

Sintonia PID do tanque de aquecimento da Planta Didática Smar com pidtuner

Luiz Vitor da Silva Vieira Reis¹

Rodrigo de Lima Araújo²

Victor Matheus Rodrigues de Carvalho³

Universidade de Brasília - Departamento de Engenharia Mecânica - Grupo de Automação e Controle - Automação de processos

1 - luizvitorreis@gmail.com

2 - rodrigolima65@gmail.com

3 - victor.matheusrdc@gmail.com

Resumo. Neste trabalho é apresentado o desenvolvimento e aplicação de um controlador PID, utilizando a ferramenta pidtuner do software MATLAB, para controlar o processo do tanque de aquecimento da planta didática Smar.

Palavras-chave: controlador PID, Planta Didática smar, pidtuner.

1. INTRODUÇÃO

Alguns sistemas de controle não apresentam a resposta requerida pelo projetista, nesse caso é necessário a aplicação de um controlador, que quando aplicado ao modelo da planta fará com que o sistema apresente a resposta que queremos. A figura 1 mostra os parâmetros que caracterizam a resposta de um sistema de controle. Tais parâmetros são: t_s - Tempo de assentamento, t_r - Tempo de subida, t_p - Tempo de pico, M_p - Ultrapassagem percentual, entre outros.

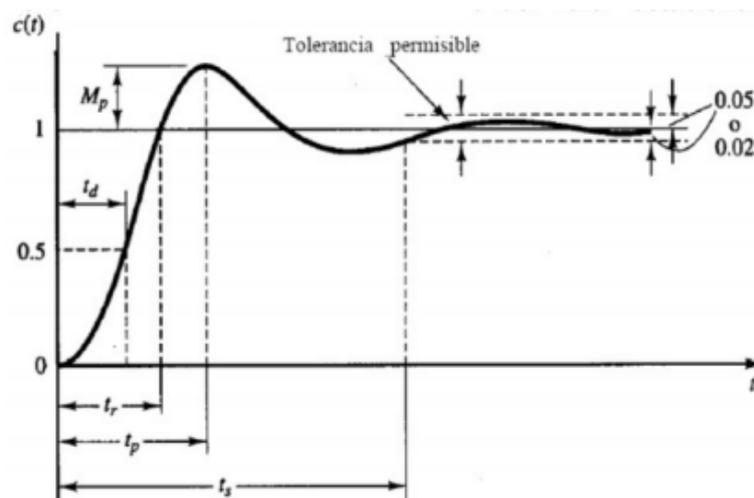


Figure 1. Parâmetros da resposta de um sistema de controle.

Tais parâmetros podem ser controlados a partir da aplicação de um controlador PID (proporcional - integrativo - derivativo), esse controlador é um dos mais utilizados na indústria. A forma geral de um controlador PID é dada por:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

Em que $u(t)$ é o sinal de saída, $e(t)$ é o sinal de erro do sistema em malha fechada, e K_p , K_i e K_d são os ganhos das ações proporcional, integral e derivativa respectivamente. A ação proporcional produz um sinal de saída que é proporcional à amplitude do erro $e(t)$, e permite eliminar as oscilações do sinal de saída. A ação integral produz um sinal de saída que é proporcional à magnitude e à duração do erro, ou seja, ao erro acumulado. Isso fornece uma alternativa para corrigir o erro de off-set gerado pela ação integral e acelera a resposta do sistema, permitindo-o chegar ao valor de referência mais rapidamente. Já a ação derivativa fornece uma correção antecipada do erro, diminuindo o tempo de resposta e melhorando a estabilidade do sistema. O controlador descrito pela equação 1 pode ser visto em diagrama de blocos na figura 2

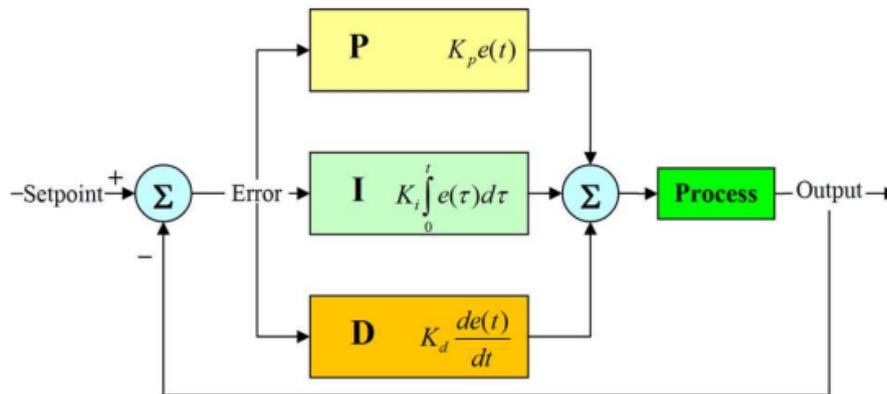


Figure 2. Sistema de controle em malha fechada com controlador PID.

