

---

# PID-TOOLBOX:

## UMA FERRAMENTA PARA O ENSINO E

## AJUSTE DE CONTROLADORES PIDS

*Jorge Otávio Trierweiler*  
*Maurício Simões Posser*

{Jorge,Posser}@enq.ufrgs.br  
Departamento de Engenharia Química  
Universidade Federal do Rio Grande do Sul

---

### RESUMO

O controlador PID é sem dúvida o algoritmo de controle mais comum, sendo muito utilizado na indústria de processos. A maioria das malhas de controle *feedback* são controladas por este algoritmo ou variações secundárias dele. É implementado em muitas formas diferentes, como um controlador analógico sozinho ou como uma parte de um SDCD (sistema digital de controle distribuído). O ajuste de controladores tipo PID faz parte da vida diária de milhares de engenheiros de instrumentação e controle.

Embora sejam considerados como sendo o "feijão com arroz" entre os diversos tipos de controladores, praticamente não há nenhum *toolbox* em *MATLAB* que automatize o projeto de controladores PIDs e, mais ainda, permita a fácil e sistemática comparação entre os diferentes métodos existentes. Para preencher esta lacuna, foi desenvolvido o *PID Controller Design Toolbox* (ou simplesmente *PID-TB*), o qual apresenta diversos métodos para determinação dos parâmetros de um controlador PID. Os métodos de projeto diferem com respeito do conhecimento sobre a dinâmica do processo requerida. Por exemplo, o método de ajuste de Ziegler-Nichols (ZN) baseado em  $K_u$  e  $P_u$  está relacionado ao ganho do processo e a sua rapidez. Já uma outra versão também proposta por ZN baseia-se em parâmetros que são simplesmente obtidos da aproximação da dinâmica do sistema através de uma função de primeira ordem com tempo morto.

O *PID-TB* contempla uma série de métodos pertencentes a ambas categorias, permitindo uma comparação sistemática entre eles, bem como a facilidade de documentar os resultados obtidos. Além disto, a interface gráfica desenvolvida para o *PID-TB* é seqüencial e auto-explicativa, conduzindo o usuário através dos diversos passos necessários para o desenvolvimento do projeto de um controlador. Adicionalmente, um eficiente sistema de ajuda, o *Help On-line*, também está disponível contendo toda a teoria necessária para compreender e interagir melhor com o *PID Controller Design Toolbox*, constituindo-se dessa forma numa valiosa ferramenta de ensino.

# 1 INTRODUÇÃO

Controladores do tipo PID são sem sombra de dúvida os mais usados na indústria de processos. Embora sejam considerados como sendo o "feijão com arroz" entre os diversos tipos de controladores praticamente não há nenhum toolbox em MATLAB que automatize o projeto de controladores PIDs e, mais ainda, que permita a fácil e sistemática comparação entre os diferentes existentes.

## 1.1 Objetivos

Visando suprir o ambiente MATLAB com uma ferramenta para o projeto e comparação de controladores PIDs foi desenvolvido o PID-Toolbox, o qual tem por objetivos:

1. Apresentar diferentes métodos de ajuste de controladores PIDs geralmente apresentados em cursos de graduação em controle de processos.
2. Permitir através de simulações a visualização da contribuição dos diversos parâmetros de ajuste de controladores PIDs.
3. A partir de dados de planta, utilizando técnicas simples de identificação, tanto contínua quanto discreta, desenvolver modelos dinâmicos capazes de capturar as características dinâmicas fundamentais para um bom projeto de controlador.
4. Através de critérios que levam em conta a robustez e o desempenho em malha fechada permitir a sistemática comparação entre os diferentes métodos.
5. Facilitar a documentação dos resultados gerados, permitindo a fácil exportação dos mesmos para planilhas e editores de texto.

## 1.2 Público Alvo

Como controladores tipo PID são intensamente utilizados em qualquer curso de graduação em controle de processos, o público alvo natural do PID-TB são os alunos dessa disciplina, os quais além de visualizarem as diferenças entre os diversos métodos aprendem a utilizar uma ferramenta que irá ajudá-los na sua vida profissional. Naturalmente o PID-TB pode e deve ser utilizado intensamente por Engenheiros que sejam responsáveis pelo ajuste de controladores em plantas industriais.

# 2 FUNDAMENTOS TEÓRICOS

O controlador tipo PID é sem dúvida o algoritmo de controle mais comum. A maioria das malhas de controle *feedback* são controladas por este algoritmo ou variações secundárias dele. É implementado em muitas formas diferentes, como um controlador analógico sozinho ou como uma parte de um SDCD (sistema digital de controle distribuído). O ajuste de controladores tipo PID faz parte da vida diária de milhares de engenheiros de instrumentação e controle.

O PID-TB apresenta diversos métodos para determinação dos parâmetros de um controlador PID. Os métodos de projeto diferem com respeito do conhecimento sobre a dinâmica do processo requerida. Por exemplo, o método de ajuste de Ziegler-Nichols (ZN) baseado em  $K_u$  e  $P_u$  é

caracterizado por esses dois parâmetros ( $Ku$  e  $Pu$ ), sendo um relacionado ao ganho do processo (ganho ultimate,  $Ku$ ) e outro que descreve a sua rapidez (i.e., período ultimate,  $Pu$ ). Já uma outra versão também proposta por ZN baseia-se em parâmetros que são simplesmente obtidos da aproximação da dinâmica do sistema através de uma função de primeira ordem com tempo morto. O PID-TB contempla uma série de métodos pertencentes a ambas categorias. Mas antes de apresentá-los iremos ver como os modelos dinâmicos podem ser obtidos a partir de dados de planta utilizando técnicas de identificação contínua e discretas.

## 2.1 Métodos de Identificação

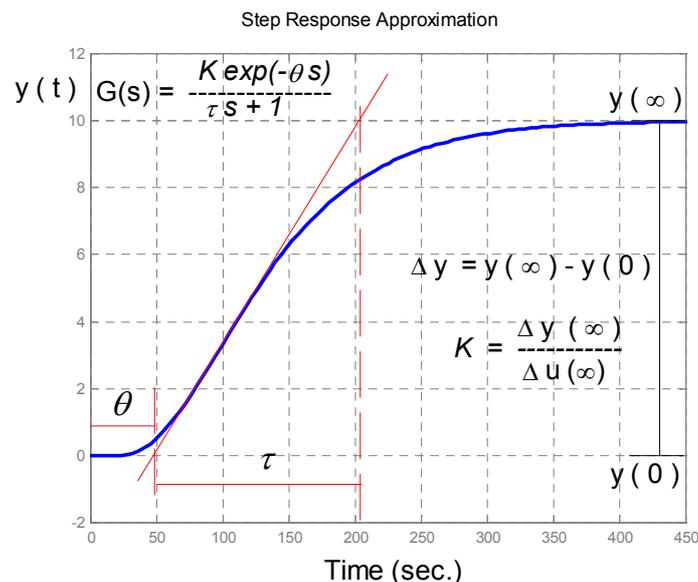
A dinâmica de muitos processos industriais pode ser satisfatoriamente aproximada por funções de primeira ou segunda ordem com tempo morto. Naturalmente os métodos apresentados sucintamente no apêndice podem ser utilizados para obtê-los a partir de um sinal genérico de entrada. Aqui iremos mostrar como esses parâmetros podem ser facilmente obtidos a partir da resposta a uma perturbação degrau.

*Aproximação através de uma função de primeira ordem com tempo morto:*

Uma função de primeira ordem com tempo morto é dada por:

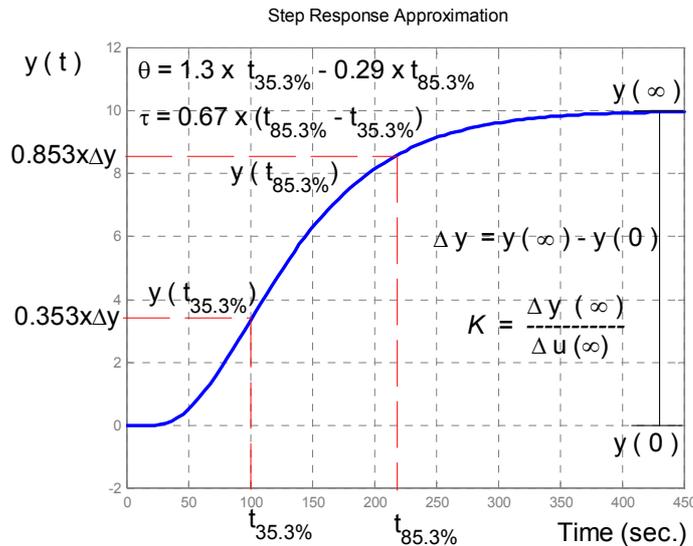
$$G(s) = \frac{K}{\tau s + 1} \exp(-\theta s) \quad (1)$$

para se obter os parâmetros que descrevem essa função (i.e.,  $K$ ,  $\tau$  e  $\theta$ ) o *Método da Tangente* e o *Método de Sundaresan and Krishnaswamy* [SK77] podem ser facilmente aplicados.



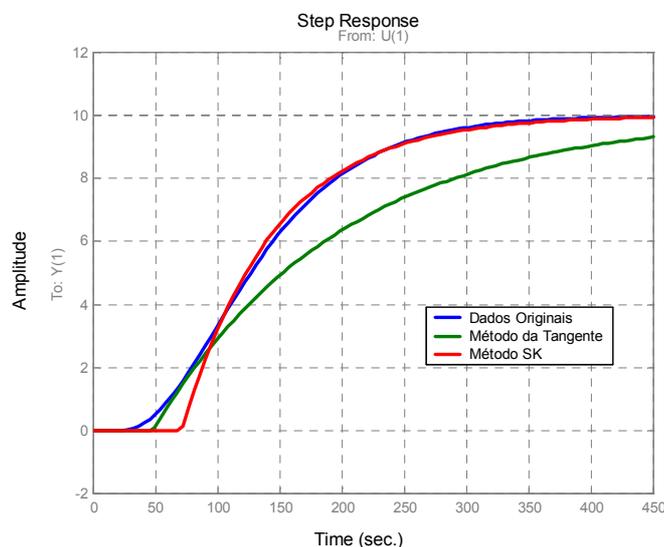
**Figura 1:** Representação esquemática do método da tangente

A figura 1 mostra esquematicamente como são obtidos os parâmetros do modelo aproximado através do método da tangente. Um ponto chave desse método consiste em se passar uma tangente no ponto de inflexão (i.e., ponto onde a derivada da curva atinge o seu valor máximo, ou seja, o ponto de maior inclinação). O tempo morto será dado pelo ponto correspondente a interseção da reta tangente com a linha correspondente ao valor base, que, no caso da figura, é igual a zero ( $y=0$ ). Já a constante de tempo será dada pela diferença entre o tempo correspondente a interseção da reta tangente com a linha  $y(\infty)$  e o tempo morto, conforme indicado na figura.



