II CONGRESSO INTERNACIONAL DE **GESTÃO E TECNOLOGIAS** COINTER - PDVGT 2018

DESENVOLVIMENTO DE UM CONTROLADOR HÍBRIDO FUZZY-PID E SUPERVISÓRIO PARA ESTABILIZAÇÃO DE DRONE QUADRICÓPTERO

DEVELOPMENT OF AN HYBRID FUZZY-PID CONTROLLER AND ELABORATION OF A SUPERVISORY TO STABILIZE A QUADCOPTER DRONE

Apresentação: Comunicação Oral

Daniel Q.M. Resende¹, Asafe S. Silva²; Élton F.S. Lima³; Alexander P.C. de Sena⁴

DOI: <https://doi.org/10.31692/2596-0857.ICOINTERPDVGT.2018.00002>

Resumo

Este trabalho descreve o desenvolvimento de um controlador híbrido Fuzzy-PID para o controle de voo de um Veículo Aéreo Não Tripulado (VANT) de baixo custo do tipo quadricóptero. O trabalho ainda contempla o desenvolvimento de um supervisório que permite visualizar graficamente dados de interesse em tempo real, além de possibilitar a sintonização wireless dos ganhos do controlador PID. Para estimar a orientação do conjunto no espaço tridimensional foi utilizada uma unidade inercial composta de acelerômetro e giroscópio. Na montagem do sistema embarcado foi utilizado o microcontrolador Atmel SAM3X8E ARM Cortex-M3. O protótipo foi comandado por comunicação à distância por meio de radiofrequência.

Palavras-Chave: Quadricóptero, VANT, Lógica Fuzzy, Controle PID, Estabilização, Supervisório.

Abstract

This paper describes the development of a hybrid Fuzzy-PID flight controller of a low cost Unmanned Aerial Vehicle (UAV) of type quadrotor aircraft. This article also presents the development of an interface that allows a graphic visualization of the data of interest as well as enables the wireless tuning of the PID controller's gains. In order to estimate the system's orientation in the tridimensional space, an inertial unit composed by an accelerometer and a gyroscope was utilized. In the assembly of the embedded system was used the microcontroller named Atmel SAM3X8E ARM Cortex-M3. The prototype was commanded by means of radiofrequency communication.

Keywords: Quadrotor Aircraft, UAV, Fuzzy Logic, PID Control, Stabilization, Interface.

¹ Engenharia Mecânica, Instituto Federal de Pernambuco, asafesilva01@gmail.com

² Engenharia Mecânica, Instituto Federal de Pernambuco, dqmresende413@gmail.com

³ Engenharia Mecânica, Instituto Federal de Pernambuco, elton.franklin.lima@gmail.com

⁴ Doutor, Instituto Federal de Pernambuco, alexander.sena@caruaru.ifpe.edu.br

Introdução

O interesse em relação às pesquisas sobre Veículos Aéreos não Tripulados (VANTs) vem aumentando de forma significativa nos últimos anos, principalmente devido à vasta gama de aplicações que vão desde o uso para o entretenimento até a utilização no campo civil e militar. Os VANTs podem atuar de forma totalmente autônoma ou controlada remotamente. Dentre eles, os quadricópteros vêm ganhando destaque, onde é possível perceber uma grande quantidade de pesquisas focadas na modelagem dinâmica e métodos de controle automático para estabilização e regulação de voo (Sá, 2012).

De acordo com Suzuki (2009) apud Moraes (2017), a dificuldade na construção do controlador de um quadricóptero, está no fato dele apresentar uma natureza instável, sendo também susceptível a perturbações externas, como rajadas de vento, além de ser um sistema subatuado, ou seja, o número de atuadores (quatro) é menor que o número de graus de liberdade (seis), conseqüentemente a movimentação no espaço tridimensional depende da correlação de dois ou mais atuadores. Existem inúmeras vantagens no uso de quadricópteros quando comparado a outros tipos de VANTs. Entre elas, Moraes (2017) cita a capacidade de locomoção em locais com pouco espaço, a possibilidade de realizar manobras precisas em baixa e alta velocidade, poder levantar voo vertical, aterrissar em locais de difícil acesso e sua capacidade de realizar voos pairados.