Portanto, para aplicação do controlador devemos conhecer o modelo da planta, e determinar os ganhos K_p , K_d e K_i em função da resposta que queremos. Com a utilização da ferramenta *pidtuner* podemos obter os valores ótimos para os ganhos do controlador.

1.1 PLANTA DIDÁTICA SMAR

A planta didática SMAR permite aplicarmos várias malhas de controle, servindo assim como uma ferramenta que simula as aplicações industriais reais. A planta é monitorada via PC e utilizando um software de supervisão, que efetua a aquisição de dados dos equipamentos e o apresenta por meio de animações de telas, a planta disponível no laboratório (GRACO - UnB) é a PD 3 que possui tecnologia Foundation Fieldbus com controles de temperatura, vazão e nível. Permite também atuar nos registros modificando valores internos dos equipamentos e nos modos operacionais das malhas de controle. A planta didática pode ser acessada a partir dos aplicativos instalados no PC, sendo eles o **SYSTEM 302** e o **LogicView**. A tela de sinótico da planta didática Smar, permite um fácil acesso a mesma, permitindo ações como: abertura de válvulas, acionamento das bombas, atuar sobre o aquecedor, etc. O sinótico pode ser visto na figura 3.

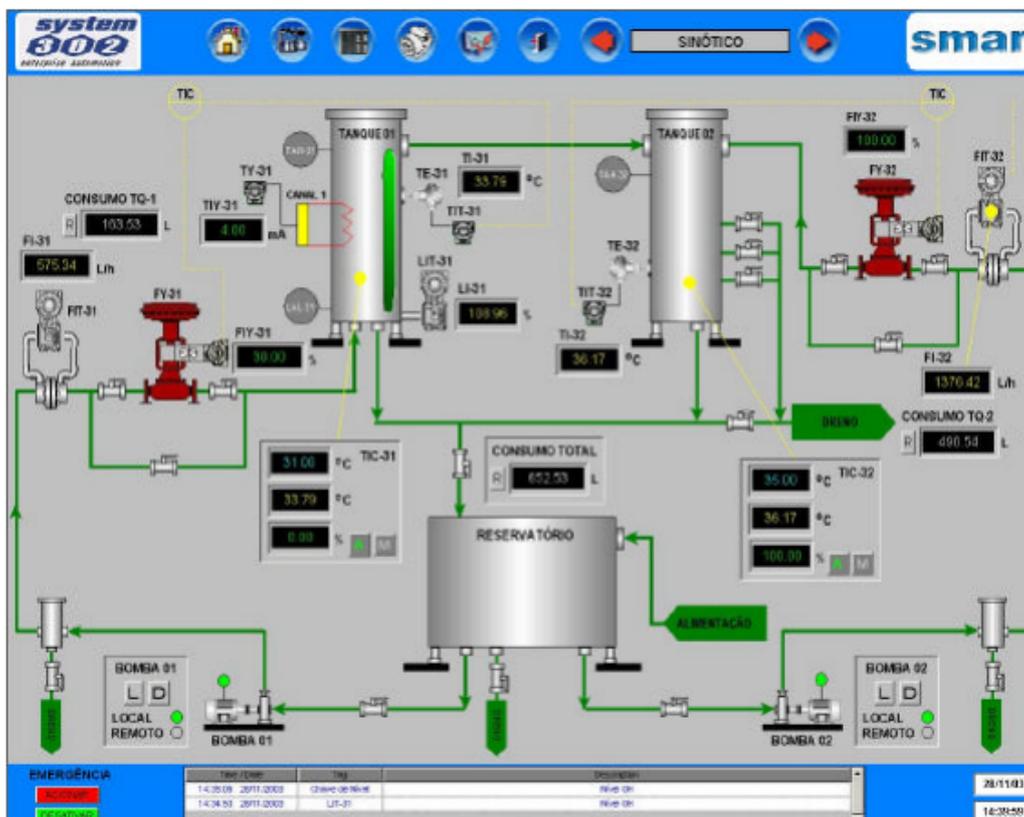


Figure 3. Sinótico da planta didática SMAR.

A planta didática possui 3 tanques, denominados tanque de armazenamento de água (Reservatório), tanque de água quente (Tanque 01) e tanque de mistura (Tanque 02) mostrados na figura 3. O tanque de água quente possui uma chave de nível e um termostato para respectivas regulagens de nível e temperatura da água do tanque. A chave de nível acusa nível baixo quando a capacidade do tanque for inferior. O termostato acusa temperatura alta no tanque. Quando a temperatura for alta a animação da sinalização de alarme do termostato ficará vermelha a 90 % de sua capacidade. Neste trabalho foi aplicado o método de controle PID no tanque de aquecimento da planta, os principais componentes envolvidos no processo são: **Resistência de Imersão**, estas resistências estão localizadas dentro do tanque de água quente e servem para aquecer a água deste tanque, temos também o **Conversor Estático**, que é o responsável por alimentar eletricamente as resistências de imersão, temos ainda o **Termostato**, e a **Chave de Nível**, já descritos anteriormente. Todas as especificações técnicas desses dispositivos estão descritas no manual da planta, disponível em [2].