**Figura 2:** Representação esquemática do método proposto em [SK77]

Quando o sinal que está sendo aproximado tiver muito ruído fica extremamente difícil se determinar o ponto de inflexão. Uma método simples que evita a utilização do ponto de inflexão foi proposta por *Sundaresan and Krishnaswamy* [SK77] e consiste (veja figura 2) na determinação dos tempos  $t_{35,3\%}$  e  $t_{85,3\%}$  que correspondem respectivamente aos tempos em que a resposta atinge 35.3% e 85.3% da sua variação final  $\Delta y$  (i.e.,  $\Delta y = y(\infty) - y(0)$ ). De posse desses valores pode-se calcular facilmente o tempo morto do processo e sua constante de tempo através das seguintes equações:  $\theta = 1,3 \times t_{35,3\%} - 0,29 \times t_{85,3\%}$  e  $\tau = 0,67 \times (t_{85,3\%} - t_{35,3\%})$ . Esses valores de  $\theta$  e  $\tau$  praticamente minimizam a diferença entre o valor medido e o predito pelo modelo aproximado se o critério dos mínimos quadrados for tomado como base da função de otimização.



**Figura 3:** Representação esquemática do método proposto em [SK77]

A figura 3 mostra a comparação entre os dois métodos de aproximação. Observe que o método de SK tende sempre a gerar um tempo morto maior e uma constante de tempo menor que o método da tangente. Em termos de controle, as aproximações geradas pelo método SK deverão em geral serem escolhidas. No PID-TB, a resposta do sistema é aproximada pelo método SK, mas o usuário através de uma interface amigável poderá facilmente alterar os valores inicialmente gerados.

Para muitos processos a aproximação de primeira ordem com tempo não é suficientemente acurada. Nesses casos pode-se aproximar a resposta utilizando-se sistemas de segunda ordem com tempo morto. Métodos similares aos apresentados aqui podem ser encontrados na literatura, veja p.ex. [SEM89, seção 7.3]. Esses métodos por motivos didáticos foram também incluídos no PID-TB. Naturalmente, quando se deseja obter um modelo mais acurado deve-se fazer uso de técnicas de identificação para sistemas discretos ou contínuos. O PID-TB já possui uma interface que permite acessar automaticamente as funções disponíveis no "System Identification Toolbox"[SID99]. No apêndice A o leitor encontrará uma pequena discussão desses métodos de identificação. A mesma interface permite que se acesse as funções disponíveis no "CONTinuous-Time System Identification Toolbox" [CONTSID 99].

## 2.2 Parâmetros de ajuste do controlador PID

Em um controlador PID a ação de controle é gerada através da soma de três termos: ação proporcional ( $U_P$ ), integral ( $U_I$ ) e derivativa ( $U_D$ ). Embora todos os controladores PID possuam esses três termos, as implementações do algoritmo PID poderão diferir consideravelmente. O PID-TB implementa a forma padrão da ISA (Instrument Society of America), a qual pode ser escrita da seguinte forma:

$$\Delta U(s) = K_P \left\{ \underbrace{\left[ bY_{set}(s) - Y(s) \right]}_{U_P} + \underbrace{\frac{[Y_{set}(s) - Y(s)]}{\tau_I s}}_{U_I} + \underbrace{\frac{\tau_D s}{N} [cY_{set}(s) - Y(s)]}_{U_D} \right\} \quad (2)$$

Para se evitar ações de controle extremamente brusca provocadas pela ação derivativa durante mudanças bruscas de setpoint se utiliza no PID-TB o parâmetro  $c=0$  em todas as simulações. Portanto os parâmetros de ajuste desse controlador são os seguintes:  $b$ ,  $K_P$ ,  $\tau_I$ ,  $\tau_D$  e  $N$ . O parâmetro  $b$  é responsável por amenizar a ação de controle inicial para mudanças no setpoint, reduzindo dessa forma consideravelmente o *overshoot* para esse tipo de perturbação. Através do correto ajuste do parâmetro  $b$  se consegue atingir ótimos desempenhos tanto para mudanças de setpoint (características servo), quanto na compensação de distúrbios na carga (características regulatórias). O único método de ajuste disponível no PID-TB que faz uso desse parâmetro é o método de Kappa-Tau. Nos demais métodos  $b=1$ . O parâmetro  $N$  serve como filtro do ruído, o qual é o principal vilão de controladores com ação derivativa. No PID-TB,  $N$  é considerado como sendo igual 10, mas esse valor pode ser alterado pelo usuário. Quanto maior for valor de  $N$  mais próximo da ação derivativa pura estará o controlador. Já quanto menor for  $N$  mais filtrado será o ruído. A tabela abaixo ilustra o efeito dos parâmetros  $K_P$ ,  $\tau_I$  e  $\tau_D$  na resposta do sistema.

**Tabela:** Efeito dos parâmetros  $K_P$ ,  $\tau_I$  e  $\tau_D$

Resposta em malha fechada	Tempo de subida	Overshoot	Tempo de assentamento	Offset	Efeito do ruído
$\uparrow K_P$	$\downarrow$	$\uparrow$	$\uparrow \approx$	$\downarrow$	$\uparrow \approx$
$\uparrow 1/\tau_I, \downarrow \tau_I$	$\downarrow$	$\uparrow \uparrow$	$\uparrow$	$\ominus$	$\approx$
$\uparrow D = \tau_D$	$\approx$	$\downarrow \downarrow$	$\downarrow \downarrow$	$\approx$	$\uparrow \uparrow$

$\downarrow$  Diminui,  $\uparrow$  Aumenta,  $\approx$  Pequenas variações,  $\ominus$  Elimina

## 2.3 Métodos de Ajuste Implementados

O PID-TB possui duas categorias de métodos de ajuste implementadas: (a) Métodos de ajuste baseados em funções de primeira ordem com tempo morto (também referidos como métodos baseados em  $K$ ,  $\tau$  e  $teta$ ) e (b) Métodos de ajuste baseados em  $K_u$  e  $P_u$ .

### 2.3.1 Métodos baseados em funções de Primeira ordem com tempo Morto

Dada a sua grande simplicidade aliada a sua grande aplicabilidade e satisfatória qualidade em aproximar sistemas reais compostos por dinâmicas de primeira ordem em série (esse tipo de dinâmica geralmente aparece em processos industriais), fez com que uma categoria de métodos fossem especialmente desenvolvidos para ajustar controladores PIDs a partir do ganho  $K$ , tempo morto  $\theta$  e constante de tempo  $\tau$  de uma função de primeira ordem com tempo morto. Naturalmente esses métodos, que a partir de agora serão denominados de métodos baseados em  $K$ ,  $\tau$  e  $teta$ , Cabe salientar que principalmente para sistemas subamortecidos a aplicação desse tipo de método de ajuste não é recomendável. Entretanto, para uma imensa classe de problemas essa classe de métodos representa um bom compromisso entre tempo investido no projeto do controlador e desempenho alcançado no projeto. Por isso, podemos entender facilmente porque esses métodos fazem parte de qualquer curso de graduação em controle de processos. No PID-TB o usuário poderá encontrar uma gama enorme desse tipo de métodos, os quais serão nas próximas subsecções sucintamente discutidos.

#### 2.3.1.1 Ziegler-Nichols baseado em $K$ , $\tau$ e $teta$

No seu artigo clássico, Ziegler e Nichols<sup>46</sup> também propuseram regras de ajuste visando alcançar uma razão de decaimento de  $\frac{1}{4}$  na resposta em malha fechada para sistemas de primeira ordem com tempo morto. A tabela a seguir sumariza a regra de ajuste por eles proposta. Cabe salientar que controladores ajustados dessa forma são extremamente sensíveis a razão  $\theta/\tau$ , não sendo *recomendado* o seu uso *fora* do intervalo  $0,1 < \theta/\tau < 1$ .

#### 2.3.1.2 Método de Cohen-Coon

Cohen-Coon também desenvolveram regras de ajuste de controladores visando se obter respostas em malha fechada com uma razão de decaimento de  $\frac{1}{4}$  (aproximadamente 20% de overshoot). Esse tipo de resposta apresenta melhor desempenho para rejeitar distúrbios na carga que é o principal objetivo a ser alcançado para processos contínuos que trabalham em um mesmo ponto de operação. Para esses sistemas, o principal papel desempenhado pelo controlador é a rejeição de distúrbios. Esse método possui um intervalo de aplicação similar ao ZN ( $K$ ,  $\tau$  e  $teta$ ), ou seja, não deverá ser utilizado fora do seguinte intervalo:  $0,1 < \theta/\tau < 1$ .

---

<sup>46</sup> John G. Ziegler e Nathaniel B. Nichols escreveram o seu papel seminal em ajuste de controladores industriais (Optimum Settings for Automatic Controllers, ASME Transactions, Nov. 1942, p759), o qual trouxe ordem ao caos. Simplicidade substituiu a complexidade. Posteriormente, esse artigo foi republicado várias vezes, o que seguramente o torna um dos artigos mais publicados sobre esse assunto. Desde de então vários autores o tomam como base de comparação quando apresentam um novo método de ajuste. Naquele época, eles não tiveram nenhuma idéia sobre a grande contribuição que tinham feito. Ambos morreram em 1997, Nichols em abril a idade 82 e Ziegler em dezembro com idade 88. Um capítulo na vida de controle automático terminou, mas o livro continua e quem sabe poderá contar com a sua valiosa contribuição.

**Tabela 1:** Métodos de ZN( $K, \tau, \theta$ ) e Cohen-Coon

	ZN ( $K, \tau$ e $\theta$ )			Método de Cohen-Coon		
	P	PI	PID	P	PI	PID
$K_P$	$\frac{1}{K} \left( \frac{\tau}{\theta} \right)$	$\frac{0,9}{K} \left( \frac{\tau}{\theta} \right)$	$\frac{1,2}{K} \left( \frac{\tau}{\theta} \right)$	$\frac{\left\{ 1 + \frac{1}{3} \left( \frac{\theta}{\tau} \right) \right\}}{K} \left( \frac{\tau}{\theta} \right)$	$\frac{\left\{ 0,9 + \frac{1}{12} \left( \frac{\theta}{\tau} \right) \right\}}{K} \left( \frac{\tau}{\theta} \right)$	$\frac{\left\{ \frac{4}{3} + \frac{1}{4} \left( \frac{\theta}{\tau} \right) \right\}}{K} \left( \frac{\tau}{\theta} \right)$
$\tau_I$	—	$3,33 \times \theta$	$2,0 \times \theta$	—	$\left[ \frac{30 + 3 \left( \frac{\theta}{\tau} \right)}{9 + 20 \left( \frac{\theta}{\tau} \right)} \right] \times \theta$	$\left[ \frac{32 + 6 \left( \frac{\theta}{\tau} \right)}{13 + 8 \left( \frac{\theta}{\tau} \right)} \right] \times \theta$
$\tau_D$	—	—	$0,5 \times \theta$	—	—	$\left[ \frac{4}{11 + 2 \left( \frac{\theta}{\tau} \right)} \right] \times \theta$

### 2.3.1.3 ITAE ("Integrated Time-Weighted Absolute Error")

A integral do erro absoluto ponderado no tempo ( $ITAE = \int_0^{\infty} t |e(t)| dt$ ) é o critério desempenho que melhor concilia propriedades de desempenho e robustez. Smith et al. [OR94] determinaram os parâmetros de controladores P, PI e PID que minimizam *ITAE* para diferentes valores de  $\theta/\tau$ . A forma da função erro depende tanto do tipo de perturbação quanto da localização onde ela ocorre na malha de controle. Smith et al. desenvolveram expressões tanto para degrau no setpoint quanto no distúrbio na carga as quais são listadas na Tabela 2.

