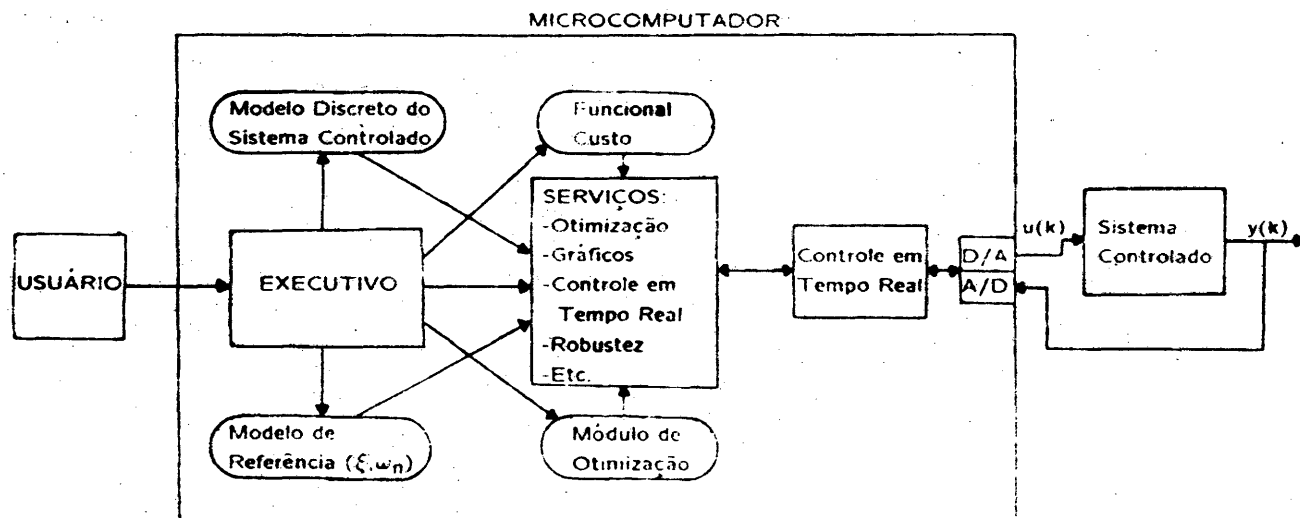


Figura 3.1. Principais características do ambiente integrado.

que o projetista controlasse o sistema físico real com o PID otimizado, e não apenas seu modelo.

O diagrama de blocos de um sistema integrado incorporando todas as facilidades mencionadas no parágrafo anterior é mostrado na Figura 3.1. Diversos módulos são disponíveis, de modo que o projetista possa selecionar facilmente as opções, alterar parâmetros de projeto, efetuar controle em tempo real, etc. A programação foi feita na linguagem C e o software pode ser executado em microcomputadores do tipo IBM e compatíveis.

A interface A/D-D/A sendo utilizada no momento é a Data Translation DT2808 (User Manual for PCLAB, 1985). O período de amostragem para controle em tempo real é definido programando-se, por software, o timer 0 do microcomputador. Isto é transparente para o usuário, que necessita apenas especificar o valor do período de amostragem desejado.

O módulo executivo, além de gerenciar os serviços de otimização, das saídas gráficas, etc., também orienta o usuário quanto à entrada e alteração dos dados: modelo de referência, modelo do sistema a ser controlador, etc.

Além de controlar o sistema físico com o controlador PID otimizado, o módulo de controle em tempo real também coleta dados referentes à saída deste sistema, $y(k)$ na Figura 3.1, para comparação com a saída caso o sistema físico fosse exatamente descrito pelo modelo (1). Caso a diferença entre essas saídas seja inaceitável, a fase de projeto deve continuar; talvez o projetista tenha especificado um modelo de referência muito rápido, o que causou saturação do controle, e então, da equação (11), ele tem que reduzir o valor de ω_n , etc.

4. EXEMPLOS

Dois exemplos são apresentados para ilustrar a eficiência e utilidade da técnica de otimização e do ambiente integrado descritos anteriormente. O primeiro exemplo refere-se a um sistema simulado, e conseqüentemente o modelo (1) está disponível. Contudo, o segundo exemplo concerne um sistema físico real e assim a fase de modelagem constitui, por si, um problema não trivial. Para obter um modelo discreto deste segundo exemplo utilizaremos a técnica descrita em (Hemerly, 1990), relativa à identificação estrutural e paramétrica de processos industriais.

