

PROJETO DE CONTROLADORES PI/PID DISCRETIZADOS PARA VELOCIDADE ANGULAR DE UM ROBÔ MÓVEL COM TRACÇÃO DIFERENCIAL

ALLAN ALEX FRANÇA, SAMMARA R. VILAR, LINCOLN M. ARAÚJO, ADEMAR G. COSTA JUNIOR

Laboratório de Instrumentação, Sistemas de Controle e Automação (LINSCA)

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba (IFPB)

E-mails: allan.alex@outlook.com, sammaravilar@gmail.com, {lincoln.araujo, ademar.costa}@ifpb.edu.br

Abstract— This paper presents the results of the PI / PID digital controllers design for the angular speed of a nonholonomic mobile robot based on the NI LabVIEW Starter Kit (DaNI 2.0) robotics platform with the use of LabVIEW for FPGA programming contained in the robotic platform. For this, initially it was performed the mathematical modelling of the dynamical system through methods based on the reaction curves of the system with the application of a step signal at the input. With the established sampling time and discretization of the transfer functions, the systems were analyzed and chosen for the simulation of PI / PID controllers, tuned by the methods of Ziegler-Nichols, CHR and Johnson & Moradi. For the experimental phase, the two best controller designs of the simulation phase were chosen, with the results of the two phases being compared, obtaining values very close, validating the project.

Keywords— PI controller, PID controller, mobile robotics, LabVIEW, linear model identification.

Resumo— Este trabalho apresenta os resultados do projeto de controladores digitais PI/PID para a velocidade angular de um robô móvel não holonômico baseada na plataforma de robótica NI LabVIEW Starter Kit (DaNI 2.0) da National Instruments, com a utilização do LabVIEW para a programação do FPGA contido no robô móvel. Para isso, inicialmente foi realizada a modelagem matemática do sistema dinâmico por meio de métodos baseados nas curvas de reação do sistema com a aplicação de um sinal do tipo degrau na entrada. Com o tempo de amostragem estabelecido e a discretização das funções de transferências, os sistemas foram analisados e escolhidos para a realização da simulação dos controladores PI/PID, sintonizados pelos métodos de Ziegler-Nichols, CHR e Johnson-Moradi. Para a fase experimental, foram escolhidos os dois melhores projetos de controladores da fase de simulação, com os resultados das duas fases sendo comparados, obtendo valores bastante próximos, validando o projeto.

Palavras-chave— Controlador PI, controlador PID, robótica móvel, LabVIEW, identificação de modelos lineares.

1 Introdução

Os robôs móveis têm sido alvo de vários estudos de forma a incluí-los em ambientes complexos como domésticos, industriais, urbanos e militares (Romero *et al.*, 2014). O controle de velocidade dos sistemas robóticos implementados nos diversos ambientes é uma preocupação, uma vez que esses devem ser ágeis, eficazes e rápidos, mantendo os padrões de segurança, evitando colisões com obstáculos e situações que possam colocar em perigo seres humanos presentes no mesmo ambiente (Ferreira Filho and Pimenta, 2015; Almasri, Alajlan and Elleithy, 2016).

Segundo Kheir *et al.* (1996), um projeto de um sistema de controle compreende, de uma maneira geral, duas etapas: i) modelagem/identificação do sistema a ser controlado; ii) projeto de controladores que satisfaçam as especificações de desempenho, estabilidade relativa exigida, simulação utilizando computadores digitais, implementação do controlador no sistema real.

Na literatura existem diversas técnicas de controle que trazem eficiência, estabilidade, precisão, robustez e confiabilidade ao sistema dinâmico. Um dos controladores mais utilizados é o PID, com suas ações proporcional, integral e derivativa (Astrom and Hagglund, 1995).

Modelos matemáticos que possam ser utilizados para o projeto de controladores baseados em modelos para a robótica móvel, partem do conhecimento sobre a cinemática e a dinâmica (Jazar, 2010; Tzafestas, 2014). Como uma alternativa a obtenção de mo-

delos matemáticos baseados nas equações fenomenológicas que regem o comportamento dinâmico dos sistemas, o uso da identificação de sistemas que se baseia em dados experimentais é uma opção (Ljung, 1999). Técnicas mais simples baseadas na aplicação de sinais do tipo degrau à entrada do sistema dinâmico e a análise da curva de reação (a saída) possuem diversas propostas na literatura, como por exemplo, os métodos de Ziegler-Nichols, Nishikawa, Mollenkamp, Smith, Harriot, entre outros (O'Dwyer, 2012).

Xiao and Wang (2011) projetaram um PID autossintonizável para a movimentação de um robô móvel por meio do uso dos mínimos quadrados para a estimação de parâmetros, implementado em um DSPIC. Sharma and Jain (2016) implementam o controle de velocidade em uma plataforma robótica móvel usando controladores PI, PID e alocação por polos, utilizando o modelo de um motor DC, com seus parâmetros fornecidos pelo fabricante, com o projeto baseado no plano s .

Pandey, Jha and Chakravarty (2017) projetam o controlador digital PID para a velocidade angular de um robô móvel de três rodas, por meio do uso do modelo do motor DC baseado em uma função de transferência em Laplace, com dados dos parâmetros utilizados do trabalho de Oguntoyinbo (2009).