**Tabela 2:** Métodos ITAE para Setpoints e Distúrbios na Carga

	ITAE-Setpoint		ITAE - Distúrbio Carga		
	PI	PID	P	PI	PID
$K_P$	$\frac{0,586}{K} \left( \frac{\tau}{\theta} \right)^{0,916}$	$\frac{0,965}{K} \left( \frac{\tau}{\theta} \right)^{0,855}$	$\frac{0,49}{K} \left( \frac{\tau}{\theta} \right)^{1,084}$	$\frac{0,859}{K} \left( \frac{\tau}{\theta} \right)^{0,977}$	$\frac{1,357}{K} \left( \frac{\tau}{\theta} \right)^{0,947}$
$\tau_I$	$\frac{\tau}{\left[ 1,03 - 0,165 \left( \frac{\theta}{\tau} \right) \right]}$	$\frac{\tau}{\left[ 0,796 - 0,147 \left( \frac{\theta}{\tau} \right) \right]}$	—	$\frac{\tau}{0,674} \left( \frac{\theta}{\tau} \right)^{0,680}$	$\frac{\tau}{0,842} \left( \frac{\theta}{\tau} \right)^{0,738}$
$\tau_D$	—	$0,308 \tau \left( \frac{\theta}{\tau} \right)^{0,929}$	—	—	$0,381 \tau \left( \frac{\theta}{\tau} \right)^{0,995}$

### 2.3.1.4 IMC e Método da Síntese Direta

Utilizando-se uma função de primeira ordem com tempo morto como modelo pode-se utilizar a estrutura de controle IMC ("Internal Model Control") para projetar controladores PI e PID. Em adição ao modelo, o usuário tem que selecionar a constante de tempo do filtro de primeira ordem ( $\lambda$ ), a qual pode ser vista como sendo a velocidade que o sistema deverá ter em malha fechada consistindo dessa forma como sendo um parâmetro que estabelece o compromisso entre robustez e desempenho. Quanto menor for  $\lambda$  mais rápida será a resposta, portanto maior será desempenho, mas mais sensível a incertezas (isto é, a qualidade da aproximação) será a resposta. A Tabela 3 apresenta os parâmetros do controlador PIDs ajustados por esse método. Observe que nessa Tabela encontrasse valores recomendados para  $\lambda$ , os quais representam um bom compromisso entre desempenho e robustez. Outro método de ajuste que leva a resultados semelhantes ao IMC é o da síntese direta. Nesse método o controlador é calculado de tal forma que uma determinada resposta em malha fechada seja alcançada. Note que os parâmetros do controlador gerados por esse método são praticamente os mesmos que os obtidos pelo método IMC.

**Tabela 3:** Síntese Direta e IMC

	Síntese Direta		IMC		
	PI	PID	PI	PI melhorado	PID
$K_P$	$\frac{\tau}{K(\lambda + \theta)}$	$\frac{2\tau + \theta}{2K(\lambda + \theta)}$	$\frac{\tau}{\lambda K}$	$\frac{2\tau + \theta}{2\lambda K}$	$\frac{2\tau + \theta}{2K(\lambda + \theta)}$
$\tau_I$	$\tau$	$\tau + \frac{\theta}{2}$	$\tau$	$\tau + \frac{\theta}{2}$	$\tau + \frac{\theta}{2}$
$\tau_D$	—	$\frac{\tau\theta}{2\tau + \theta}$	—	—	$\frac{\tau\theta}{2\tau + \theta}$
	Filtro adicional	$\frac{\theta\lambda}{2(\lambda + \theta)}$	$\frac{\lambda}{\theta} > 1,7$	$\frac{\lambda}{\theta} > 1,7$	$\frac{\lambda}{\theta} > 0,25$

### 2.3.1.5 Método de Tau

Uma considerável evolução dos métodos simples de ajuste de controladores foi proposta por • ström & Hägglund [AH95], o qual foi denominado pelos autores de método de Kappa-Tau. O nome se deve a parametrização adicional definida que é utilizada para ajustar o resultado do projeto dos controladores obtido por métodos mais sofisticados de ajuste, os quais utilizam a máxima sensibilidade como critério de ajuste. Quando a parametrização Tau é utilizada, as características da resposta degrau e tomada como base de ajuste dos parâmetros do controlador. Nesse caso utiliza-se como parâmetro adicional de ajuste o tempo morto normalizado, chamado pelos autores de Tau, o qual é definido por:

$$Tau = \frac{\theta}{\theta + \tau} \quad (3)$$

A Tabela 4 apresenta a título de ilustração os ajuste para o caso de um controlador PID e sistemas estáveis. Em [AH95], o leitor poderá encontrar tabelas similares para controladores PI e PI e PID para sistemas integrantes (sistemas com pólo na origem).

**Tabela 4:** Fórmulas para o ajuste de controladores PID baseadas no método Kappa-Tau utilizando-se a parametrização Tau. A Tabela apresenta os parâmetros da função:  $a_0 \times \exp(a_1 \text{Tau} + a_2 \text{Tau}^2)$ .

$\text{Tau} = \theta/(\theta + \tau)$	$M_S = 1.4$ (Projeto p/ Robustez)			$M_S = 2.0$ (Projeto p/ Perform.)		
$a_0 \times e^{(a_1 \text{Tau} + a_2 \text{Tau}^2)}$	$a_0$	$a_1$	$a_2$	$a_0$	$a_1$	$a_2$
$K_p \times K(\theta/\tau)$	3.8	-8.4	7.3	8.4	-9.6	9.8
$\tau_I/\theta$	5.2	-2.5	-1.4	3.2	-1.5	-0.93
$\tau_D/\tau$	0.48	2.8	-2.1	0.28	3.8	-1.6
$\tau_D/\theta$	0.89	-0.37	-4.1	0.86	-1.9	-0.44
$\tau_D/\tau$	0.077	5.0	-4.8	0.076	3.4	-1.1
$b$	0.40	0.18	2.8	0.22	0.65	0.051

### 2.3.2 Métodos baseados em $K_u$ e $P_u$

Por darem informações sobre as características dinâmicas justamente na região onde a maioria dos controladores *feedback* normalmente trabalham, ou seja, a região de corte, o ganho ultimoativo ( $K_u$ ) e o período ultimoativo ( $P_u$ ) são informações extremamente importantes sobre a dinâmica dos sistemas. Esses valores podem ser facilmente calculados para qualquer função de transferência e poderão ser então aplicados para projetar controladores para sistemas com uma dinâmica de elevada ordem, não necessitando que se faça nenhuma aproximação como feita nos métodos anteriores. O PID-TB possui implementados dois métodos baseados em  $K_u$  e  $P_u$ : Método de Ziegler-Nichols e o Método de Kappa-Tau (com a parametrização Kappa).

#### 2.3.2.1 Ziegler-Nichols baseado em $K_u$ e $P_u$

O ganho ultimoativo é o maior valor que um controlador puramente proporcional poderá ter antes de instabilizar a malha de controle, ou seja, é o ganho que leva os pólos em malha fechada do sistema até a fronteira de estabilidade transformando o sistema em marginalmente estável. Ziegler-Nichols em seu famoso artigo tiveram a visão de utilizar esse valor de ganho ( $K_u$ ), juntamente com o período a ele associado ( $P_u$ ), como base para um método de ajuste de controlador que revolucionou a sua época. Ziegler-Nichols relacionaram o ganho do controlador com uma fração de  $K_u$  e as constantes  $\tau_I$  e  $\tau_D$  com uma fração de  $P_u$ . A fração foi determinada de tal forma que a resposta em malha fechada tivesse uma razão de decaimento de  $1/4$ . A Tabela 5 apresenta o método de ZN.

#### 2.3.2.2 Método de Kappa

- ström & Hägglund [AH95] propuseram uma melhoria no método originalmente proposto por Ziegler-Nichols, o qual, como já mencionamos, foi denominado pelos autores de método de Kappa-Tau. Na seção 2.3.1.5 vimos a versão desse método segundo a parametrização Tau. A versão baseada em  $K_u$  e  $P_u$  foi parametrizada pelos autores através do parâmetro Kappa definido por:

$$\kappa = \frac{1}{K_u K} \quad (4)$$

A Tabela 5 apresenta a título de ilustração os ajuste para o caso de um controlador PID e sistemas estáveis. Em [AH95], o leitor poderá encontrar a tabela equivalente para controladores PI.

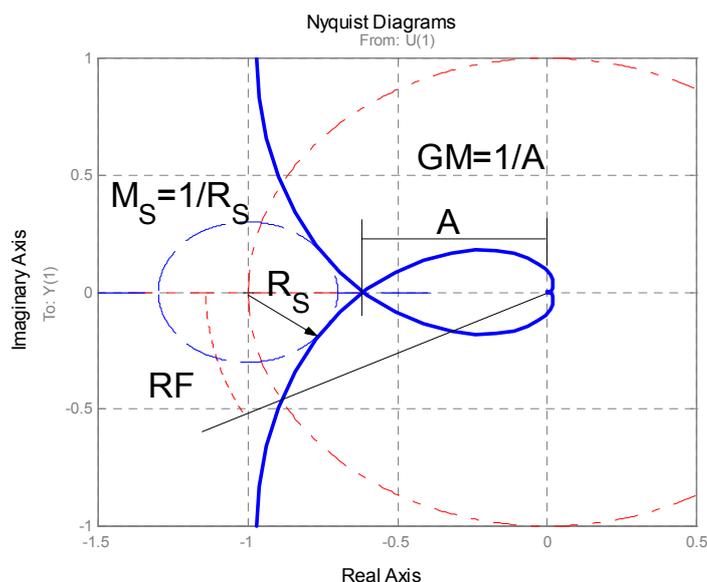
**Tabela 5:** Fórmulas para o ajuste de controladores PID para os métodos de Ziegler-Nichols e Kappa-Tau utilizando-se a parametrização Kappa. Para o método de Kappa-Tau a tabela apresenta os parâmetros da função:  $a_0 \times \exp(a_1\kappa + a_2\kappa^2)$  onde  $\kappa = 1/(K_u K)$ .