1.2 PID Tuner

O *PID tuner* consiste em uma ferramenta do MATLAB que pode ser acessada via linha de comando ou por meio de uma aplicação de interface com usuário. Por padrão, o algoritmo escolhe ganhos proporcional, integral e derivativo objetivando uma resposta balanceada entre aspectos como tempo de resposta, largura de banda e margem de estabilidade. Aliás, a sintonia automática leva o sistema a uma margem de fase de 60 graus. No que tange ao formato da instrução, segue que

$$pidTuner(P, 'p').$$

O primeiro parâmetro corresponde ao modelo que se deseja controlar, ao passo que o segundo diz respeito ao controlador que deve ser projetado. Esses últimos podem possuir de 1 a 2 graus de liberdade. Como exemplo, para controladores com 1 grau de liberdade, os seguintes elementos podem ser criados:

- 'P': Ação proporcional apenas;
- 'I': Ação Integral apenas;
- 'PI': Ação proporcional-integral;
- 'PD': Ação proporcional-derivativo;
- 'PDF': Ação proporcional-derivativo com um filtro de primeira-ordem no termo derivativo;
- 'PID': Ação proporcional-integral-derivativo;
- 'PIDF': Ação proporcional-integral-derivativo com filtro de primeira ordem no termo derivativo.

A possibilidade de inserção de filtros na ação derivativa pode ser importante haja vista a presença de ruídos em sinais de medição nos processos. Isso evita, portanto, a amplificação de tais ruídos devido a essa ação. Nota-se, também, a necessidade da ferramenta de ter o modelo do processo.

Outra forma para projetar o controlador via PID tuner consiste em utilizar uma interface com o usuário na qual é possível introduzir dados de entrada-saída coletados do processo, ou especificar diretamente qual a função de transferência da planta. A vantagem de utilizar essa aplicação está no fato de que, caso se observe uma resposta ainda inadequada do projeto, pode-se alterar os parâmetros do controlador, a fim de torná-lo mais robusto ou equilibrado. A figura 4 apresenta a tela inicial da aplicação.

1.3 System Identification

A ferramenta *System Identification* do MATLAB provê todo o algoritmo para identificação de sistemas. Os dados requeridos consistem em dados do processo em malha aberta, ou seja, com atuações diretas na planta como mostrado na figura 5. Para o caso de um processo de aquecimento, como o que será analisado nesse trabalho, atuar-se-ia no aquecedor (entrada - $u(t)$) para acompanhar a temperatura (saída - $y(t)$).

Em posse desses dados, pode-se facilmente, usando essa aplicação, retirar o modelo do processo. Utiliza-se um exemplo fornecido pelo MATLAB para demonstrar seu funcionamento. Assim, considera-se a relação de entradas e saídas ilustradas na figura 6. Deseja-se obter um modelo de processo de primeira ordem com atraso.

A figura 7 introduz a página inicial da aplicação. Nela, é possível importar dados de entrada e saída que serão analisados. Além disso, pode-se ainda pré-processá-los para eliminação de ruídos, por exemplo. A partir de tal relação, a ferramenta consegue estimar funções de transferências, modelo de processos lineares, não lineares e no espaço de estados, entre outros. Como deseja-se obter o modelo do processo, escolhe-se essa opção juntamente com a forma que se deseja aproximar.

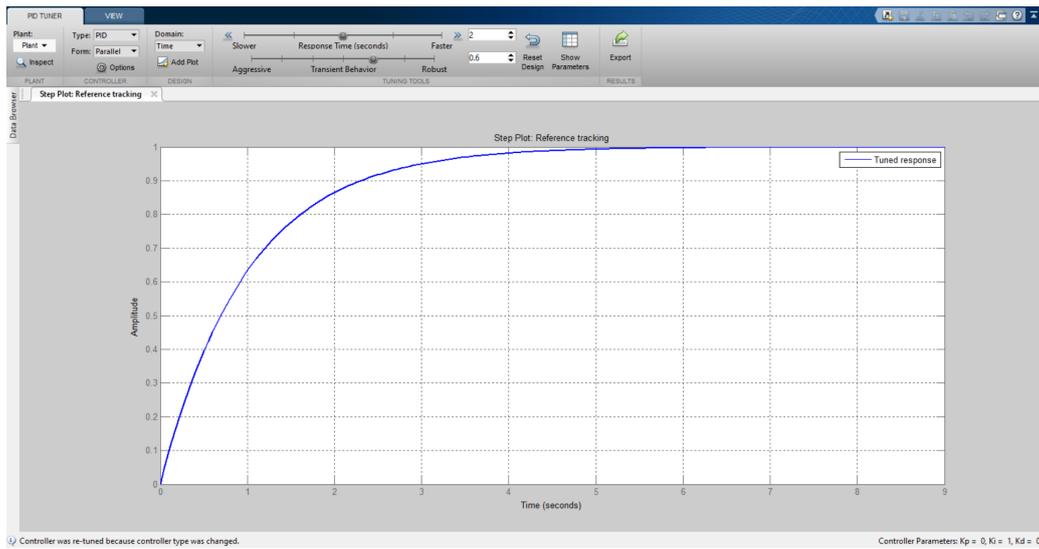


Figure 4. Tela inicial da aplicação PID Tuner.