4.1. Exemplo 1

Considere o sistema de terceira ordem com a seguinte função de transferência

$$G(s) = \frac{4.23}{s^3 + 2.14s^2 + 9.28s + 4.23} \quad (12)$$

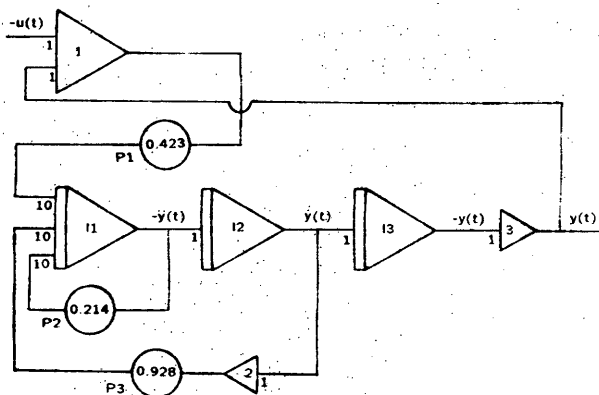


Figura 4.1. Simulação do sistema do exemplo 1.

Este sistema possui pólos em -0.5 , $-0.82 + j2.79$ e $-0.82 - j2.79$. Devido ao pólo real lento em -0.5 , este sistema é particularmente difícil de ser controlado e a maioria das técnicas heurísticas para sintonizar controladores PID estão fadadas a fracassarem. Efetivamente, este sistema foi também considerado por (Anderson, Blankenship e Lebow, 1988), onde foi mostrado que a técnica de Ziegler-Nichols praticamente fracassa.

O sistema (12) foi simulado no computador analógico Comdyna GP-6, conforme mostrado na Figura 4.1. Na Figura 4.2. temos a resposta natural do sistema (12) para uma entrada do tipo degrau unitário, estando o eixo do tempo em múltiplos do período de amostragem $T = 0.1s$, isto é, o tempo varia de 0 a 10 segundos.

A função de transferência discreta $G(z)$ correspondente, para período de amostragem $T = 0.1s$, pode ser obtida, por exemplo, utilizando-se MATLAB e é dada por:

$$G(z) = \frac{0.0007z^{-1} + 0.0025z^{-2} + 0.0006z^{-3}}{1 - 2.7225z^{-1} + 2.5336z^{-2} - 0.8073z^{-3}} \quad (13)$$

Podemos agora iniciar o procedimento de otimização. Obviamente, desejaríamos que a resposta ao degrau do sistema de controle em malha fechada fosse mais rápida que aquela em malha aberta, apresentada na Figura 4.2. Portanto, escolhendo $\xi = 0.7$, uma escolha inicial razoável para ω_n é $\omega_n = 2$ rad/s, o que, de (9), implica $t_p \approx 2.2s$. Além dos parâmetros em (13) e dos valores de ξ e ω_n mencionados, fornecemos os seguintes dados para o algoritmo de otimização: $y_{ref}(k) = 2.0$, $n = 100$, valores iniciais dos parâmetros $(k_p, k_i, k_d) = (0, 0, 0)$, tolerância para o método de Powell = 0.1. No presente exemplo utilizamos $|e_m(k)|$ na função custo $J(k_p, k_i, k_d)$.

O algoritmo de otimização no momento está sendo executado em um microcomputador do tipo IBM-XT a 10MHz, com coprocessador aritmético INTEL 8087. Com as condições iniciais apresentadas no parágrafo anterior, o algoritmo de otimização convergiu, após aproximadamente 50 segundos de computação, para os valores ótimos de parâmetros $(k_p^o, k_i^o, k_d^o) = (1.346686; 0.120385; 0.287572)$. O desempenho do sistema de controle e o controle correspondente estão apresentados na Figura 4.3.

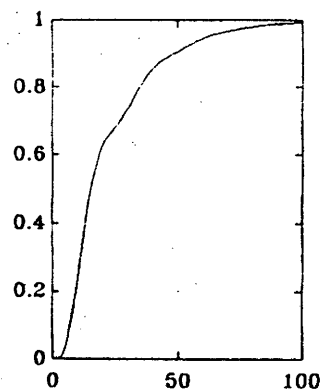


Figura 4.2. Resposta do degrau.