	Ziegler- Nichols ( $K_u, P_u$ )			Kappa PID p/ $M_S=1.4$			Kappa PID p/ $M_S=2.0$		
	P	PI	PID	$a_0$	$a_1$	$a_2$	$a_0$	$a_1$	$a_2$
$K_P/K_u$	0,5	0,45	0,6	0.33	-0.31	-1.0	0.72	-1.6	1.2
$\tau_I/P_u$	—	1/1,2	1/2	0.76	-1.6	-0.36	0.59	-1.3	-0.38
$\tau_D/P_u$	—	—	1/8	0.17	-0.46	-2.1	0.15	-1.4	0.56
$b$	1	1	1	0.58	-1.3	3.5	0.25	0.56	-0.12

## 2.4 Critérios de Análise de Robustez e Desempenho

### 2.4.1 Critérios de Robustez

Os processos industriais são não lineares e dinâmicas rápidas são normalmente desconsideradas durante o projeto do controlador. Esses e outros fatores fazem com que os modelos lineares invariantes no tempo (LTI) sejam apenas, como qualquer outro modelo, uma aproximação da realidade. Essa aproximação normalmente é suficiente para o projeto do controlador, se uma certa margem de segurança for garantida durante o projeto. A margem de ganho, a reserva de fase e a máxima sensibilidade são típicas medidas dessa margem de segurança. A figura 4 mostra esquematicamente em um diagrama de Nyquist como essas grandezas podem ser determinadas.



**Figura 4:** Representação da reserva de fase (RF), margem de ganho (GM) e máxima sensibilidade ( $M_S$ )

Normalmente reserva de fase (RF) e margem de ganho (GM) inferiores a  $30^\circ$  e 1,7 são indicadores de projetos muito sensíveis a incertezas, ou seja, a qualidade do modelo utilizado, e, por isso, são chamados de pouco robustos. A margem de ganho é uma medida de quanto o sistema (controlador+planta) suportaria uma variação no ganho estático, enquanto que a reserva de fase pode ser interpretada como tempo morto suportável pela malha antes de instabilizar. Esse tempo morto pode ser interpretado em termos práticos como sendo uma medida das implicações em termos de estabilidade em se desconsiderar dinâmicas rápidas, tais como: dinâmica das válvulas e dos medidores, durante o projeto do controlador. Embora RF e GM sejam boas medidas da robustez do projeto, não são suficientes para uma completa análise, uma vez que valores de RF e GM dentro da faixa recomendável não garantem que o diagrama de Nyquist do controlador+planta não passe muito perto do ponto  $-1$ . Pontos próximos a  $-1$  indicam uma elevada tendência a instabilização. Portanto, a medida mais conclusiva da robustez é a Máxima Sensibilidade ( $M_S$ ) a qual é definida como sendo o inverso da menor das distâncias do ponto  $-1$ , ou seja, o ponto mais sensível e fácil de ser instabilizado. O método de Kappa-Tau faz uso de  $M_S$  como critério de projeto. Duas são as versões disponíveis:  $M_S=1,4$  recomendado para projetos mais robustos e  $M_S=2$  para projetos com maior desempenho. Um bom projeto de controlador terá que ter os valores de  $M_S$  no seguinte intervalo:

$$1,2 \text{ (muito robusto, menor desempenho)} < M_S < 2,2 \text{ (pouco robusto, maior desempenho)}$$

O PID-TB já calcula automaticamente os valores de GM, RF e  $M_S$  para todos os controladores projetados.

#### 2.4.2 Critérios de desempenho

Um bom projeto de controlador deve conciliar boas características de robustez com bom desempenho. Na seção anterior vimos como podemos quantificar facilmente a robustez do projeto através de índices de robustez, tais como: GM, RF e  $M_S$ . Naturalmente, o desempenho pode ser inferido através da simples visualização gráfica das simulações geradas para distúrbios específicos, dos quais se destacam: a mudança de setpoint (características servo) e distúrbios na carga (características regulatórias). Embora de fácil e intuitivo entendimento, a comparação de simulações além de ser um tanto quanto subjetiva, são de difícil documentação, uma vez que requerem um considerável espaço para o seu armazenamento. Portanto, se faz mister que se defina índices que permitam o fácil acompanhamento do desempenho do sistema.

O PID-TB através da Integral do Erro Quadrático (*Integrated Square Error, ISE*) e da Integral do Erro Absoluto (*Integrated Absolute Error, IAE*) permite o fácil acompanhamento do desempenho do sistema. Esses índices de desempenho são definidos pelas seguintes expressões:

$$\text{Integral do Erro Quadrático} \quad ISE = \int_0^T e^2(t) dt \quad (5)$$

$$\text{Integral do Erro Absoluto} \quad IAE = \int_0^T |e(t)| dt \quad (6)$$

O limite  $T$  das integrais corresponde ao tempo de simulação selecionado pelo usuário. Naturalmente, quanto menores forem os valores ISE e IAE maior será o desempenho do sistema. A ISE, pelo fato de elevar ao quadrado o erro, tem o seu valor mais penalizados pelo maiores valores de erro, enquanto que IAE não faz nenhuma diferenciação no tamanho relativo dos erros.

### 3 IMPLEMENTAÇÃO

A implementação do *PID Controller Design Toolbox* fundamentou-se em alguns preceitos básicos, os quais surgiram da observação prática de outros *toolboxes* disponíveis no *Matlab*. Alguns aspectos considerados positivos foram mantidos ou então aprimorados, porém os demais foram reformulados, o que implicou em um desafio maior em termos do algoritmo e da programação, resultando em um produto final de excelente qualidade.

A seguir serão listados alguns dos conceitos que nortearam a programação:

- Interface gráfica com disposição seqüencial e auto-explicativa:  
Uma interface gráfica seqüencial significa uma interface com os objetos dispostos de maneira tal a seguir a seqüência lógica na elaboração de um projeto de controlador; enquanto que o fato de ser auto-explicativa, auxilia o usuário na tomada de decisões.
- Opções desabilitadas enquanto não se dispõe de informações suficientes:  
Quando não se dispõe de informações suficientes, as opções se encontram desabilitadas, desta forma o usuário é obrigado a seguir a seqüência lógica de um projeto. Também pode ocorrer o caso de não haver opções específicas para uma determinada ação, neste caso, não havendo a necessidade da mesma, ela é desabilitada, fazendo com que o usuário não precise ficar pensando se deve ou não informá-la.
- Flexibilidade de inserção de modelos definidos pelo usuário:  
O *PID Controller Design Toolbox* não é uma ferramenta rígida, permitindo que o usuário adicione modelos próprios, tanto para a simulação (modelos não lineares), quanto para a estrutura do controlador que será utilizado na simulação.
- *Help On-line*:  
O *Help On-line* serve para esclarecer dúvidas sobre a utilização do *PID Controller Design Toolbox*, atuando também como elemento didático, de tal maneira que o usuário vai ampliando e consolidando seus conhecimentos a medida que o projeto do controlador vai avançando.
- Visualização e manipulação dos resultados:  
O *PID Controller Design Toolbox* permite a elaboração de diversos projetos de controladores do tipo **P** (Proporcional), **PI** (Proporcional Integral) e **PID** (Proporcional Integral Derivativo) ao mesmo tempo, utilizando os diversos métodos de ajuste disponíveis. Assim, tem-se como resultado final, um conjunto de controladores que podem ser comparados entre si para a seleção do melhor para uma posterior implementação. Para facilitar a comparação dos diversos controladores projetados, existe a opção **Show Table** que gera uma tabela com os controladores projetados para um dado modelo. Nesta tabela existem índices que permitem avaliar a qualidade dos controladores e desta forma ordená-los segundo um dos índices escolhidos.
- Documentação através do programa *MS Excel*:  
Depois da ordenação dos controladores realizada na tabela, os resultados podem ser exportados diretamente para o programa *MS Excel* para sua documentação.  
A documentação dos resultados é um dos aspectos que muitos dos *toolboxes* disponíveis no *Matlab* não possuem, ficando apenas limitados a etapa de cálculo e no máximo apresentação dos resultados, mas em ambiente *Matlab*; enquanto que muitas vezes, o trabalho não se limita a esta etapa, obrigando o usuário a desenvolver programas que façam a comunicação com programas capazes de documentar os resultados obtidos, ou então utilizar a velha prática (mas muito útil) *Ctrl+C* e *Ctrl+V*.
- A não utilização do *Workspace* para armazenamento de variáveis globais:  
Este é outro aspecto que muitos *toolboxes* disponíveis no *Matlab* não possuem, fazendo com que o *Workspace* fique repleto de informações desnecessárias. A não utilização do *Workspace*

para armazenamento de variáveis globais implica na sua manutenção apenas com as informações que o usuário deseja que estejam disponíveis nele. Dessa forma, o *Workspace* não fica "poluído", o que muitas vezes acaba dificultando análises posteriores que se deseja realizar na linha de comando do *Matlab*.

### 3.1 Funções utilizadas para o Projeto de Controladores

Todos os métodos apresentados nesse trabalho estão implementados no PID-Toolbox. O usuário poderá utilizar as funções tanto através de comandos de linha quanto através de uma GUI especialmente desenvolvida para facilitar a documentação do trabalho realizado. A lista das principais funções disponíveis no PID-TB podem ser obtidas através do comando *help contents* no MATLAB.

```
» help contents

Universidade Federal do Rio Grande do Sul (UFRGS) - RS - Brasil
MATLAB PID Controller Design Toolbox
Version 5.3.1 (R11.1) Jul-2000

Main Command
Pid_gui          - Open the PID Controller Design Toolbox.

PID Tuning Methods
ZN_PID          - Ziegler-Nichols Tuning Method.
CC_PID          - Cohen-Coon Tuning Method.
IMC_PID         - IMC-PID Tuning Method.
ITAE_PID        - ITAE-PID Tuning Method.
KT_PID          - Kappa-Tau PID Tuning Method.

General Purpose Functions
Bodedt          - Bode frequency response for continuous-time linear
                 systems.
lNyquist        - Nyquist frequency response for continuous-time linear
                 systems.
Export2xls      - Export table to MS-Excel.
Overdamp        - Determine an overdamped TF using the [RK94] method.
Raiz            - Calculates the systems roots.
Met_tan         - Tangent method approach.
Set_Time        - Calculates the settling time.
Trapezio        - Find the area under the curve y = f(x) using the
                 trapezoidal rule.
W_U             - Calculate the ultimate frequency of the second order
                 with dead time
                 transfer function.
W_U_Zn          - Calculate the ultimate frequency of the first order with
                 dead time
                 transfer function.
SisoIdent       - Open the SISO identification interface.
```