Neste contexto, este trabalho apresentará o desenvolvimento de um controlador híbrido Fuzzy-PID para o controle de voo de um VANT tipo quadricóptero, utilizando para este objetivo, um supervisor para a sintonização dos ganhos do controlador em tempo real.

Fundamentação Teórica

Segundo Ogata (2011) o controlador PID (Proporcional-Integral-Derivativo), como o próprio nome sugere, é uma técnica que combina as três ações de controle, proporcional, integral e derivativo. A combinação dessas ações fornece as vantagens individuais de cada uma das três técnicas, juntando todas em um único controlador. O funcionamento do PID discretizado é regido pela Eq. (1).

$$u(t) = K_p e(t) + u_i(kT) + K_d e(t)T + K_d(t) \frac{e(t) - e(kt)}{T} \quad (1)$$

Onde K_p é o ganho proporcional do sistema; K_i é o ganho integral do sistema; K_d é o ganho derivativo do sistema; T é o período de amostragem; $e(t)$ é o erro atual do sistema; $e(kT)$ é o erro no instante anterior ao atual; $ui(kT)$ é o valor integral no instante anterior ao atual e $u(t)$ é a saída PID propriamente dita do sistema.

Este controlador é bastante popular em função de sua simplicidade e aplicabilidade em diversos sistemas. Outra vantagem destes controladores, é que regras empíricas podem ser aplicadas, tais como Ziegler-Nichols, permitindo o ajuste dos parâmetros do controlador sem a necessidade do conhecimento do modelo matemático do sistema. O uso deste tipo de controlador se baseia na consideração de que, em torno do ponto típico de operação, um sistema se comporta de maneira aproximadamente linear. Esta premissa funciona bem em grande parte dos processos, para uma determinada janela de tempo e operação, a partir da qual normalmente se faz necessária uma ressintonia do controlador (Carvalho, 2010).

O controle fuzzy pertence ao grupo dos controladores baseados em Inteligência Artificial (I.A.) por representar um novo paradigma dentro da engenharia de controle, baseado em transferência do conhecimento sobre um problema. Ao contrário dos controladores clássicos, o controlador fuzzy dispensa o conhecimento do modelo matemático da planta a ser controlada (Xavier Filho, 2008). Um controlador fuzzy é um sistema baseado em regras que interagem para produzirem o comportamento desejado para o sistema. As regras são avaliadas simultaneamente, ou em paralelo, onde a saída do controlador é obtida interpolando-se as ações recomendadas por cada regra. O controle fuzzy tem a capacidade de considerar vários critérios de desempenho simultaneamente, tanto na forma matemática como na forma linguística. Segundo Campos e Saito (2004), os objetivos a serem alcançados por um controlador fuzzy são os seguintes: controlar e operar automaticamente processos complexos, não-lineares e multivariáveis, com desempenho pelo menos igual ao dos operadores; respeitar as especificações e restrições operacionais; ser simples, robusto e operar em tempo real. O VANT quadricóptero é um sistema complexo e não-linear, portanto a obtenção de seu modelo matemático requer uma série de parâmetros difíceis de serem obtidos. Uma apresentação detalhada sobre Fuzzy pode ser encontrada em Simões e Shaw (2007).

Metodologia

Um VANT do tipo quadricóptero possui uma estrutura constituída de quatro braços simetricamente distribuídos, onde nas extremidades estão localizados os quatro motores com

suas respectivas hélices. A disposição dos motores elimina o efeito do torque gerado por eles (Sá, 2012). Portanto, para seu funcionamento correto é necessário que dois motores adjacentes girem em sentidos contrários fazendo com que a plataforma fique em equilíbrio quando os motores estiverem com velocidades angulares iguais. A rotação de cada motor é controlada de forma independente, para que possa ser gerado o torque e o impulso necessário para movimentar a estrutura. Um quadrrorotor possui quatro movimentações básicas: Thrust, Pitch, Rolle e Yaw. Seguindo as numerações dos motores apresentadas na Fig. (1), o movimento de thrust ocorre quando os quatro motores aumentam a velocidade igualmente. O movimento de pitch ocorre quando a velocidade dos motores 1 e 3 estão diferentes (sempre que a velocidade do 3 é maior que a do 1, ele se move para frente, no contrário, se move para trás). O movimento de roll acontece quando as velocidades de 2 e 4 são distintas (no caso em que a velocidade do 4 é maior que a do 2, ele se move para direita, no contrário, se move para esquerda). E por fim, o movimento de yaw ocorre quando as velocidades dos motores 1 e 3 aumentam ou diminuem em relação aos motores 2 e 4, fazendo com que ele rotacione em torno de si, no eixo perpendicular aos seus “braços”.