Neste trabalho são apresentados a identificação de modelos matemáticos obtidos por curvas de reação e o projeto de controladores PI/PID digitais com o objetivo de controlar a velocidade angular das rodas da plataforma robótica NI LabVIEW *Starter Kit* (DaNI 2.0), que representa um sistema não holo-

nômico com tração diferencial, utilizando o LabVIEW.

Este trabalho está dividido da seguinte forma: na segunda seção é apresentada uma formulação matemática para robôs móveis com duas rodas e uma de suporte, por meio das equações fenomenológicas do sistema. Na terceira seção é apresentada a plataforma de robótica móvel utilizada e na quarta seção são apresentadas as metodologias utilizadas para a realização do projeto. Na quinta seção são apresentados os resultados obtidos e as discussões sobre eles e por fim, na sexta seção são realizadas as conclusões e perspectivas futuras para a pesquisa.

2 Formulação Matemática para Robôs Móveis

Os modelos matemáticos para representar o comportamento dinâmico de um determinado sistema podem ser obtidos por meio das equações fenomenológicas que a regem ou por meio do uso de técnicas de identificação de sistemas (Ljung, 1999), sendo a segunda uma técnica alternativa em relação a primeira.

Dentre as diversas técnicas de identificação de sistemas, encontram-se as que se baseiam na resposta da curva de reação ao ser aplicado um sinal do tipo degrau à entrada do sistema dinâmico. Esse é um método determinístico, o qual existem diversas formas que podem ser utilizados para investigar se tais

modelos matemáticos obtidos podem ser utilizados para representar o comportamento dinâmico do sistema.

Os modelos matemáticos para representar o comportamento dinâmico de robôs móveis por meio de equações fenomenológicas utilizam conceitos de cinemática, que permite o estudo do movimento, velocidades e acelerações, e a dinâmica, que permite avaliar, além dessas, também as forças envolvidas (Jazar, 2010; Tzafestas, 2014).

Na robótica móvel, surgem diversos modelos para as configurações em que o robô móvel seja construída, baseado, por exemplo, na quantidade e no tipo de rodas que possua. Na configuração utilizada neste trabalho, o robô móvel possui três rodas, com duas acionadas individualmente e uma terceira que gira livremente (roda de suporte), onde essa terceira tem efeitos desprezíveis na dinâmica do robô móvel. Com acionamento diferencial nas duas rodas, diferenças de velocidade entre as mesmas fazem com que o robô móvel mude sua orientação.

Bish *et al* (2007) apresentam um modelo matemático baseado para um robô móvel na configuração de três rodas ilustrada na Figura 1, semelhante ao utilizado neste trabalho, representando a velocidade das duas rodas com acionamento diferencial, dados em m/s, como:

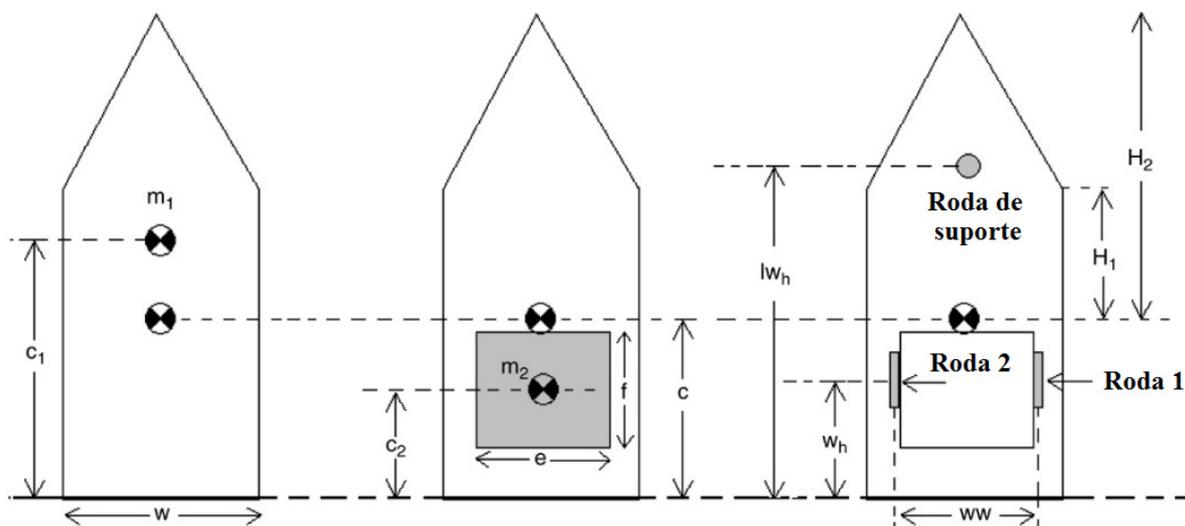


Figura 1. Dimensões do robô móvel para o modelo matemático de Bish *et al* (2007).

$$v^{\omega_{1,2}} = [u_1 \cos(q_3) + u_2 \sin(q_3) + u_3(c - \omega_h)]\hat{\mathbf{b}}_1 + [-u_1 \sin(q_3) + u_2 \cos(q_3) \pm u_3(\omega\omega/2)]\hat{\mathbf{b}}_2 \quad (1)$$

sendo: $\omega\omega$, a distância entre as rodas (em metros); ω_h , a distância da parte traseira do robô ao centro da roda (em metros); c , a distância da parte traseira do robô ao seu centro de massa (em metros); q_i ($i =$

1,2,3), coordenadas que definem o deslocamento angular (em rad); $\hat{\mathbf{b}}_1$ e $\hat{\mathbf{b}}_2$, vetores.