O eixo do tempo está em múltiplos do período de amostragem $T = 0.1s$. A curva pontilhada no primeiro gráfico da Figura 4.3 corresponde à saída do modelo de referência $y_m(t)$, e a curva cheia corresponde à saída do sistema controlado. O controle é mostrado no segundo gráfico da Figura 4.3. De modo a decidir se o projeto é realista ou não, temos que saber, por exemplo, qual é o nível de saturação do controle. Em outras palavras, temos que saber o valor de "a" na Figura 2.1. No presente caso este valor é 5V. Considerando então que da Figura 4.3 o desempenho do sistema de controle é satisfatório e que o controle é realista, podemos dar por encerrada a etapa de projeto. (Uma vez que o sistema controlado está sendo simulado em um computador analógico, o modelo (13) é bastante preciso. Portanto, se aplicarmos controle em tempo real obteremos praticamente os mesmos gráficos mostrados na Figura 4.3. Estes gráficos serão omitidos, para economizar espaço). Caso tivesse havido saturação no controle, isto é, caso $u(t)$ tivesse excedido 5V na Figura 4.3, o projetista teria que continuar com a fase de projeto, especificando um valor menor para ω_n , etc.

Convém ressaltar que a solução de (11) não é única. Assim, o projetista é aconselhado a executar outras vezes o algoritmo de otimização, partindo de diferentes condições iniciais para (k_p, k_i, k_d) .

Ainda em relação ao exemplo 4.1, faremos uma verificação da robustez do sistema de controle ao distúrbio de carga. Para tanto, utilizaremos o controlador obtido na Figura 4.3 consideraremos $d(k) = 0$, para $k \leq 100$, $d(k) = -0.5$, para $k > 100$. Na Figura 4.4 temos o resultado obtido, que parece aceitável.

Seria natural o usuário se perguntar se um desempenho melhor que aquele da Figura 4.4 poderia ser obtido se o distúrbio de carga fosse considerado no processo de otimização. Ou seja, o usuário estaria interessado em reduzir a influência do distúrbio de carga, mas sem penalizar em excesso o desempenho transitório do sistema de controle. Afinal, estes objetivos são em geral conflitantes (Hang, 1989). Na Figura 4.5 apresentamos o resumo do projeto considerando-se o distúrbio de carga no processo de otimização. Comparando-se as Figuras 4.4 e 4.5, percebe-se que na Figura 4.5 a influência do distúrbio de carga foi reduzida (a saída $y(t)$ decai menos e retorna mais rapidamente ao valor de referência). Adicionalmente o desempenho transitório do sistema de controle difere pouco daquele mostrado na Figura 4.4. Assim, caso robustez ao distúrbio de carga fosse relevante no presente exemplo, o controlador obtido na Figura 4.5 seria preferível àquele obtido na Figura 4.3.

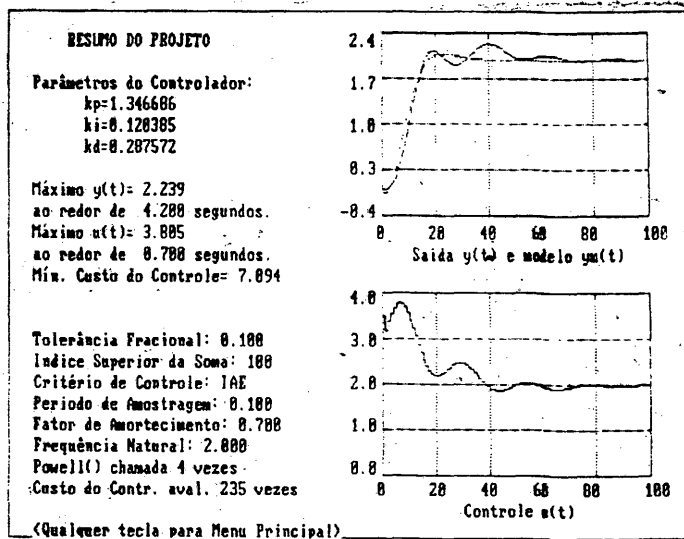


Figura 4.3. Resumo do projeto para o exemplo 1.