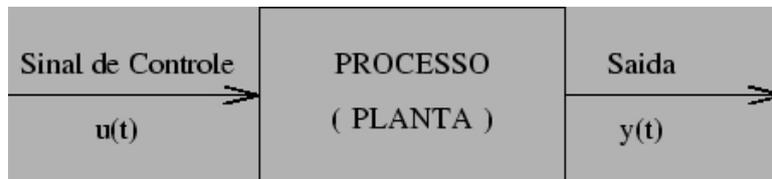


Figure 5. Sistema em malha aberta.

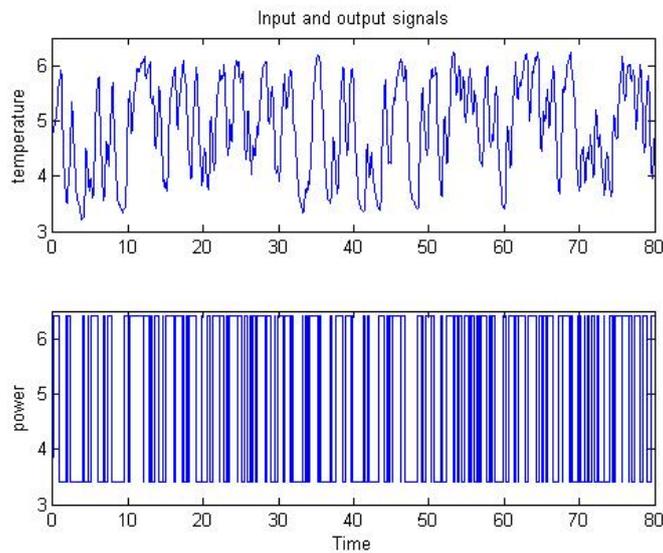


Figure 6. Exemplo de entrada e saída de um processo.

Estimado os valores, a própria ferramenta valida o modelo indicando um *fit* que informa o quão próximo o modelo está do processo real. Note pela figura 8 que o modelo gerado correspondeu a um fit de 82,11%. Dependendo do processo e de qual exato deve ser o modelo, esse resultado pode ser tolerado.

2. METODOLOGIA

2.1 Tags da Planta

A planta didática SMAR utiliza-se de um sistema Foundation Fieldbus para que haja comunicação da mesma com os dispositivos externos. Esse sistema consiste em um fluxo de informações em série, digital cuja finalidade é conectar equipamentos Fieldbus. Tendo-se então uma conexão entre os dispositivos de campo por meio de uma rede local.

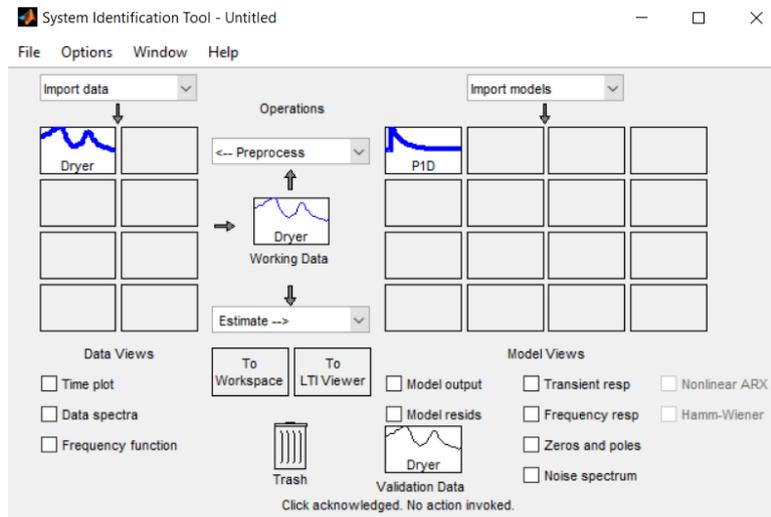


Figure 7. Tela de navegação da ferramenta ident.