Para a roda livre, a representação matemática é estabelecida por:

$$v^{\omega} = [u_1 \cos(q_3) + u_2 \sin(q_3) + u_3(c - l\omega_h)]\hat{\mathbf{b}}_1 + [-u_1 \sin(q_3) + u_2 \cos(q_3)]\hat{\mathbf{b}}_2 \quad (2)$$

com $l\omega_h$, sendo a distância da roda livre até a parte traseira do robô (em metros).

Em termos do uso de métodos de identificação de sistemas para a obtenção de modelos matemáticos que não dependam da manipulação algébrica das equações fenomenológicas, propostas com redes neurais (Roy, Barai and Dey, 2013), mínimos quadrados recursivos (Guerra *et al*, 2004) e por caixa cinza (Okuyama *et al*, 2017) apresentaram resultados que possibilitam seu uso com o objetivo de projetar controladores baseados em modelos matemáticos.

3 Plataforma de Robótica Móvel Utilizada

O sistema dinâmico robótico utilizado é uma plataforma robótica para ensino, pesquisa, e prototipagem desenvolvida pela *National Instruments*, denominada de DaNI 2.0, ilustrada na Figura 2. Essa plataforma é dotada de sensores, atuadores e uma placa *Single-Board RIO*, que é um dispositivo de aquisição e controle embarcado baseado em FPGA (*Field Programmable Gate Array*) e portas de entrada/saída em uma única placa (National Instruments, 2016).

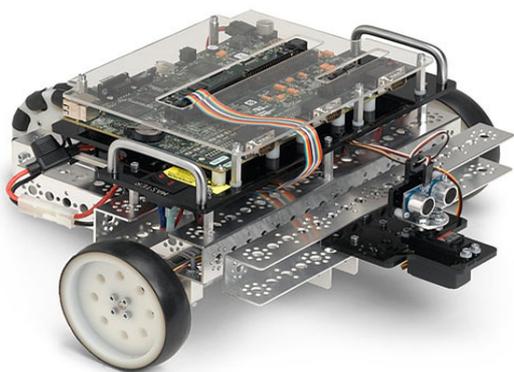


Figura 2. Plataforma robótica DaNI 2.0. Fonte: National Instruments, 2016.

A plataforma robótica DaNI pode ser programada por meio da linguagem gráfica LabVIEW, utilizando o *toolbox Robotics*. O uso do LabVIEW permite a programação do FPGA contido na placa *Single-Board RIO*, sendo possível a execução de tarefas como o controle a altas taxas de velocidade, a temporização de entradas e saídas, bem como o processamento de sinais (National Instruments, 2016). A arquitetura simplificada de *hardware* dessa plataforma é ilustrada na Figura 3.

Por esse trabalho se tratar do controle da velocidade angular para cada uma das rodas do robô de forma independentes, ou seja, cada roda é tratada como um sistema monovariável. Os elementos na plataforma DaNI utilizados são a placa de controle dos motores (*Sabertooth 2x12 R/C*) que aciona os dois motores recebendo sinais de PWM do FPGA, os motores DC *Pitsco* (atuadores) e os *encoders* ópticos *Pitsco* (sensores de realimentação), todos contidos no diagrama de blocos da Figura 3.

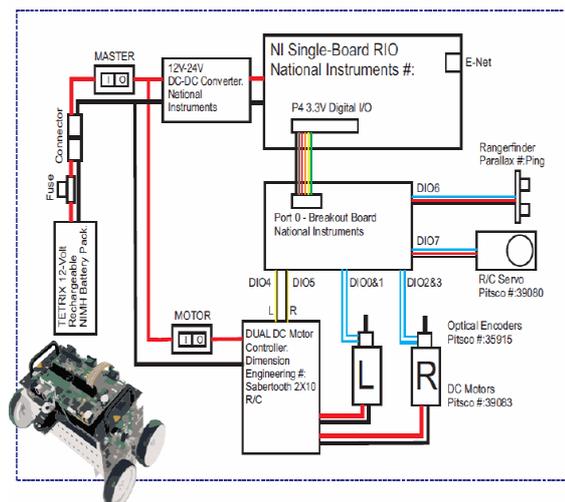


Figura 3. Arquitetura simplificada do *hardware* da plataforma robótica DaNI. Fonte: National Instruments, 2016.

Um pulso PWM de 50 Hz com um *duty cycle* variável de 5% (1000 μ s) a 10% (2000 μ s), determina o sentido de giro e a velocidade de rotação do motor. Com um *duty cycle* de 7,5% o motor não rotaciona, com 5% apresenta velocidade máxima para um sentido de giro e com 10%, apresenta velocidade máxima no sentido de giro contrário, havendo uma relação matemática linear entre as duas variáveis.

4 Metodologia Empregada

O desenvolvimento do projeto foi dividido em cinco etapas. Na primeira etapa foi realizada a modelagem matemática utilizando os métodos em malha aberta baseados na curva de reação do sistema dinâmico propostos por Ziegler-Nichols, Broida, Sundaresan, Smith de 1ª ordem, e Molenkamp (Coelho and Coelho, 2004; O'Dwyer, 2012, pg. 164).