As funções utilizadas para o ajuste dos controladores são genericamente denominadas *XXX\_pid*, onde *XXX* corresponde ao nome do método utilizado, o qual poderá ser IMC, ITAE, ZN (Ziegler-Nichols), CC (Cohen-Coon) e KT (Kappa-Tau). A título de ilustração vamos mostrar a

seguir o help da função ITAE\_PID. Todas as demais funções de ajuste foram construídas e são acionadas de forma similar.

```
» help ITAE_pid

[Kp,Ti,Td]=ITAE_PID(num,den,teta,d_s)
[Kp,Ti,Td]=ITAE_PID(K,tau,teta,d_s)
[Kp,Ti,Td]=ITAE_PID(t,y,d_s)

ITAE PID Tuning Method

Output Parameters:
-----
Kp -> proportional constant
Ti -> Integration Time (Reset Time)
Td -> Derivative Time

PI or PID-tuning method is automatically selected based on
the number of output parameters. Use
[Kp,Ti]=ITAE_pid(...) for PI-Tuning Settings [OR94,Tab.15.4]
[Kp,Ti,Td]=ITAE_pid(...) for PID-Tuning Settings [OR94,Tab.15.4]

Input Parameters:
-----
num-> numerator, k, or t
den-> denominator, tau, or y
teta-> pure time delay
d_s -> 'd'-> setting for disturbance (default)
      's'-> setting for set point

Examples:
[Kp,Ti,Td]=ITAE_pid(t,y,'s') t-> time y-> system's step response
[Kp,Ti,Td]=ITAE_pid(num,den,teta) G(s)=num(s)/den(s)*exp(-teta*s)
[Kp,Ti,Td]=ITAE_pid(k,tau,teta) G(s)=k/(tau*s+1)*exp(-teta*s)

SEE: [OR94, Tables 15.4]
```

## 3.2 Interface Gráfica com o usuário

A interface gráfica com o usuário é a responsável pela fácil utilização do *PID-TB*. Com ela é possível gerenciar uma série de ações conjuntas de maneira simples e direta, poupando um enorme tempo no projeto de controladores do tipo *P*, *PI*, e *PID*. Além disto, ainda existe o apelo didático, o qual induz o usuário a seguir as etapas lógicas de um projeto de controlador, recebendo auxílio de diversas maneiras, como no caso do *Help On-line*, por exemplo.

Existe uma série de janelas que compõem o *PID-TB*, todas gerenciadas a partir da janela principal. A seguir será apresentado um exemplo prático que permitirá percorrer as diversas etapas de um projeto de controlador auxiliada pela interface gráfica com o usuário.

### 3.2.1 Janela principal

A interface gráfica com usuário do *PID-TB* é acionado pelo comando `pid_gui` na linha de comando do *MATLAB*. A figura 5 mostra a janela principal do *Toolbox*. O primeiro passo para começar o ajuste do controlador é a importação dos modelos LTI, com os quais se deseja trabalhar. Tais modelos devem ser gerados no *Workspace* do *MATLAB* para que seja possível importá-los.

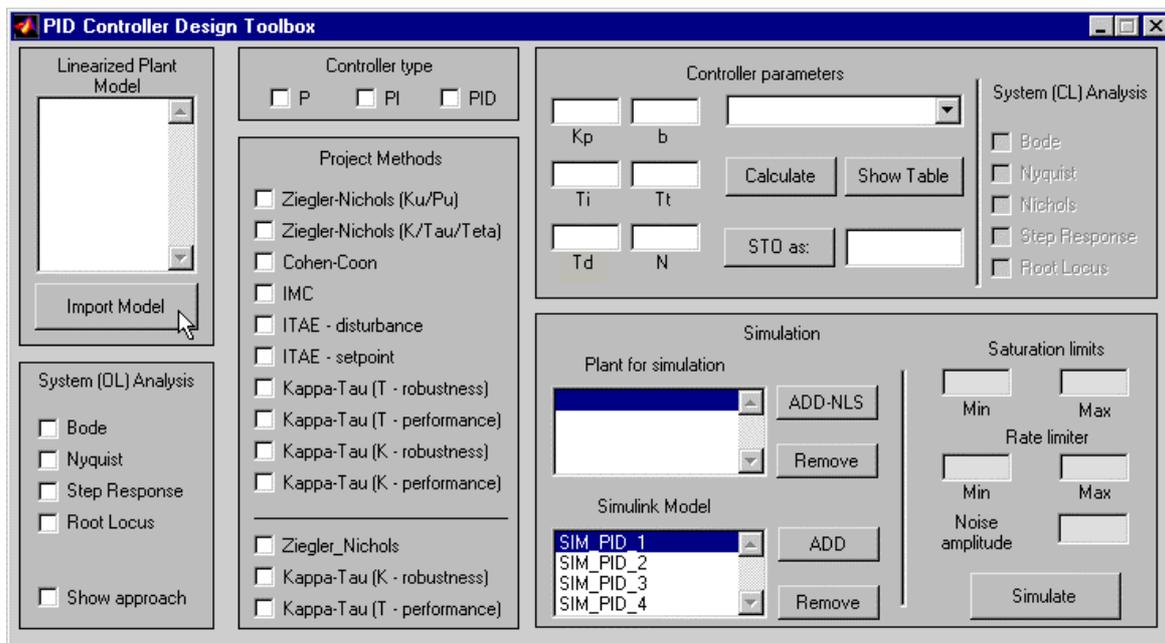


Figura 5: Janela principal do PID-TB

### 3.2.2 Importação e identificação de modelos

A importação é feita pressionando-se o botão *Import Model* dentro do *frame Linearized Plant Model* da janela principal. Neste momento abre-se a janela mostrada na figura 6 onde o usuário é solicitado a escolher dentre os modelos LTI disponíveis no *Workspace* quais ele deseja trabalhar. Existe ainda a possibilidade de se utilizar um conjunto de dados provenientes de um processo qualquer para a identificação do respectivo modelo. Para tanto, seleciona-se o *radio-button Data to Identification*, como é mostrado na figura 7.

Os modelos LTI's disponíveis no *Workspace* do MATLAB são apresentados no campo *Available data*, os quais são selecionados através dos botões com setas indicativas para o campo *Selected data*. Ao se pressionar *OK* os modelos selecionados são automaticamente importados para a janela principal.

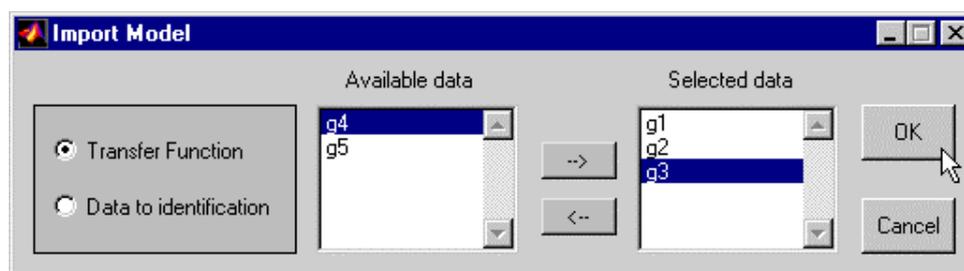


Figura 6: Janela para importar modelos LTI do *Workspace* do MATLAB.

No caso de se dispor de um conjunto de dados de um processo, eles serão importados para identificação selecionando o *radio button* correspondente. Ao ser pressionado o botão *OK*, a janela de importação de dados para identificação é automaticamente aberta já com os valores padrões sugeridos nos campos de entrada de dados.

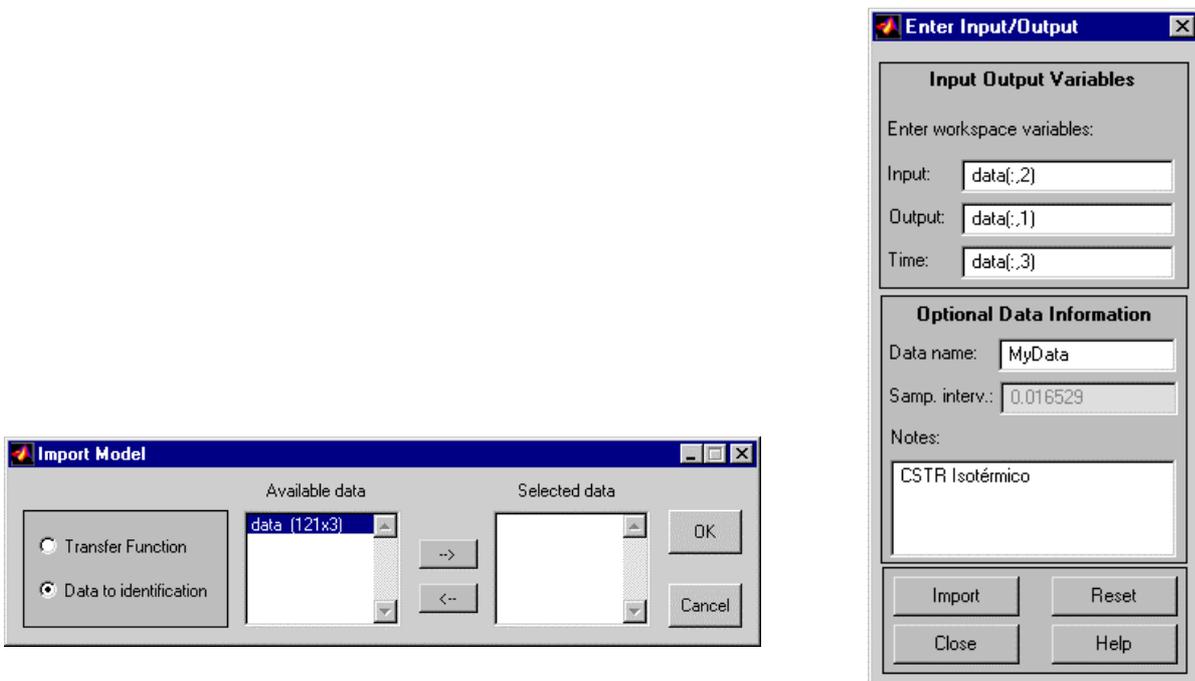


Figura 7: Janelas para importar dados do *Workspace* do *MATLAB* para identificação.

A figura 8 a seguir apresenta a janela para identificação de um ou mais modelos que representem de forma satisfatória o conjunto de dados provenientes do processo que se deseja projetar um controlador.