Figura 1. Modos de movimentação da plataforma. Fonte: Adaptada de: Romero et al., 2014.

O subsistema de sensoriamento é composto de dois sensores inerciais encapsulados em um mesmo chip (MPU-6050) - um giroscópio, responsável pela medição da velocidade angular em torno dos três eixos cartesianos, e um acelerômetro capaz de medir acelerações lineares nestes eixos. Para atenuação de ruídos nos sinais dos sensores, testaram-se os filtros complementares, de Kalman e de Madgwick. O filtro que obteve o melhor desempenho foi do tipo complementar, que funciona como um passa-baixa para o acelerômetro, e um passa-alta para o giroscópio. Desta forma, realizando o tratamento apropriado e a fusão dos dados destes sensores, foi possível a implementação da Unidade de Medição Inercial (Inertial

Measurement Unit, ou IMU), que estrategicamente foi posicionada o mais próximo do centro de massa do veículo.

O subsistema de processamento é o responsável por realizar a aquisição, filtragem e fusão dos dados advindos dos sensores inerciais, bem como calcula a ação de controle com o objetivo de estabilizar o veículo no ar. Neste trabalho foi utilizado um Atmel SAM3X8E ARM Cortex-M3, que possui um núcleo de 32 bits com 84 MHz de clock.

O subsistema de atuação é o responsável pela recepção dos sinais provenientes do controlador e acionamento dos atuadores. Os motores utilizados são do tipo brushless (sem escovas), os quais possuem uma relação de 1000RPM/V e uma corrente máxima de 16A. Tais motores são alimentados indiretamente por uma bateria de Li-Po (11,1VCC), capaz de fornecer 5200mAh. O subsistema de atuação é composto por controladores eletrônicos de velocidade (Electronic Speed Controller, ou ESC), que recebem sinais PWM (Pulse Width Modulation) do microcontrolador, e geram um sinal “trifásico” para os motores, obtendo assim, uma relação proporcional entre o sinal de PWM e a velocidade angular. O protótipo desenvolvido possui a estrutura FY 450 apresentada na Fig. (2).

Figura 2. Plataforma utilizada na pesquisa. Fonte: Autoria própria.

Sistema de Controle

Em tese, a implementação do controlador Fuzzy combinada ao controlador PID poderia apresentar uma solução eficaz ao problema de não linearidade do sistema. Diferentemente dos tipos de controladores Fuzzy-PID apresentado em Suzuki (2009), nesta pesquisa optou-se por inserir o controlador Fuzzy atuando como a “parcela proporcional” do PID, também atribuindo um ganho ao mesmo, conforme a Fig. (3). As duas malhas de controle atuam na movimentação Pitch e Roll e possibilitaram a sua estabilização e o seu deslocamento horizontal. Diferentemente de outras pesquisas da literatura, neste trabalho o controlador Fuzzy-PID foi responsável por controlar a velocidade angular do quadricóptero,

ao invés de controlar diretamente o ângulo. Esta escolha se deu pelo fato dos sinais da unidade inercial possuírem um ruído que atrapalhava o controlador devido a sensibilidade do acelerômetro. Com esta abordagem alternativa, o sinal principal passa a ser o do giroscópio, que é menos suscetível às vibrações da estrutura. Portanto o sinal de erro, que é a diferença entre o setpoint e a variável a ser controlada, é medido em graus/s.

Figura 3. Diagrama de bloco do controlador híbrido Fuzzy-PID aplicado. Fonte: Autoria própria.