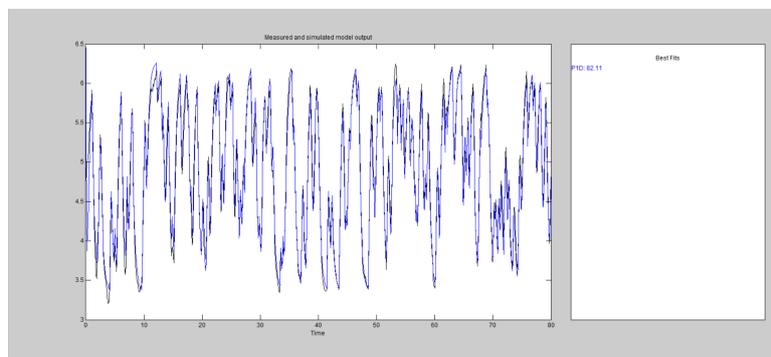


Figure 8. Sistema em malha aberta.

A comunicação com dispositivos específicos é feita por meio da identificação destes, que é feita por tags. Pode-se ler um dado em um dispositivo ou escrever nele, uma vez que sua tag for conhecida. Para que fosse feito o controle PID da planta, as tags que necessariamente tinham de ser conhecidas eram:

$$TY - 31_PID1.OUT.VALUE$$

,

$$TIT - 31_AI1.OUT.VALUE$$

,

$$TY - 31_AO1.PV.VALUE$$

Em que a primeira tag é apenas de escrita na temperatura de imersão. A segunda e a terceira são apenas de leitura e acerca dos dados temperatura no tanque 1 e o outro era a temperatura do aquecedor do tanque.

2.2 Identificação

Para que fosse feito o controle da planta didática SMAR, primeiramente foi montado um arquivo no Simulink que continha os blocos necessários para fazer a comunicação da planta com o MatLab via OPC Tool. A figura 9 explicita tal diagrama. Note que pelo bloco *OPC Read* é possível acessar as tags disponibilizadas pelo servidor OPC do system 302, bem como escrever em certas tags utilizando o bloco *OPC Write*.

Uma vez que os blocos foram devidamente organizados, aplicou-se, em seguida, excitações no processo por meio de um degrau. Na diagrama da figura 9 isso era feito pela simples modificação da *Constant2*. Essa constante, por sua vez, escrevia na tag relacionada com o aquecedor em unidades de engenharia (0% a 100%). Assim, os dados de saída, em °C, relacionados com a temperatura do tanque 1, foram também sendo coletados para ver a resposta do sistema à entrada já citada. Deve-se mencionar que o ponto de operação do processo para a identificação foi de

$$FY - 31_AO1.PV.VALUE(Valvula) = 5\%$$

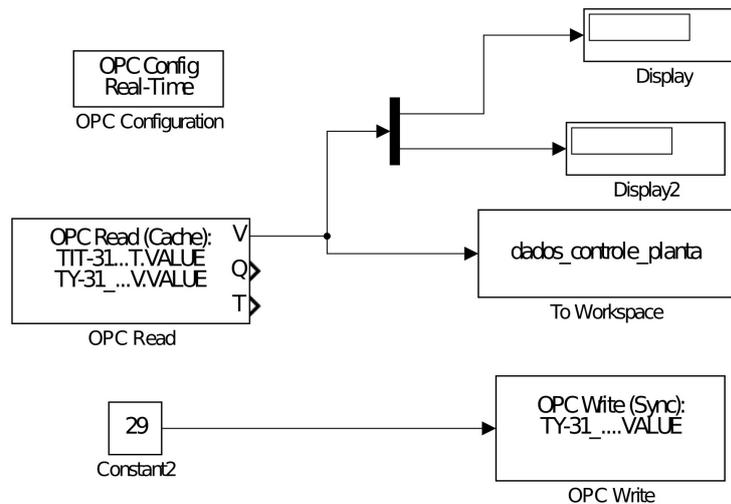


Figure 9. Diagrama para identificação de sistema em malha aberta.

$$TY - 31_PID1.OUT.VALUE(aquecedor) = 0\%$$

O reservatório manteve-se exposto à temperatura ambiente durante todo processo e identificou-se o processo em torno de tal ponto de operação.

Obtendo-se a resposta do sistema no Simulink, exportou-se estes dados para o MatLab onde foi feita a modelagem do sistema através da ferramenta *System Identification*. Importou-se, portanto, a relação entrada/saída e solicitou-se a identificação de um modelo de processo e, levando-se também em consideração os conhecimentos sobre o processo, o melhor *fit* de modelo foi escolhido. Analisou-se em seguida se o *fit* obtido está dentro dos limites aceitáveis de erro para o controle do sistema, obtendo-se assim a modelagem e os parâmetros do sistema em análise.

2.3 Controle de Temperatura

Antes de adentrar no controle PID com o pidTuner, testou-se a sintonia implementada no próprio programa da Smar. Visto o comportamento em relação ao degrau na referência, começou-se o projeto do controlador objetivando comparar os PIDs.

Tendo em posse o modelo do sistema, usou-se o pidTuner de duas formas distintas. A primeira foi simplesmente usando o comando

$$pidtune('Gse','PID')$$

já que desejava-se projetar um PID. A outra forma que se fez foi por meio da aplicação pidTuner como o mostrado na figura 4. Nele, ajustamos o controlador de acordo com a resposta desejada pelo grupo.