Na segunda etapa do projeto, com o tempo de atraso modelado por Padé de ordem 1 (Astrom and Hagglund, 1995), foi realizada a discretização do sistema, primeiramente escolhendo o tempo de amostragem utilizando o método de Isermann (1980) e depois utilizando os métodos de ZOH (*Zero Order Holder*), Tustin e mapeamento de polos e zeros para a discretização das funções de transferências obtidas no plano *s* (Fadali and Visioli, 2012).

Na terceira etapa foi realizada a validação do sistema para a escolha dos dois melhores modelos discretizados que foram utilizados para o projeto do controlador discreto, por meio do índice MRSE (*Mean Relative Squared Error*), dado por:

$$MRSE = \sqrt{\frac{\sum_{k=1}^N ((y(k) - \hat{y}(k))^2)}{\sum_{k=1}^N (y(k))^2}} \times 100\% \quad (3)$$

Na quarta etapa, os cálculos dos parâmetros dos controladores PI e PID foram realizados de acordo com a função de transferência do modelo contínuo (em *s*), utilizando os métodos de Ziegler-Nichols e

CHR sem sobressinal para sistemas de primeira ordem com atraso (O'Dwyer, 2012) e o Johnson e Moradi (Johnson and Moradi, 2005) para sistemas de segunda ordem com atraso. Para a ação integral foi utilizada o método de Tustin e para ação derivativa, o mapeamento de polos e zeros.

Na última etapa, foi implementado experimentalmente os dois controladores escolhidos da etapa de simulação.

5 Resultados

Nesta seção são apresentados os resultados das etapas de identificação do modelo matemático para o robô móvel DaNI 2.0, sua discretização, simulação e os parâmetros calculados dos controladores PI e PID contínuo e discreto desenvolvido, assim como a implementação prática do sistema.

5.1 Identificação de Modelos Matemáticos

Um sinal do tipo degrau com amplitude entre 10 rpm e 100 rpm, dentro da faixa de velocidade de trabalho escolhida, foi aplicado às duas rodas do robô móvel DaNI 2.0 em malha aberta, com sua resposta ilustrada na Figura 4.

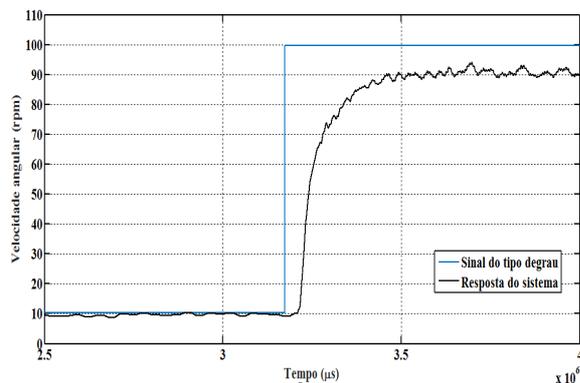


Figura 4. Resposta do sistema ao sinal do tipo degrau em malha aberta.

A partir da curva de resposta ao sinal do tipo degrau (Figura 4) foram aplicados os métodos de Ziegler-Nichols em malha aberta, de Smith em 1ª ordem, de Sundaesan, de Broida e de Mollenkamp para obter as funções de transferências que representassem o comportamento dinâmico do sistema na faixa de trabalho de interesse. A Tabela 1 ilustra as funções de transferências obtidas, destacando que são bem semelhantes para ambas as rodas, o que possibilitou usar apenas uma como modelo matemático.

Por meio do índice RMSE foi observado que as curvas que mais se aproximaram do sinal de referência foram as de Smith de 1ª ordem e de Mollenkamp, com índices RMSE de 5,85% e 5,62%, respectivamente. A Figura 5 ilustra os resultados obtidos com os métodos de Smith de 1ª ordem e de Mollenkamp, comparados com o dado experimental de saída.

Tabela 1. Funções de transferência obtidas por testes em malha aberta.

Método	Função de transferência
Ziegler-Nichols	$Gp(s) = \frac{0,8962}{0,0543s + 1} e^{-0,04s}$
Sundaesan	$Gp(s) = \frac{0,8962}{0,0658s + 1} e^{-0,0293s}$
Broida	$Gp(s) = \frac{0,8962}{0,0308s + 1} e^{-0,0443s}$
Smith 1ª Ordem	$Gp(s) = \frac{0,8962}{0,0422s + 1} e^{-0,0404s}$
Mollenkamp	$Gp(s) = \frac{0,8962}{(0,0478s + 1)(0,004s + 1)} e^{-0,0304s}$

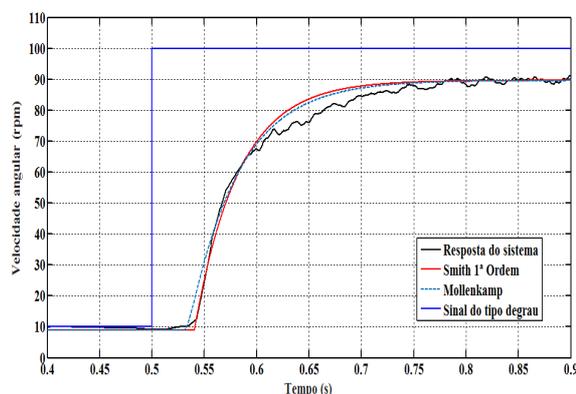


Figura 5. Identificação do sistema por curva de reação utilizando os métodos de Smith de 1ª ordem e de Mollenkamp.