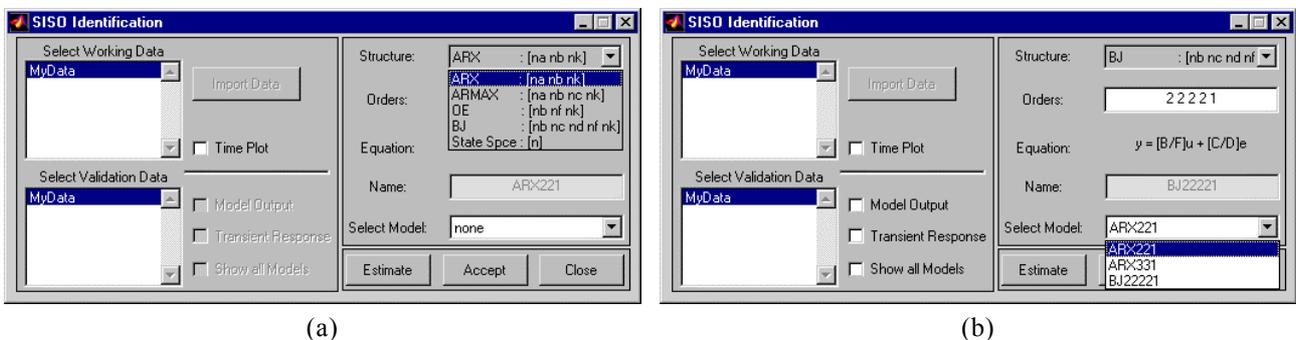


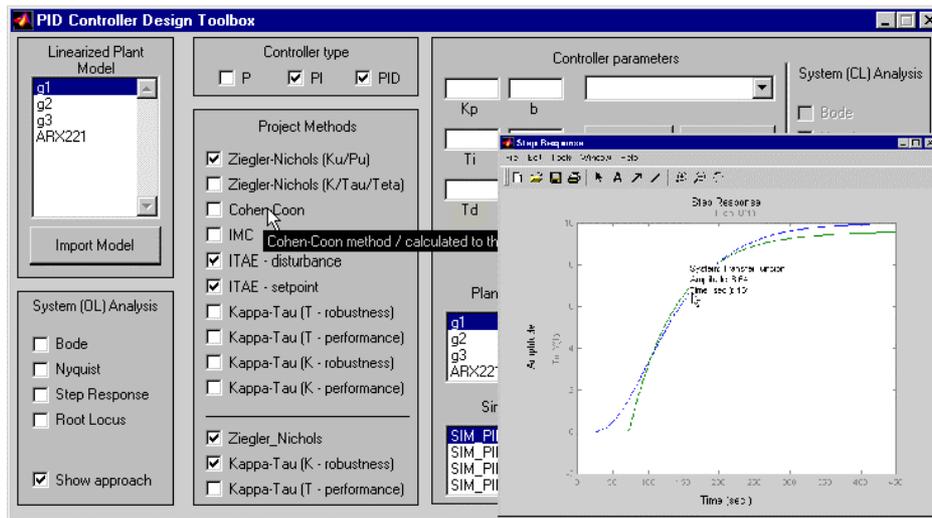
Figura 8: Janela para identificação dos modelos pertinentes ao conjunto de dados importado.

Para se proceder com a identificação, basta selecionar o conjunto de dados a ser utilizado e um dos diversos tipos de modelos disponíveis (figura 8(a)), escolhendo a ordem que se deseja para o mesmo e depois pressionar o botão *Estimate*. Os modelos estimados vão sendo armazenados em uma lista na forma de *popup-menu*, como visto na figura 8(b). Algumas análises sobre os modelos estimados podem ser feitas, como por exemplo *Model Output*. Para tanto, deve-se selecionar um conjunto de dados de validação no campo *Select Validation Data*. Com os resultados obtidos destas análises, pode-se selecionar os modelos a serem utilizados, exportando-os para a janela principal do *PID\_TB* através do botão *Accept*.

### 3.2.3 Pré-análise, projeto dos controladores

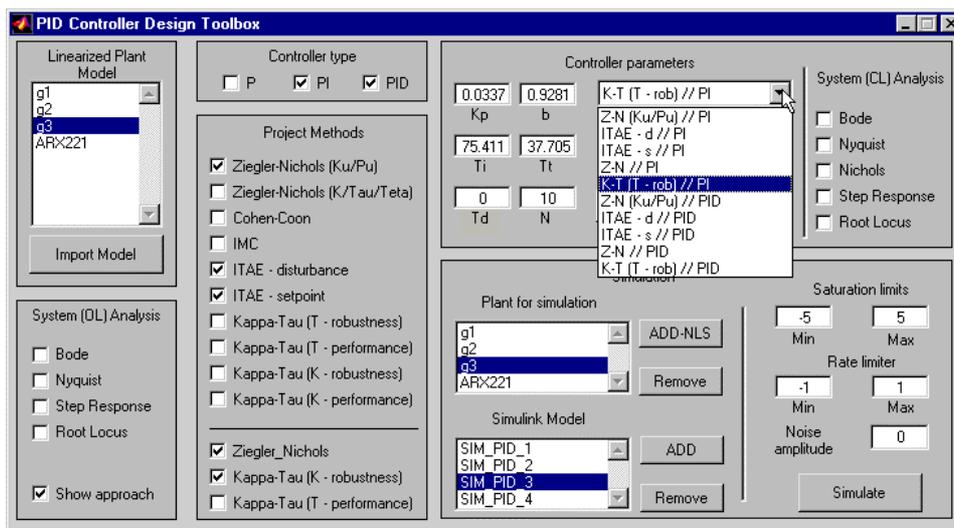
Após importar os modelos que se deseja utilizar, pode-se ter uma idéia do seu comportamento através de uma pré-análise do modelo, ou seja, do modelo quando não existe nenhum controlador

atuando sobre ele. Para se proceder com uma análise do modelo, basta selecionar o tipo de estudo desejado clicando nos respectivos *check-boxes*. A figura 9 apresenta a resposta do modelo selecionado  $g_3$  a uma perturbação do tipo degrau unitário, juntamente com a resposta ao mesmo sinal para um modelo aproximado de  $g_3$ , o qual será utilizado por alguns dos métodos de ajuste de controladores.



**Figura 9:** Seleção do modelo, pré-análise através de uma resposta do tipo degrau unitário, seleção dos controladores que serão projetados e os respectivos métodos a serem utilizados.

Após selecionado o modelo a ser utilizado, os tipos de controladores a serem projetados e os seus métodos, pressiona-se o botão *Calculate* do *frame Controller Parameters* da janela principal. Desta forma, serão calculados todos os parâmetros para todos os tipos de controladores utilizando os métodos selecionados. Estes resultados são armazenados em uma lista do tipo *popup-menu* e os parâmetros correspondentes ao elemento selecionado desta lista é apresentado ao lado. A figura 10 a seguir mostra o que os resultados após o cálculo dos parâmetros dos controladores.



**Figura 10:** Resultados obtidos através do cálculo dos parâmetros dos controladores.

Os parâmetros dos controladores calculados podem ser alterados manualmente e salvos como parâmetros definidos pelo usuário para testes posteriores.

### 3.2.4 *Simulação e análise dos controladores em malha fechada*

Para a simulação dos controladores projetados, deve-se selecionar um modelo no campo *Plant for Simulation*, que não precisa necessariamente ser o modelo utilizado no projeto do controlador, podendo inclusive ser um modelo não linear do processo feito no *Simulink*. Depois de selecionada a planta, deve-se ainda escolher qual tipo de modelo de controlador será utilizado. Existem quatro modelos do *Simulink* já propostos, permitindo ainda que o usuário adicione seus próprios modelos personalizados. Os modelos propostos são os seguintes:

- SIM\_PID\_1: PID implementado apenas com *anti-Windup*.
- SIM\_PID\_2: PID implementado com *anti-Windup*, limites de saturação e na variação das ações de controle.
- SIM\_PID\_3: PID implementado com *anti-Windup*, limites de saturação e na variação das ações de controle e com distúrbios.
- SIM\_PID\_4: Análogo ao anterior, porém ainda com filtros nos distúrbios e nas ações de controle.

Nestes modelos, a planta é submetida a uma variação no *setpoint* e um distúrbio na carga. Desta forma pode-se analisar a efetividade do controlador para estes dois casos típicos de variações. A figura 11, a seguir, apresenta o esquema do controlador SIM\_PID\_3 implementado no *Simulink*, enquanto que a figura 12 a implementação do *anti-Windup* nos controladores PID's utilizados.

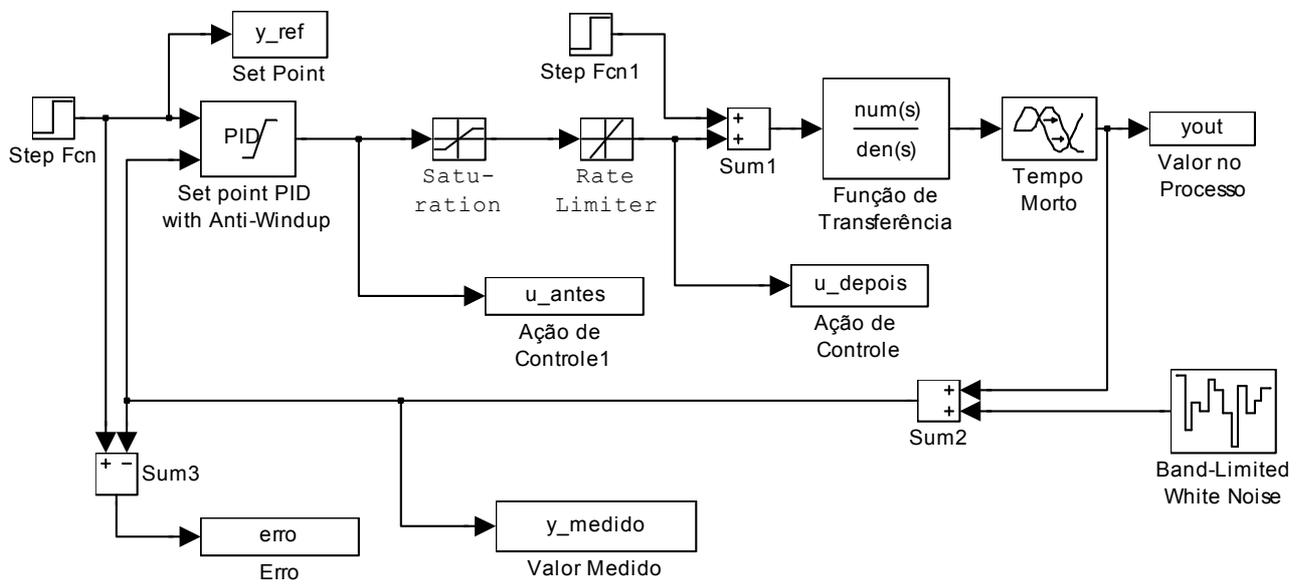
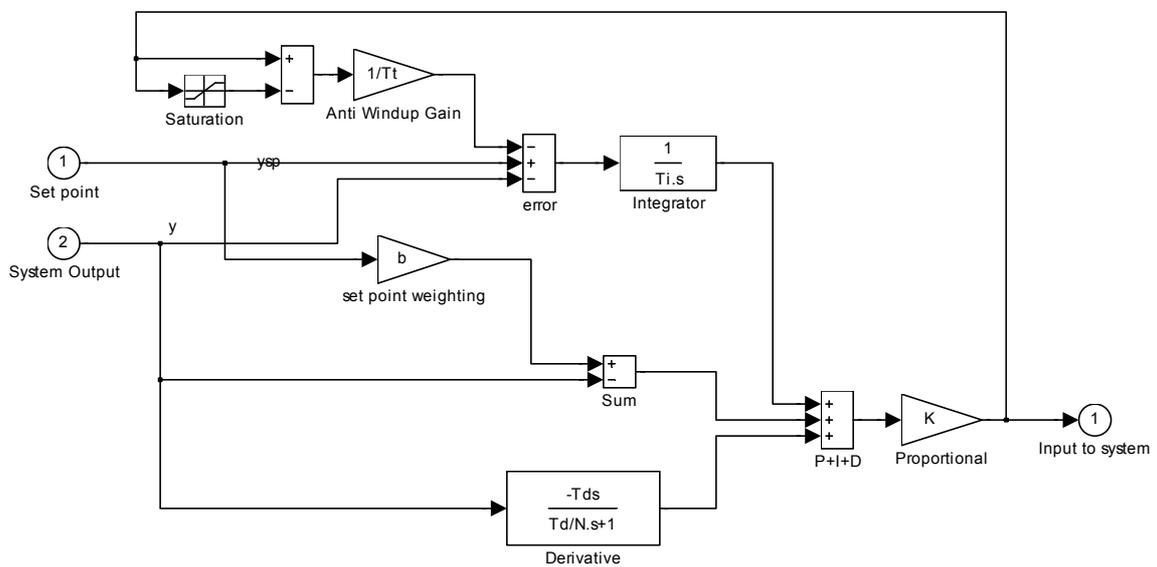