Por fim, por meio de um diagrama no Simulink - figura 10 - executou-se ambos projetos do PID.

Algumas, porém, são ainda importantes. A cada teste realizado na planta, trocou-se a água presente no reservatório haja vista a premissa de projeto que a temperatura de entrada nos tanques são constantes. Caso isso não ocorresse, toda a água circulante tenderia a uma mesma temperatura, alterando a dinâmica de todo o processo.

3. RESULTADOS

Para fins de comparação com o resultado da aplicação do controlador PID projetado pelo pidTuner, traçou-se a curva da figura 11 que consistiu na execução do controle automático direto do sinótico da Smar.

Notou-se um processo oscilante e que não alcançou o setpoint especificado em 35°C. Logo, traçou-se como objetivos o projeto de um PID capaz de eliminar as altas oscilações e que alcançasse, de fato, a referência.

Como especificado na seção anterior, o primeiro passo foi a aplicação de um degrau, cujo valor foi de 10%. A resposta, então, foi coletada e foi por meio da qual se obteve a relação entrada-saída da resposta em malha aberta mostrada na figura 12.

Usando a ferramenta *System Identification*, testou-se dois modelos de processo característicos: um de primeira ordem

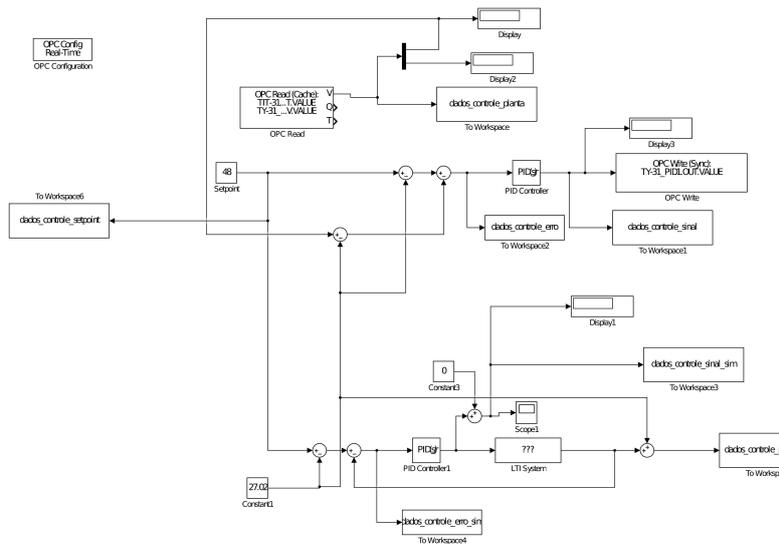


Figure 10. Diagrama para controle em malha fechada.

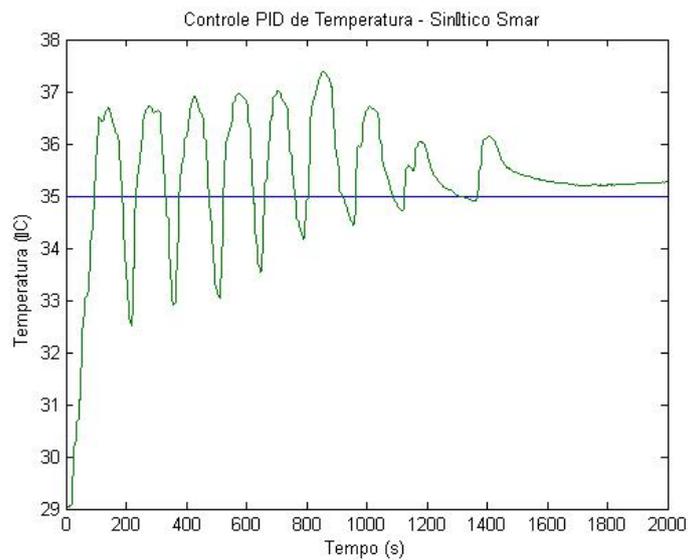


Figure 11. Controle automático PID do sinótico.

com atraso e outro acrescentando um zero na função de transferência. O resultado é mostrado na figura 13 e explicita que o melhor fit foi de 95%.

Isso correspondeu à função de transferência

$$G(s) = \frac{K(1 + \tau_d s)}{1 + \tau_p s} e^{-\theta s} \quad (1)$$

em que

$$K = 1,8684,$$

$$\tau_p = 1287,7,$$

$$\theta = 52,075,$$

$$\tau_d = 54,807.$$

Tal modelo foi, então, o parâmetro de entrada para o pidTune. Nessa forma, os parâmetros do PID entregues pelo comando foram:

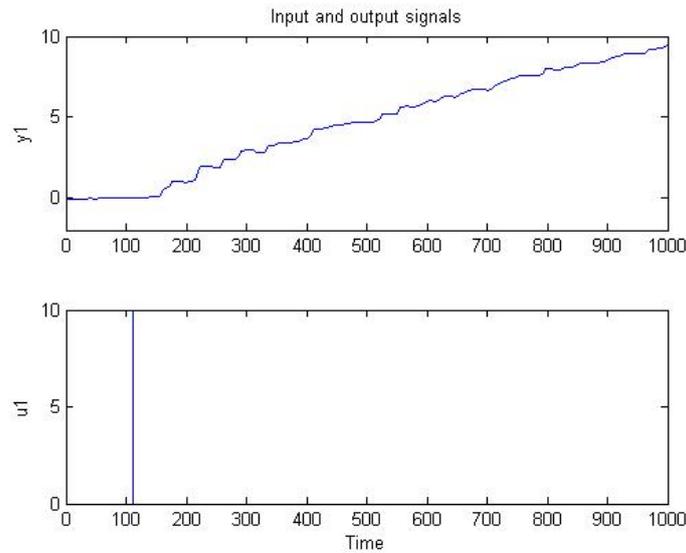


Figure 12. Relação entrada-saída do sistema em malha aberta.

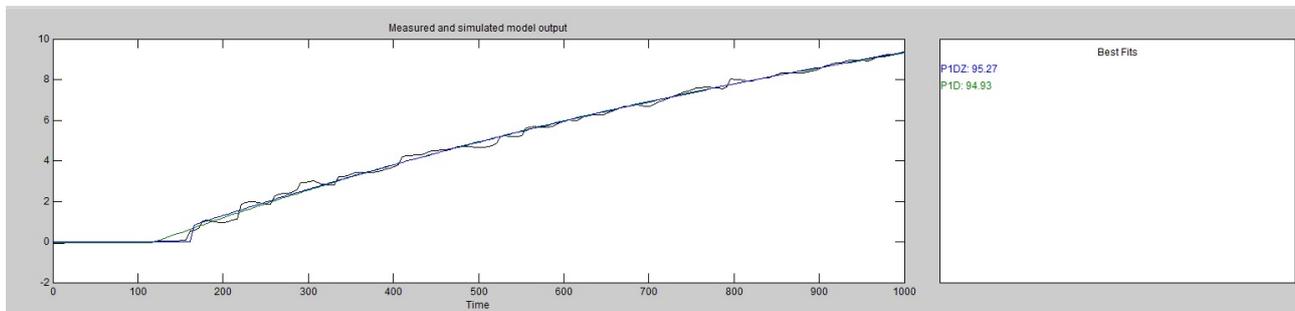


Figure 13. Relação entrada-saída do sistema em malha aberta.

$$K_p = 0,929$$

$$k_i = 0,00622$$

$$K_d = 0.$$

Aplicando-o no processo, obteve-se o resultado mostrado na figura 14.

É notório que o modelo simulado e real coincidiram quanto ao tempo de subida. Todavia, vê-se, nesse resultado, a alta inércia do processo que não foi bem identificado pelo modelo. Outro ponto importante é que esse PID demonstrou menor robustez quando comparado ao PID original da planta. Aqui também explicita uma característica do processo, visto que, pelo fato da temperatura ter tido um sobressinal e isso ser uma grande não-linearidade do processo, já que a água no reservatório cresce de temperatura, a dinâmica é alterada. Assim, percebeu-se a necessidade de usar valores para o ganho derivativo a fim de não existir esse sobressinal indesejado.

Com isso, utilizou-se o aplicativo pidTuner e ajustou-se a resposta para entrar em regime o mais rapidamente possível e não existir, ou ter pouco sobressinal. Feito isso, os parâmetros do PID foram alterados para:

$$K_p = 2,7818$$

$$k_i = 0,0021613$$

$$K_d = -65,7236$$

$$Filtro = 0,02268.$$

Esse filtro visa impedir que a ação derivativa amplifique ruídos de medição.

Aplicando-se, portanto, tal PID projetado, obteve-se o resultado mostrado na figura 15.

Eliminando o problema do sobressinal, citado anteriormente, viu-se maior proximidade entre modelo e processo real. Além disso, o controlador demonstrou bom desempenho sendo robusto diante de perturbações no processo. As oscilações da temperatura ficavam entre 1°C para cima e para baixo, o que é bem menor do que o comportamento do PID original da planta.