5.2 Discretização do Sistema Dinâmico

As funções de transferência identificadas por Smith de 1ª ordem e por Mollenkamp foram discretizadas pelos métodos de ZOH, Tustin e mapeamento de polos e zeros, com o tempo de amostragem de 20 ms, de acordo com o método de Isermann.

O método de ZOH foi o que obteve o melhor índice RMSE para os dois sistemas discretizados sendo, portanto, o escolhido como aproximação discreta para o projeto do sistema de controle discreto. As Figuras 6 e 7 ilustram as aproximações em tempo discreto para os métodos de Smith de 1ª ordem e para Mollenkamp, respectivamente, onde o índice RMSE foi de 4,02% para o primeiro com o ZOH e de 1,93% para o segundo também com ZOH.

A Tabela 2 ilustra as funções de transferência discretizadas dos sistemas escolhidos devido ao valor do índice RMSE para ilustrar o comportamento dinâmico do robô móvel na faixa de trabalho escolhida, lembrando que as funções de transferências estão com ordem superior ao do método devido à aproximação de Padé de ordem 1.

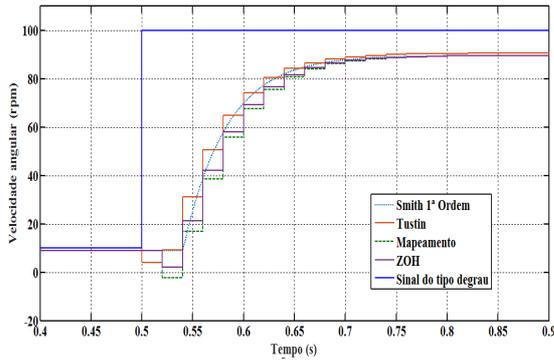


Figura 6. Curva de reação do sistema em malha aberta e as aproximações em tempo discreto para o método de Smith de 1ª ordem.

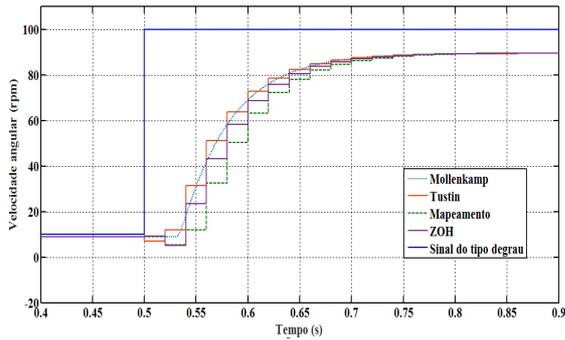


Figura 7. Curva de reação do sistema em malha aberta e as aproximações em tempo discreto para o método de Mollenkamp.

Tabela 2. Funções de transferências em tempo discreto escolhidas pelo índice RMSE.

Método	Função de transferência
Smith de 1ª ordem - ZOH	$Gp(z) = \frac{-0,0003z^2 - 0,0744z + 0,2873}{z^2 - 0,9941z + 0,2313}$
Mollenkamp - ZOH	$Gp(z) = \frac{0,002z^3 - 0,0473z^2 + 0,2460z + 0,0221}{z^3 - 0,9331z^2 + 0,1828z - 0,00119}$

5.3 Projeto do Controlador PI/PID

O sistema dinâmico em malha aberta apresentou um erro de regime permanente de aproximadamente 10%, o que torna necessário o projeto de um controlador para que esse erro seja reduzido.

O controlador PI/PID foi projetado para o controle de velocidade baseado nos modelos discretizados ilustrados pelas funções de transferência da Tabela 2. O controlador PI/PID para o sistema de primeira ordem por Smith pode ser calculado por meio da Equação (5), utilizando a aproximação de Tustin para a ação integral e o mapeamento de polos e zeros para a ação derivativa.

$$PID(z) = K_p \left[1 + \frac{T}{2T_i} \frac{(z+1)}{(z-1)} + \frac{(z-1)}{(z - e^{-\frac{T}{NT_d}})} \right] \quad (5)$$

onde: K_p é o ganho da ação proporcional; T_i e T_d são os tempos integral e derivativo, respectivamente; T é o tempo de amostragem; N é o valor da constante de tempo do filtro para a ação derivativa, em que foi adotado o valor igual a 1. A sintonia dos parâmetros dos controladores PI e PID para a função de transfe-

rência do método de Smith de primeira ordem foi realizada pelos métodos de Ziegler-Nichols e CHR

Para a função de transferência do sistema de segunda ordem modelada pelo método de Mollenkamp, o método de Johnson e Moradi (Johnson and Moradi, 2005) foi utilizado, onde a arquitetura em série para o controlador PID é adotada. Dessa forma, a função PID discretizada adotando a aproximação de Tustin para a ação integral e o mapeamento de polos e zeros para a ação derivativa é escrita como:

$$PID(z) = K_p \left(1 + \frac{T}{2T_i} \frac{(z+1)}{(z-1)} \right) \left[\frac{\left(1 - e^{-\frac{T}{\tau_f}} \right) \left(z - e^{-\frac{T}{T_d}} \right)}{\left(1 - e^{-\frac{T}{\tau_f}} \right) \left(z - e^{-\frac{T}{\tau_f}} \right)} \right] \quad (6)$$

com τ_f sendo uma constante para tornar o controlador realizável.

Os valores dos parâmetros dos controladores em s , em função dos métodos escolhidos para a sintonia estão ilustrados na Tabela 3.