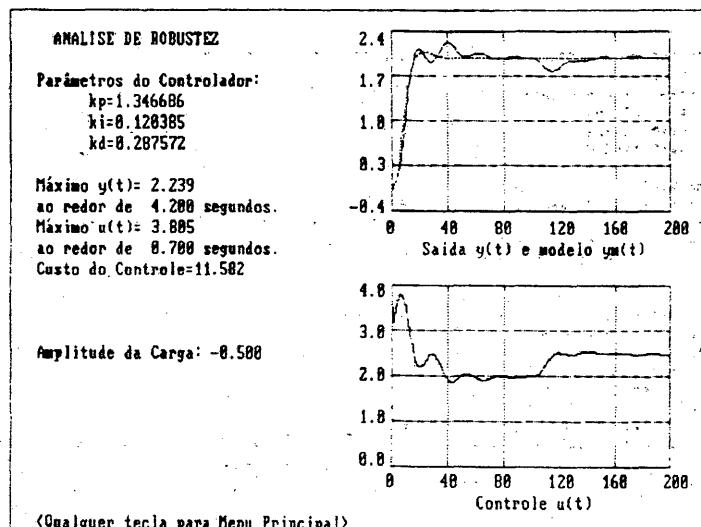


Figura 4.4. Robustez ao distúrbio de carga, relativa ao exemplo 1.

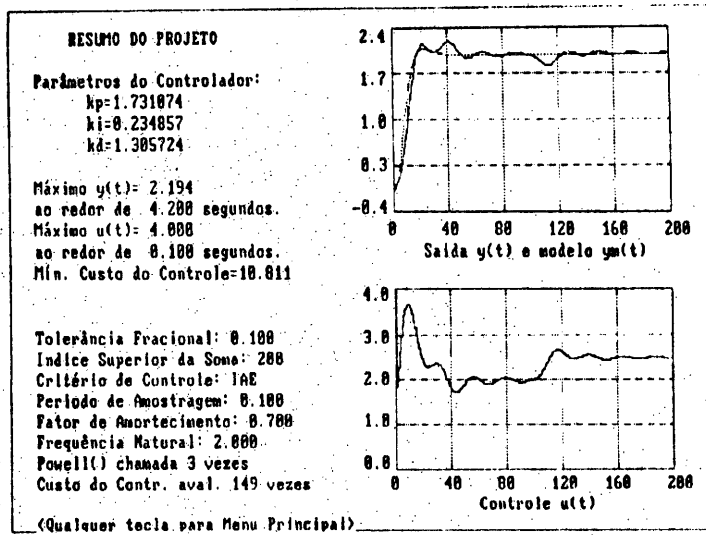


Figura 4.5. Resumo do projeto para o exemplo 1, considerando-se o distúrbio de carga.

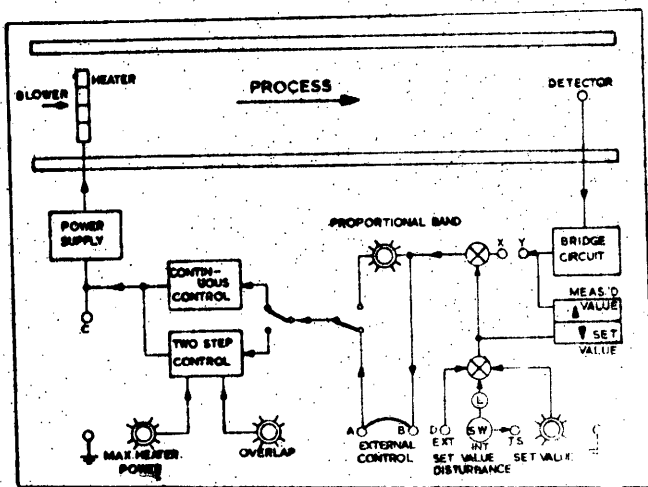


Figura 4.6. Diagrama de blocos do processo térmico PT326.