Figura 11: Modelo SIM\_PID\_3 implementado no *Simulink*.

PID controller with set point weighting and anti-windup.



**Figura 12:** Implementação do *anti-Windup* no PID.

Os resultados das simulações podem ser observados nas figuras 13 e 14 a seguir.

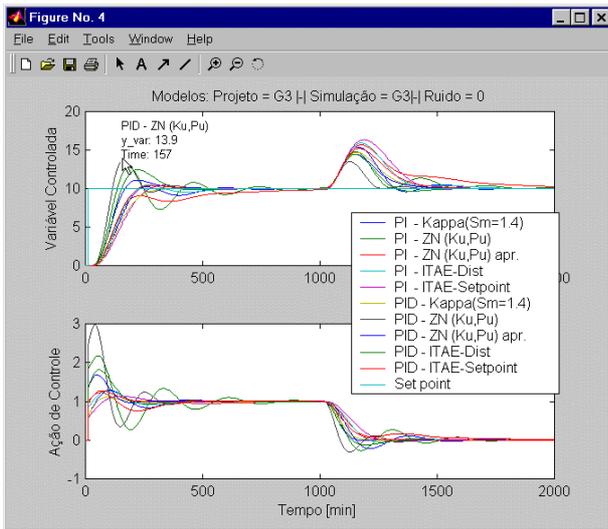


Figura 13: Simulação do modelo  $g_3$  com os controladores projetados.

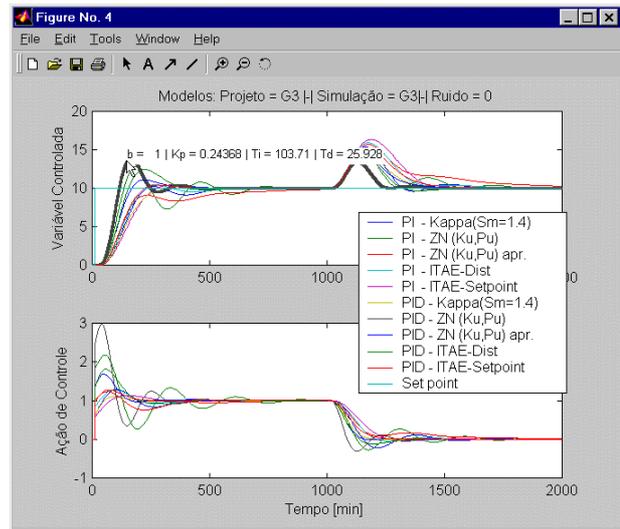


Figura 14: Simulação do modelo  $g_3$  com os controladores projetados.

Os resultados das simulações compõem-se das respostas do sistema (Variável Controlada) na porção superior da figura enquanto que na porção inferior pode-se observar as ações de controle (Ação de Controle). Para facilitar a visualização e tornar mais rápida a interpretação dos resultados, ao ser selecionada uma das linhas com o botão esquerdo são mostradas informações pertinentes à mesma, como por exemplo o tipo de método de ajuste utilizado para projetar aquele controlador (cf. figura 13). No caso de se utilizar o botão direito, informações dos parâmetros do controlador utilizado são mostradas ao mesmo tempo em que a linha selecionada é destacada das demais (cf. figura 14).

### 3.2.5 Tabela de resultados e exportação para o EXCEL

Para proporcionar uma comparação sistemática entre os diversos métodos de ajuste de controladores, pode-se utilizar o recurso *Show Table*, o qual monta uma tabela com os parâmetros dos controladores projetados bem como índices que permitem avaliar a qualidade dos mesmos. A figura 15 a seguir apresenta uma tabela gerada para os controladores projetados anteriormente.

Methods	Kp	Ti	Td	b	N	Tt	GM	PM	MS	IAE	ISE
Kappa(Sm=1.4)	0.1168	104.01	0	0.5206	52.007	10	2.3837	48.224	2.0563	2430.4	10787.
ZN (Ku,Pu)	0.1827	172.85	0	1	86.425	10	1.8161	38.536	2.3399	3489.2	15392
ZN (Ku,Pu) apr.	0.1012	184.63	0	1	92.318	10	3.3327	81.092	1.5451	2574.8	14589
ITAE-Dist	0.0836	105.02	0	1	52.511	10	3.3472	63.978	1.6390	2885.1	17464.
ITAE-Setpoint	0.0572	79.183	0	1	39.591	10	4.1421	62.841	1.5150	2289.9	13439.
Kappa(Sm=1.4)	0.1168	104.01	27.721	0.5206	52.007	10	4.6825	66.953	1.4206	1544.2	8269.9
ZN (Ku,Pu)	0.2436	103.71	25.927	1	51.855	10	2.2583	40.795	2.0797	1943.8	10498.
ZN (Ku,Pu) apr.	0.1349	110.78	27.695	1	55.391	10	4.0630	66.287	1.4974	2056.3	10787.

Figura 15: Tabela com os parâmetros dos controladores projetados seus respectivos índices de análise.

Nesta tabela os dados podem ser reordenados segundo algum dos índices disponíveis apenas clicando sobre a respectiva coluna. Após a disposição a contento do usuário, os dados podem ser exportados para uma planilha do *MS Excel* como é mostrado na figura 16.

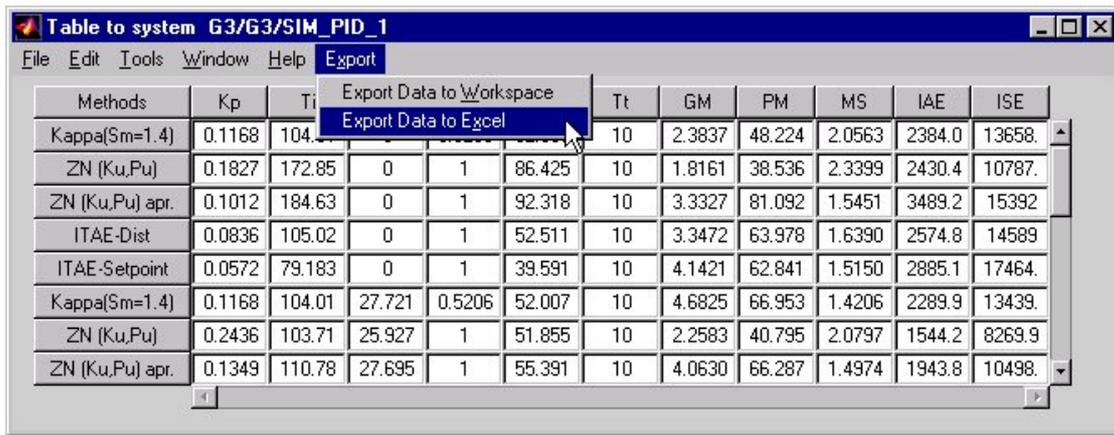


Figura 16: Exportando dados da tabela (ambiente *MATLAB*) para o *MS Excel*.

A figura 17 mostra os dados já exportados para uma planilha do programa *MS Excel*.

	A	B	C	D	E	F	G	H	I	J	K
1	G3/G3/SIM_PID_1										
2	Methods	Kp	Ti	Td	b	Tt	N	GM	PM	MS	IAE
3	Kappa(Sm=1.4)	0,116868	104,014	0	0,520677	52,00699	10	2,383739	48,22475	2,056381	2384,086
4	ZN (Ku,Pu)	0,182757	172,8515	0	1	86,42575	10	1,816181	38,53659	2,339938	2430,437
5	ZN (Ku,Pu) apr.	0,101213	184,6367	0	1	92,31836	10	3,332753	81,09293	1,545185	3489,242
6	ITAE-Dist	0,08361	105,0236	0	1	52,51179	10	3,347231	63,9781	1,639012	2574,834
7	ITAE-Setpoint	0,057276	79,18311	0	1	39,59156	10	4,142099	62,84097	1,515032	2885,182
8	Kappa(Sm=1.4)	0,116868	104,014	27,72161	0,520677	52,00699	10	4,682572	66,95368	1,420693	2289,985
9	ZN (Ku,Pu)	0,243676	103,7109	25,92773	1	51,85545	10	2,2583	40,79564	2,079786	1544,22
10	ZN (Ku,Pu) apr.	0,134951	110,782	27,69551	1	55,39101	10	4,063043	66,287	1,497423	1943,846
11	ITAE-Dist	0,132354	84,403	27,55693	1	42,2015	10	4,098724	54,37842	1,519247	2012,998

Figura 17: Planilha do *MS Excel* com os dados calculados pelo *PID Controller Design Toolbox*.

### 3.2.6 *Helps on-line*

Para auxiliar o usuário na utilização do *PID Controller Design Toolbox* estão disponíveis o *Help On-line* como comando de linha no próprio *MATLAB* e o *Help On-line* na versão em *html*. O segundo pode ser executado através de comando de linha no *MATLAB* ou na janela principal do *PID Controller Design Toolbox* clicando com o botão direito sobre um dos métodos de ajuste, por exemplo. A versão em *html* do *Help On-line* é mais completa, trazendo informações não apenas sobre a utilização do *PID Controller Design Toolbox* como também a teoria envolvida por trás desta implementação.