Tabela 3. Valores dos parâmetros de sintonia dos controladores PI/PID e os métodos de sintonia utilizados.

Método/sintonia	K_p	T_i (s)	T_d (s)
PI Smith/Ziegler-Nichols	1,049	0,1333	-
PI Smith/ CHR sem sobressinal	0,4079	0,04895	-
PID Smith/Ziegler-Nichols	1,3986	0,0808	0,0202
PID Smith/ CHR sem sobressinal	0,6993	0,0422	0,0202
PID Mollenkamp/Johnson & Moradi	1,1143	0,0552	0,004

O valor de τ_f para o controlador PID com sintonia pelo método de Johnson & Moradi é 0,304.

Diante dos resultados obtidos para os controladores PI e PID discretos das funções de Smith de 1ª ordem e Mollenkamp, verificou-se nas simulações que em todos os sistemas de malha fechada, o erro de regime é anulado, ou seja, há o controle de velocidade desejado. Porém apenas dois desses sistemas foram escolhidos para a implementação prática na plataforma robótica.

O primeiro critério para seleção dos dois melhores foi a variação de velocidade no regime transitório da resposta ao degrau, ou seja, a aceleração das rodas. O sistema em malha fechada de Smith com controlador PI por sintonia de Ziegler-Nichols apresenta um pico de aceleração no regime transitório de aproximadamente $(-178,6 \frac{RPM}{s^2})$, visto na Figura 8, ao passo que o mesmo sistema sintonizado por CHR sem sobressinal não apresenta nenhum pico de aceleração no regime transitório, exibindo uma resposta mais suave ao degrau, conforme ilustra a Figura 9.

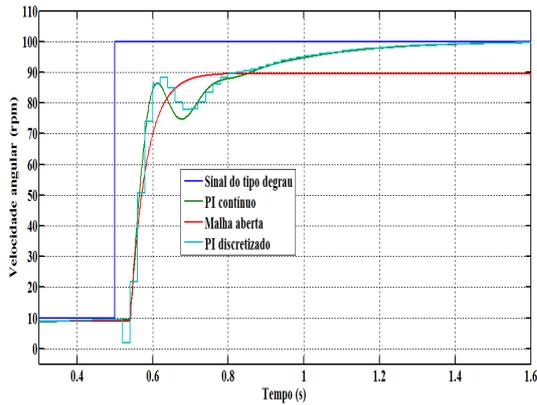


Figura 8. Simulação do controlador PI-Smith de 1ª ordem discretizado por ZOH com sintonia de parâmetros por Ziegler-Nichols.

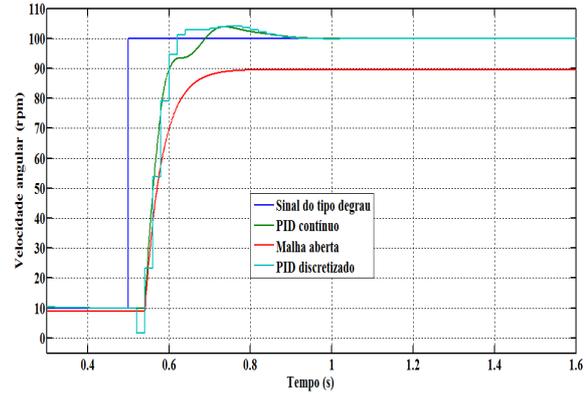


Figura 11. Simulação do controlador PID-Smith de 1ª ordem discretizado por ZOH com sintonia de parâmetros por CHR sem sobressinal.

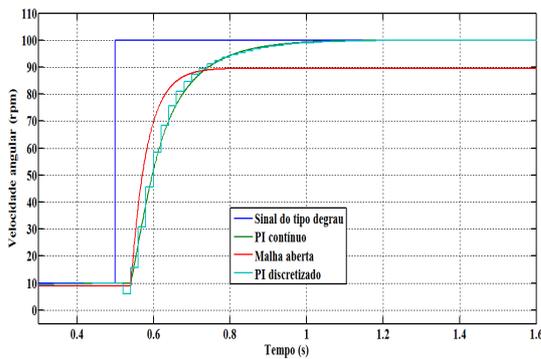


Figura 9. Simulação do controlador PI-Smith de 1ª ordem discretizado por ZOH com sintonia de parâmetros por CHR sem sobressinal.

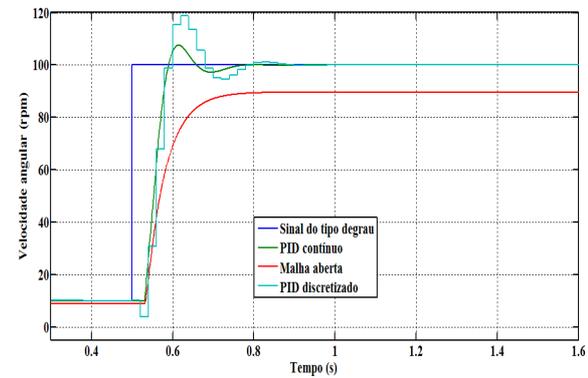


Figura 12. Simulação do controlador PID-Mollenkamp discretizado por ZOH com sintonia de parâmetros por Johnson & Moradi.