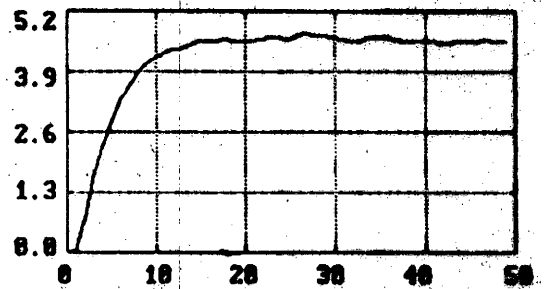


Figura 4.7. Resposta do degrau.

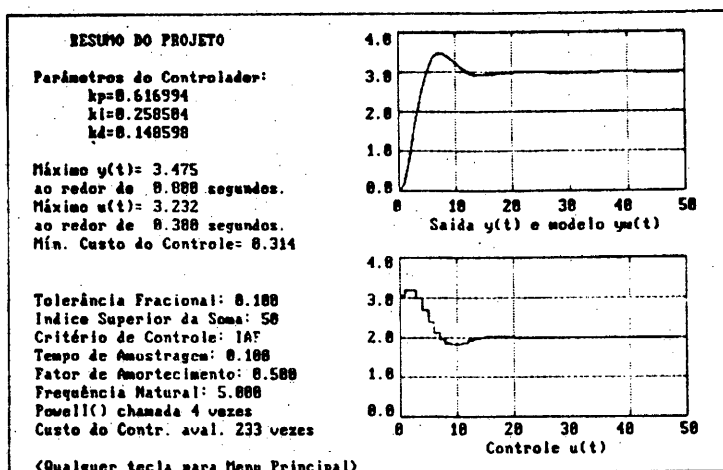


Figura 4.8. Resumo do projeto para o exemplo 2.

4.2. Exemplo 2

Consideraremos agora o processo térmico PT326 da Feedback, cujo diagrama de blocos é mostrado na Figura 4.6. Basicamente, ar é soprado através de um tubo, sendo aquecido na entrada do mesmo por uma malha de resistores, excitada pelo controle $u(t)$, aplicado no ponto X da Figura 4.6. A temperatura do ar é medida por um termistor e o sinal de saída correspondente $y(t)$ pode mensurável no ponto Y da Figura 4.6. A resposta natural a um degrau de amplitude 3 é mostrada na Figura 4.7. Com base nesta figura concluímos que um período de amostragem $T = 0.1s$ é adequado para o controle digital (Vide (Aström e Wittenmark, 1984) para detalhes sobre a seleção do período de amostragem). Na Figura 4.7 o eixo do tempo está em múltiplos do período de amostragem, e assim o tempo varia de 0 a 5 segundos.

Para modelar o processo térmico mostrado na Figura 4.6, utilizamos a técnica descrita em (Hemerly, 1990). Para período de amostragem $T = 0.1s$, obtivemos o seguinte modelo discreto de segunda ordem

$$G(z) = \frac{0.0575z^{-1} + 0.1663z^{-2}}{1 - 1.1976z^{-1} + 0.3477z^{-2}} \quad (14)$$

Da resposta natural ao degrau, mostrada na Figura 4.7, notamos que um valor razoável para ω_n é $\omega_n = 5$ rad/s. Assim, escolhendo $\xi = 0.5$, de (9) teremos um tempo de pico $t_p \approx 0.7s$. Suprindo o algoritmo de otimização com os dados adicionais $y_{ref}(k) = 3.0$, $n = 50$, valores iniciais dos parâmetros $(k_p, k_i, k_d) = (0,0,0)$ e tolerância para o método de Powell = 0.1, após aproximadamente 25 segundos de computação o algoritmo convergiu para os seguintes valores ótimos de parâmetros $(k_p^o, k_i^o, k_d^o) = (0.616994; 0.258504; 0.148598)$. O desempenho do sistema de controle resultante e o controle correspondente estão mostrados na Figura 4.8.

Do primeiro gráfico da Figura 4.8 concluímos que praticamente conseguimos reproduzir a resposta do modelo de referência, i.e., $y_m(t)$ e $y(t)$ são praticamente indistinguíveis.

Adicionalmente, do segundo gráfico da Figura 4.8 vemos que o controle não excede o nível de saturação de 5V. Podemos então

concluir que conseguiremos controlar satisfatoriamente o modelo do sistema controlado. O problema agora é saber se conseguiremos também controlar satisfatoriamente o sistema físico real mostrado na Figura 4.6.