A figura 18 mostra o resultado de se clicar com o botão direito sobre o método *IMC* de ajuste de controladores.

## 4 CONCLUSÕES E FUTURAS VERSÕES

Esse trabalho mostrou as principais potencialidades do PID-TB. Na versão em que o Toolbox se encontra já tem sido intensamente utilizado com sucesso no curso de graduação em Controle de Processos do Departamento de Engenharia Química (UFRGS). Os estudantes tem aprovado a sua utilização nessa disciplina.

As próximas versões dessa ferramenta incluirão técnicas de ajuste baseadas em critérios de otimização, assim como o método da aproximação no domínio da frequência. Além disso, serão implementadas algumas animações em SIMULINK que facilitará a visualização da teoria de controle. Também será incluída a conversão dos parâmetros do controlador PID para os formatos mais encontrados nos SDCDs comerciais.

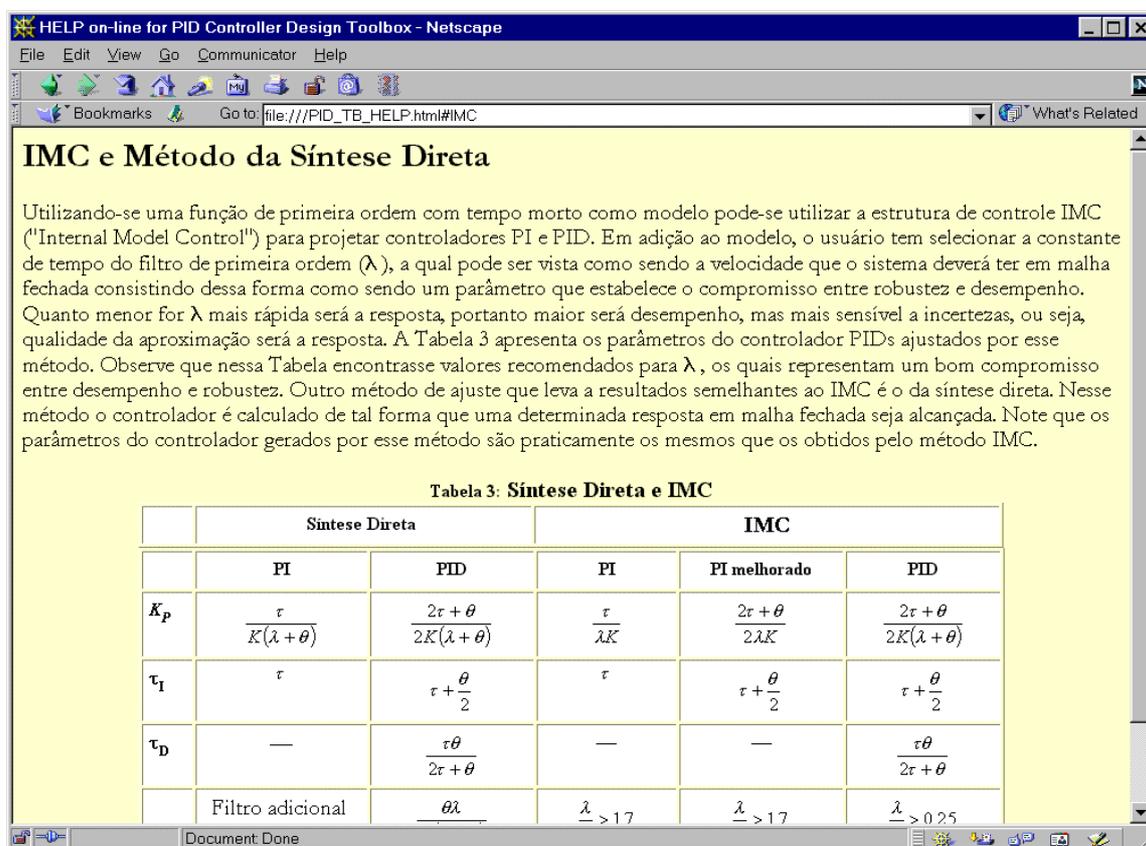


Figura 18: Help On-line em html sobre o método IMC de ajuste de controladores.

## REFERÊNCIAS

- [AH95] K.J. Åström, T. Hägglund "PID Controllers: Theory, Design, and Tuning", second edition, *Instrument Society of America*, (1995)
- [CONTSID 99] CONTSID toolbox (for CONTinuous-Time System IDentification) for Matlab

- [OR94]** B.A. Ogunnaike, W. H. Ray, "Process Dynamics, Modeling, and Control", Oxford University Press, New York, Oxford, (1994).
- [SEM89]** D.E. Seborg, T.F. Edgar, D.A. Mellichamp, "Process Dynamics and Control", *John Wiley & Sons*, (1989).
- [SID99]** Ljung, System Identification Toolbox, Mathworks, (1999)
- [SK77]** Sundaresan, K. R. and P. R. Hrishnaswamy, "Estimation of time delay time constant parameters in time, frequency, and Laplace domains", *Can. J. Chem. Eng.*, 56, pp. 257, (1977)

## APÊNDICE: IDENTIFICAÇÃO DE SISTEMAS DISCRETOS

Através de uma simples interface especialmente desenvolvida para o PID-TB é possível se utilizar todas as funções de identificação disponíveis no "System Identification Toolbox", as quais assumem que o sistema a ser identificado é linear e invariante no tempo e pode ser descrito através da seguinte função de transferência

$$y(t) = G(q)u(t) + H(q)e(t) \quad (7)$$

onde  $y(t)$  é a saída do sistema,  $u(t)$  as variáveis manipuladas e/ou distúrbios medidos,  $e(t)$  representa uma seqüência aleatória com uma função de densidade de probabilidade específica,  $H(q)$  a função de transferência dos distúrbios do processo e  $G(q)$  a função de transferência da dinâmica do processo. Dessa forma, o objetivo passa a ser a identificação dos parâmetros da equação (7) bem como a sua forma (McFarlane e Rivera, 1992).

As técnicas paramétricas que visam minimizar o erro predito são as mais utilizadas, e apresentam a seguinte forma geral:

$$A(q)y(t) = \frac{B(q)}{F(q)}u(t - n_k) + \frac{C(q)}{D(q)}e(t) \quad (8)$$

onde  $A$ ,  $B$ ,  $C$ ,  $D$  e  $F$  são polinômios em  $q^{-1}$  (para  $q^{-k} x(t) = x(t-k)$ ),  $n_k$  é o tempo morto do sistema e:

$$A(q) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a}, \quad B(q) = b_0 + b_1q^{-1} + \dots + b_{n_b}q^{-n_b}, \quad C(q) = 1 + c_1q^{-1} + \dots + c_{n_c}q^{-n_c}, \\ D(q) = 1 + d_1q^{-1} + \dots + d_{n_d}q^{-n_d} \quad e \quad F(q) = 1 + f_1q^{-1} + \dots + f_{n_f}q^{-n_f}$$

O polinômio  $A$  corresponde ao termo auto-regressivo; o termo  $B/F$  corresponde a função de transferências para as entradas do sistema, enquanto que  $C/D$  corresponde a função de transferência para os distúrbios. Existem várias outras formas mais simples que derivam da apresentada na equação (8) que geralmente são utilizadas conforme o caso, uma delas a FIR (*Finite Impulse Response*) que apresenta a seguinte forma:

$$y(t) = B(q)u(t - n_k) + e(t). \quad (9)$$

Neste modelo as informações das saídas em tempos passados não são utilizadas. Um outro modelo, muito usado, que utiliza estas informações é o ARX (*Auto Regressive with eXtra input*) que possui a seguinte forma:

$$A(q)y(t) = B(q)u(t - n_k) + e(t). \quad (10)$$

Esta forma é muito utilizada devido a sua simplicidade e possibilidade de um estudo rápido do sistema que se pretende identificar. A figura A1 mostra a sua representação em diagrama de blocos

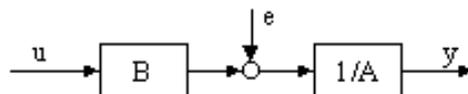


Figura A1: Esquema em diagrama de blocos da estrutura ARX.

Como pode ser observado, esta representação possui um erro com a mesma dinâmica do sistema, ou seja, ele é incorporado à mesma. No caso do sistema apresentar apenas ruídos nos instrumentos de medida, isto é, não acrescenta erros à dinâmica do sistema, mas sim diretamente nas saídas do mesmo, uma estrutura do tipo OE (*output error*),

$$y(t) = \frac{B(q)}{F(q)} u(t - n_k) + e(t) \quad (11)$$

deve ser utilizada. Essa forma possui a seguinte representação em diagrama de blocos:

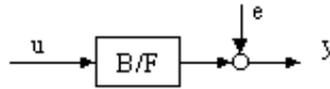


Figura A2: Esquema em diagrama de blocos da estrutura OE

A estrutura ARMAX (*Auto Regressive with Moving Average and eXogenous Variables*),

$$A(q)y(t) = B(q)u(t - n_k) + C(q)e(t) \quad (12)$$

é uma forma mais sofisticada que as anteriores para representar o erro. A sua representação na forma de diagrama de blocos é:

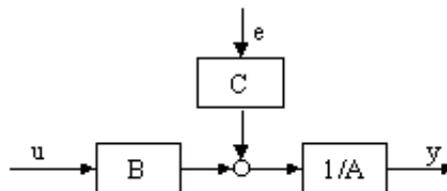


Figura A3: Esquema em diagrama de blocos da estrutura ARMAX.

Uma outra de forma de modelo muito usada é o de *Box-Jenkins*, e é o que melhor lida com os erros devido a sua estrutura, a qual é geralmente representada por:

$$y(t) = \frac{B(q)}{F(q)} u(t - n_k) + \frac{C(q)}{D(q)} e(t). \quad (13)$$

A figura A4 mostra o modelo de *Box-Jenkins* na forma de diagrama de blocos.

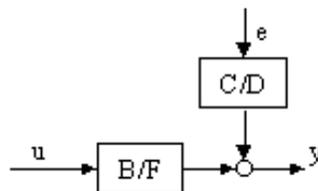


Figura A4: Esquema em diagrama de blocos da estrutura Box-Jenkins

Os parâmetros de um modelo podem ser determinados via métodos de otimização, sendo um dos mais populares o Método dos Mínimos Quadrados (MMQ). Os parâmetros envolvidos nestes modelos são estimados usando métodos de otimização iterativos, com exceção da estrutura ARX,

que possui uma forma direta para o cálculo dos seus parâmetros, sendo este um dos motivos de sua intensa utilização, principalmente para uma análise prévia da ordem do sistema e do seu tempo morto.