O sistema em malha fechada de Smith com controlador PID por sintonia de Ziegler-Nichols, apresenta muitas oscilações até a estabilidade, conforme a Figura 10, comportamento não admissível a esse projeto, já o mesmo sistema sintonizado por CHR, visto na Figura 11, apresenta um pico de aceleração de aproximadamente $(-10,36 \frac{RPM}{s^2})$ próximo do regime permanente e o sistema em malha fechada de Mollenkamp (de segunda ordem) com controlador PID por sintonia de Johnson & Moradi, apresenta um pico de aceleração de aproximadamente $(-243 \frac{RPM}{s^2})$ próximo ao regime permanente e visto na Figura 12.

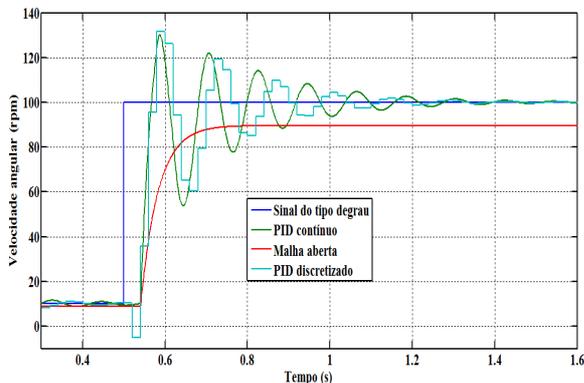


Figura 10. Simulação do controlador PID-Smith de 1ª ordem discretizado por ZOH com sintonia de parâmetros por Ziegler-Nichols.

O segundo critério para seleção dos dois melhores controladores a serem testados experimentalmente foi a verificação do percentual de *overshoot* em malha fechada, tendo em vista que o sistema não o admite, pois isso causaria um comportamento de aceleração indesejada nas rodas do robô. Nesse quesito, a Tabela 4 ilustra os percentuais de *overshoot* para os controladores projetados, observados na simulação em malha fechada.

Tabela 4. Percentuais de *overshoot* em malha fechada, para os controladores PI/PID e os métodos de sintonia dos parâmetros.

Método/sintonia	PI	PID
Smith/Ziegler-Nichols	0%	23,9%
Smith/CHR sem sobressinal	0%	3,9%
Mollenkamp/Johnson & Moradi	-	15,7%

Dessa forma, os controladores escolhidos para a fase experimental foram o PI/PID usando o modelo de Smith de 1ª ordem, discretizado por ZOH, com sintonia de parâmetros por CHR sem sobressinal, por não apresentarem *overshoot* significativo e por apresentarem picos de aceleração também pequenos, como exposto anteriormente.

5.4 Resultados Experimentais do Sistema

Na parte experimental foi desenvolvida, com o *software* LabVIEW, uma IHM (Interface Homem-Máquina) para o acompanhamento em tempo real das respostas das duas rodas da plataforma robótica NI LabVIEW Starter Kit. A Figura 13 ilustra o painel frontal desenvolvido.

Desse modo, as respostas experimentais e simuladas da velocidade angular do sistema em malha fechada com a utilização dos controladores PI e PID são ilustradas nas Figuras 14 e 15, respectivamente, com as respostas experimentais bem próximas dos resultados obtidos por simulação. Por omissão, a resposta de velocidade angular é apresentada apenas para uma das duas rodas da plataforma robótica para visualização, porém as respostas de ambas são semelhantes.

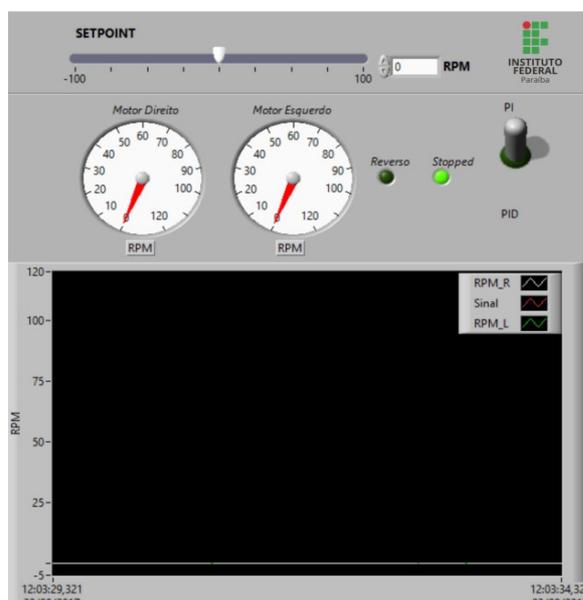


Figura 13. IHM desenvolvida em LabVIEW.

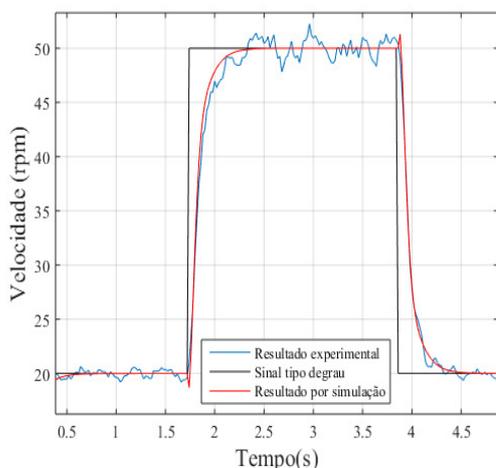


Figura 14. Respostas do sistema da plataforma robótica com o controlador PI.