Antes de aplicarmos controle em tempo real ao sistema da Figura 4.6, é conveniente que efetuem uma análise preliminar da robustez do sistema de controle, no que se refere a variação paramétrica. Isto pode ser facilmente efetuado com o ambiente integrado descrito anteriormente. Há um módulo que permite ao projetista variar os parâmetros do modelo do sistema controlado e avaliar as consequências desta variação no desempenho do sistema de controle. Objetivando economizar espaço, apresentaremos apenas as consequências de se aumentar em 8% todos os parâmetros do modelo (14). Vide Figura 4.9. Como poderíamos esperar, o desempenho na Figura 4.9 é pior que aquele mostrado na Figura 4.8. Contudo, a estabilidade é mantida e não há saturação no controle. Isto sugere que o sistema de controle resultante do processo de otimização possui alguma robustez. Assim, com certa confiança, podemos prosseguir para a última fase do projeto: aplicar controle em tempo real e assim validar o projeto.

Na Figura 4.10 apresentamos um resumo da fase de controle em tempo real. No primeiro gráfico, em curva pontilhada, temos a saída esperada, denotada por $y(t)$ Sim, i.e., a saída caso o sistema controlado da Figura 4.6 fosse exatamente descrito por (14). A curva cheia do primeiro gráfico na Figura 4.10 corresponde à saída, denotada por $y(t)$ CTR, do sistema físico mostrado na Figura 4.6, quando controlado pelo controlador PID digital com parâmetros otimizados $(k_p^o, k_i^o, k_d^o) = (0.616994; 0.258504; 0.148598)$.

No segundo gráfico da Figura 4.10 temos o controle esperado $u(t)$ Sim e o controle real $u(t)$ CTR. Considerando-se que as discrepâncias nos gráficos da Figura 4.10 são pequenas, concluímos que o projeto pode ser validado, isto é, conseguimos controlar o sistema físico real. (Caso tivéssemos fracassado, teríamos que prosseguir com o projeto, decrescendo o valor de ω_n , ou então reidentificando o sistema de modo a obter um melhor modelo, etc.).

5. CONCLUSÕES

Neste trabalho apresentamos uma técnica de otimização para sintetizar os parâmetros de controladores PID digitais e um ambiente

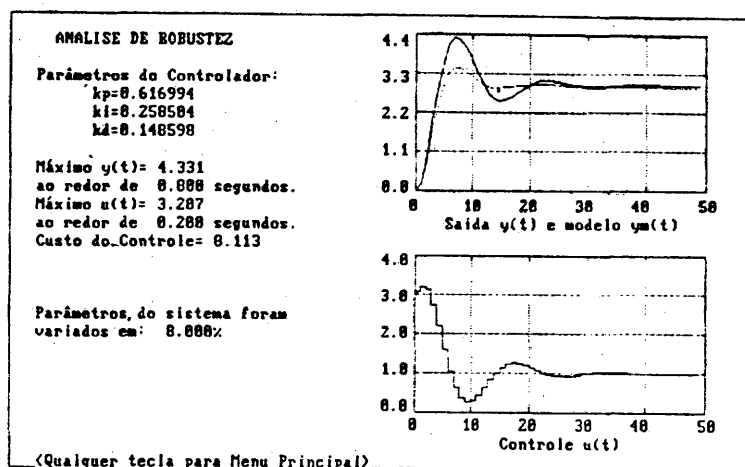


Figura 4.9. Desempenho do sistema de controle ao se elevar 8% os parâmetros de (14).

integrado para implementá-la. A otimização é efetuada no domínio do tempo e a função custo envolve o erro entre as saídas do sistema controlado e um modelo de referência, que especifica o desempenho desejado do sistema de controle. A utilização do modelo de referência facilita apreciavelmente a fase de projeto, porque os parâmetros de projeto têm influência direta no desempenho do sistema de controle. O ambiente integrado apresenta diversas facilidades, tais como módulos, gráficos e módulo para controle em tempo real, possibilitando a validação do projeto.

A técnica de otimização apresentada pode também ser utilizada para controlar sistemas suavemente não-lineares: identificamos o sistema em diferentes pontos de operação, obtendo diversos modelos (Isto pode ser feito com a técnica descrita em (Hemerly, 1990)); a seguir sintonizamos um controlador PID para cada modelo e finalmente utilizamos a técnica de programação de ganhos para selecionar os controladores em tempo real.