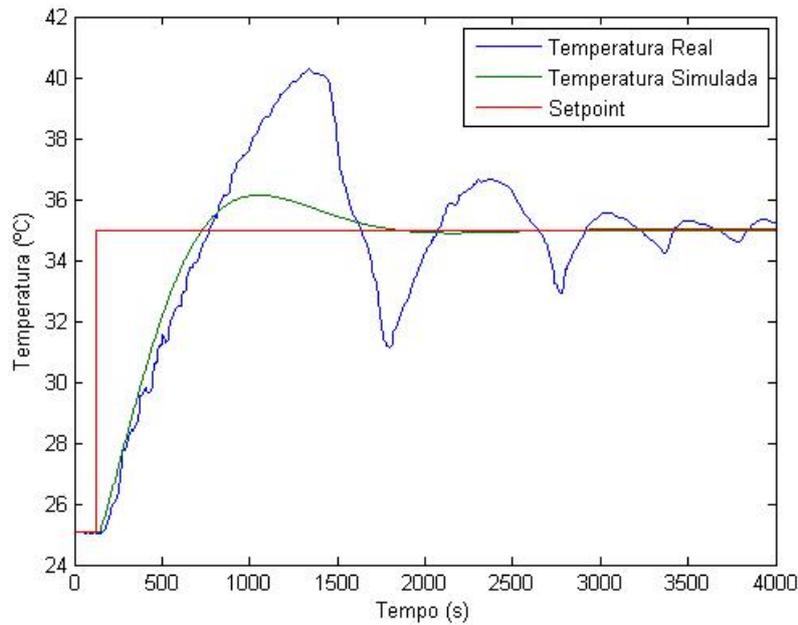


Figure 14. Controle com PID sintonizado com comando pidtune.

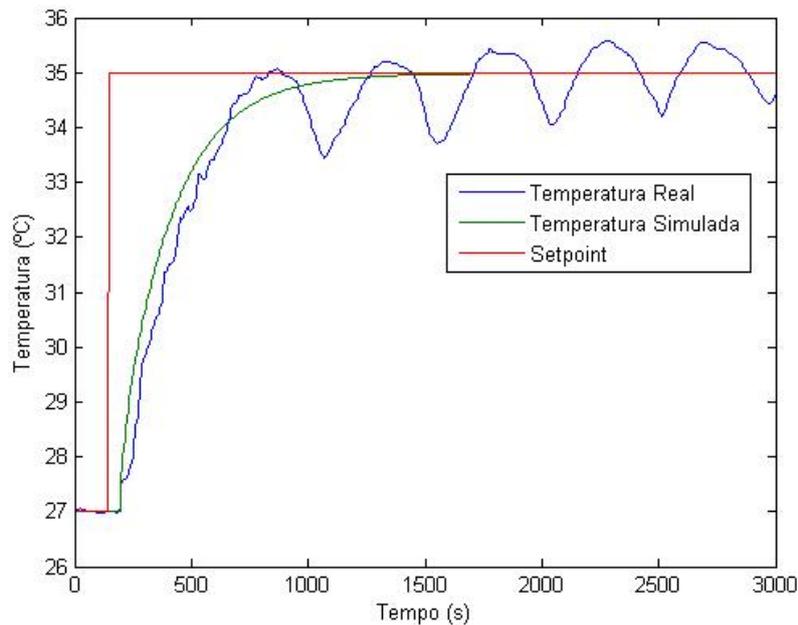


Figure 15. Controle PID ajustado no pidTuner.

4. CONCLUSÃO

Este trabalho mostrou a aplicação de controle de temperatura em tanque de aquecimento presente na planta didática da Smar. Para isso, utilizou-se a ferramenta do pidtune/pidTuner para realizar a sintonia do controlador PID.

Antes do controle, porém, obteve-se o modelo do processo por meio da *System Identification Toolbox* presente no MatLab. Contando com ajuda do Simulink a fim de realizar a comunicação OPC e resgatar as tags necessárias, fez-se diagramas capazes de realizar todas as etapas do trabalho. Por fim, testou-se três controladores na planta, a saber: o original, usando comando pidtune e por meio da aplicação pidTuner.

A partir dos resultados obtidos e interpretação destes, pode-se perceber que a integração do MatLab com a planta didática de fato permitiu a modelagem do sistema em análise e o controle do mesmo. O controle feito usando pidtune mostrou boa proximidade quanto ao tempo de subida em comparação com o modelo simulado. Todavia, não-linearidades

presentes na planta a partir do momento que a temperatura se elevava acima do setpoint atrapalharam o desempenho do controlador. Nesse caso, o controlador projetado pela aplicação pidTuner mostrou-se o mais eficiente deixando o sistema oscilar em torno do setpoint em 1°C. Notou-se, portanto, a eficiência desta ferramenta para controle de processos industriais.

Por fim, no que tange aos trabalhos futuros, pode-se utilizar a válvula presente na entrada do tanque 1 para também controlar a temperatura e, dessa forma, realizar um controle conjunto com o aquecimento. Outra aplicação seria a adoção de técnicas mais avançadas como preditiva, para antecipar não-linearidade do processo e corrigi-lo a fim de diminuir sobressinais e tornar o controle mais eficiente.

Diante de todo o exposto, conclui-se que o grupo realizou o trabalho com êxito aplicando bem a ferramenta de sintonia automática do MatLab.

5. REFERÊNCIAS

[1] Nise, S. N. **Engenharia de Sistemas de Controle**, LTC, sexta edição, 2011.

[2] Manual da planta didática SMAR, disponível em: www.smar.com/PDFs/manuals/PILOPLPMP.pdf.