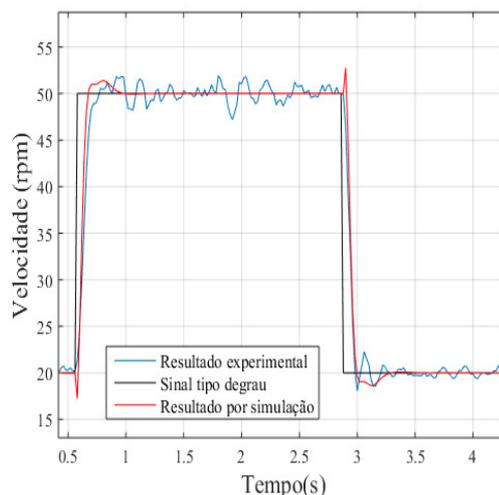


Figura 15. Respostas do sistema da plataforma robótica com o controlador PID.

6 Conclusões

Neste projeto foram apresentados os resultados obtidos com controladores PI/PID discretizados para o controle de velocidade angular das rodas a plataforma robótica DaNI 2.0, com seu controle embarcado em um FPGA.

O desenvolvimento dessa aplicação foi facilitado devido às altas taxas de amostragem da FPGA e um RTOS (*Real Time Operating System*) rodando sobre um processador de tempo real contidos na placa sbRIO-9632, inclusa na plataforma robótica, que viabilizou a aquisição do sinal de resposta ao degrau do sistema por meio dos sensores acoplados aos motores, pois a resposta transitória desse sistema é da ordem de microssegundos.

Também a utilização do bloco de função discreta no LabVIEW, e as soluções de transferência de dados entre o FPGA e o processador da SbrIO por meio de estruturas de dados do tipo FIFO (*First In First Out*) foram um diferencial imprescindível e bastante útil no desenvolvimento dessa aplicação, na medida que permitiram a aplicação da ação do controlador diretamente aos motores sem a necessidade de implementar equações de diferenças no LabVIEW, e também viabilizou a transferência dos dados de medição para serem analisados.

Trabalhos futuros estão sendo realizados para a implementação de outros tipos de controladores que serão comparados com os resultados aqui apresentados, bem como o uso de técnicas de identificação por predição do erro, que poderá aprimorar a precisão do modelo matemático obtido.

Agradecimentos

A equipe agradece ao IFPB (Instituto Federal de Educação, Ciência e Tecnologia da Paraíba) pelo apoio à pesquisa.

Referências Bibliográficas

- Almasri, M. M., Alajlan A. M. and Elleithy, K. M. (2016) Trajectory planning and collision avoidance algorithm for mobile robotics system. *IEEE Sensors Journal*, 16(12), pp. 5021-5028.
- Astrom, K. and Hagglund, T. (1995). *PID controllers: theory, design and tuning*. 2nd ed. ISA.
- Bish, R. *et al.* (2007). Mathematical modeling and computer simulation of a robotic rat pup. *Mathematical and Computer Modelling*, 45(7-8), pp. 981-1000.
- Coelho, A. A. R. and Coelho, L. S. (2004). *Identificação de sistemas dinâmicos lineares*. Ed. UFSC, Florianópolis (Brasil).
- Fadali, M. S. and Visioli, A. (2012). *Digital control engineering: analysis and design*, 2nd ed. Elsevier.
- Ferreira Filho, E. B. and Pimenta, L. C. A. (2015). Segregação de enxames de robôs heterogêneos do tipo integrador simples em múltiplos grupos usando abstrações. In: *XII Simpósio Brasileiro de Automação Inteligente (SBAI 2015)*, Natal (Brasil).
- Guerra, P. N. *et al* (2004). Linear modelling and identification of a mobile robot with differential drive. In: *I International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, Setúbal (Portugal).
- Isermann, R. (1980). Practical aspects of process identification. *Automatica*. 16 (5), pp. 575-587.
- Jazar, R. N. (2010). *Theory of applied robotics: kinematics, dynamics, and control*. 2nd ed. Springer.
- Johnson, M. A. and Moradi, M. H. (2005). *PID control: new identification and design methods*. Springer-Verlag London. Cap. 8.
- Kheir, N. A. *et al.* (1996). Control systems engineering education. *Automatica*, 32(2), pp. 147-166.
- Ljung, L. (1999). *System identification: theory for the user*. 2nd ed. Prentice Hall.
- National Instruments (2016). NI LabVIEW robotics starter kit. Disponível em: <<http://www.ni.com/datasheet/pdf/en/ds-217>>. Acesso em: nov. 2017.
- O'Dwyer, A. (2012). An overview of tuning rules for the PI and PID continuous-time control of time-delayed single-input, single-output (SISO) processes. In: R. Vilanova and A. Visioli, ed., *PID control in the third millenium: lessons learned and new approaches*, Springer, pp. 3-44.
- Okuyama, I. F. *et al* (2017). Nonlinear grey-box identification of a differential drive mobile robot. In: *XIII Simpósio Brasileiro de Automação Inteligente (SBAI 2017)*, Porto Alegre (Brasil), pp. 963-968.
- Romero, R. A. F. *et al.* (2014). *Robótica móvel*. Rio de Janeiro: LTC / Grupo Gen.
- Roy, T, Barai, R. K., and Dey, R. (2013). Identification of differentially driven wheeled mobile robot using neural networks. *International Journal of Electrical, Electronics and Computer Engineering*. 2 (2), pp. 38-45.
- Tzafestas, S. G. (2014). *Introduction to mobile robot control*. Elsevier.