O ambiente integrado descrito neste trabalho poderia ser utilizado como supervisor em um sistema multimalhas. Poderíamos, por exemplo, endereçar os controladores PID digitais, obter dados sobre os sistemas das respectivas malhas e identificar tais sistemas empregando a técnica apresentada em (Hemerly, 1990). A seguir, sintonizaríamos os parâmetros dos controladores PID e transferiríamos tais parâmetros para os microprocessadores desses controladores. Adicionalmente, isto poderia ser feito periodicamente, de tal modo que as malhas continuariam a operar próximas do ponto ótimo de operação mesmo que houvesse variações nas dinâmicas dos sistemas controlados.

6. AGRADECIMENTOS

Aos revisores, pelas sugestões efetuadas.

7. REFERÊNCIAS

AL-ASSADI, S.A.K. and AL-CHALABI, L.A.M. (1987), Optimal Gain for Proportional-Integral-Derivative Feedback, IEEE Contr. Syst. Mag., Vol. 7, N° 6, pp. 16-19, December.

ANDERSON, K.L.; BLANKENSHIP, G.L. and LEBOW, L.G. (1988), A Rule-Based Adaptive PID Controller, Proceedings of the 27th IEEE CDC, Austin, Texas, pp. 564-569, December.

ÅSTRÖM, K.J. and WITTENMARK, B. (1984), Computer Controlled Systems-Theory and Design, Prentice-Hall, Englewood Cliffs, N.J.

FU, M.; OLBROT, A.W. and POLIS, M.P. (1989), Comments on "Optimal Gain for Proportional-Integral-Derivative Feedback", IEEE Contr. Syst. Mag., Vol. 9, N° 1, pp. 100-101, January.

HANG, C.C. (1989), The Choice of Controller Zeros, IEEE Control Systems Magazine, January, pp. 72-75.

HEMERLY, E.M. (1990), Identificação Estrutural e Paramétrica de Processos Industriais, 4º Congresso Nacional de Automação Industrial, Julho, pp. 73-81.

ISERMANN, R. (1981), Digital Control Systems, Springer-Verlag, Berlin.

KINNEY, T.B. (1983), Tuning Process Controllers, Chem. Eng., p. 67, September.

POWELL, M.J.D. (1964), An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives, Computer Journal, Vol. 7, pp. 155-162, July.

SHINSKEY, F.G. (1979), Process Control Systems, 2nd Ed., McGraw-Hill, New York.

USER-MANUAL for PCLAB (1985), Data Translation, Inc., 100 Locke Drive, Marlborough, M.A.

ZANGWILL, W.I. (1967), Minimizing a Function without Calculating Derivatives, Computer Journal, Vol. 10, pp. 293-296, November.

ZIEGLER, J.G. and NICHOLS, N.B. (1942), Optimum Settings for Automatic Controllers, Trans ASME, Vol. 64, p. 759.

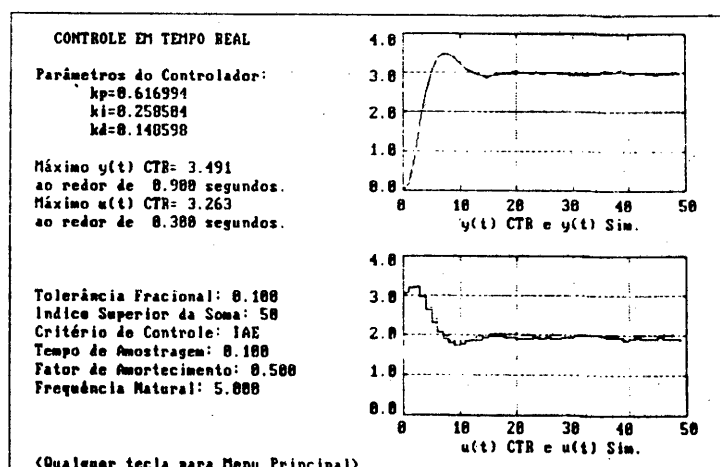


Figura 4.10. Controle em tempo real do processo térmico da Figura 4.4..