Em relação à construção das funções de pertinência da lógica Fuzzy, foi utilizada a toolbox de lógica Fuzzy do software Matlab[®]. Para a entrada “erro”, foram utilizadas sete funções de pertinência do tipo trapezoidal, com as variáveis linguísticas: ErroNegativoAlto (ENA), ErroNegativoMédio (ENM), ErroNegativoBaixo (ENB), ErroZero (EZ), ErroPositivoBaixo (EPB), ErroPositivoMédio (EPM) e ErroPositivoAlto (EPA). A Fig. (4) apresenta as funções de pertinência da entrada erro.

Figura 4. Funções de pertinência da entrada do controlador. Fonte: Autoria própria.

Para a saída do controlador Fuzzy também foram utilizadas sete funções de pertinência do tipo trapezoidal. As variáveis linguísticas adotadas foram: PwmNegativoAlto (PNA), PwmNegativoMédio (PNM), PwmNegativoBaixo (PNB), PwmZero (PZ), PwmPositivoBaixo (PPB), PwmPositivoMédio (PPM) e PwmPositivoAlto (PPA). A Fig. (5) apresenta as funções de pertinência da saída.

Figura 5. Funções de pertinência da saída do controlador. Fonte: Autoria própria.

As regras do controlador Fuzzy do tipo SE-ENTÃO foram elaboradas de acordo com o conhecimento empírico do funcionamento do sistema, e são apresentadas na Tab. (1).

Tabela 1. Regras desenvolvidas para o controlador Fuzzy. Fonte: Autoria própria.

se ERRO=ENA então PWM=PNA
se ERRO=ENM então PWM=PNM
se ERRO=ENB então PWM=PNB
se ERRO=EZ então PWM=PZ
se ERRO=EPB então PWM= PPB
se ERRO=EPM então PWM=PPM
se ERRO=EPA então PWM=PPA

Para auxiliar a programação do controlador Fuzzy no sistema embarcado foi utilizada

a biblioteca eFLL (Embedded Fuzzy Logic Library) que implementa na linguagem de programação C++, as funções para os cálculos das etapas de fuzzificação, inferência Fuzzy e defuzzificação.

Sistema de Supervisório

O processo de ajuste dos ganhos é chamado de sintonização do controlador, onde, segundo Ogata (2011) essa sintonia pode ser realizada manualmente por meio de tentativa e erro, através do conhecimento heurístico do sistema ou aplicando alguma das técnicas matemáticas desenvolvidas que auxiliam o programador a encontrar mais rapidamente os valores de ajustes necessários.

Diversos autores apresentam técnicas para o projeto de controladores, porém a sua maioria é dependente do modelo matemático da planta. De acordo com Miranda (2014), exemplos dessas variáveis são a matriz de inércia do dispositivo, os ganhos das curvas estáticas e dinâmicas de cada motor e momentos gerados por eles, entre outros. Segundo Moraes (2017), nesses casos é necessário recorrer a abordagens experimentais de sintonia de controladores. Foi desenvolvido um software com uma interface para o projetista, que possibilita a realização da sintonização dos ganhos em tempo real, além de diversas outras funções que facilitam este processo. Este software, chamado de “Drone Tuner”, foi desenvolvido na linguagem de programação Java, com o auxílio do Ambiente de desenvolvimento Netbeans. A Fig. (6) apresenta a interface do software desenvolvido.

Figura 6. Interface desenvolvida para o processo de sintonização. Fonte: Autoria própria.

Este software se comunica com o sistema embarcado através da tecnologia Bluetooth, permitindo o envio sem fio e em tempo real dos ganhos, ao mesmo tempo em que recebe os dados de interesse para a visualização de gráficos também em tempo real. O software é basicamente dividido em três partes. A primeira se destina ao envio dos ganhos para os controladores dos três movimentos básicos, conhecidos como Yaw, Pitch e Roll. A segunda é referente à possibilidade da visualização gráfica dos dados sensoriais, também dos três graus de liberdade, além da visualização das velocidades dos quatro motores em porcentagem. O terceiro recurso é relacionado ao armazenamento dos dados (sensores, motores e ganhos) em arquivos Excel, permitindo uma posterior análise do controlador através de gráficos. Com isto foi possível tornar o processo muito mais eficiente, diminuindo o tempo do projeto.

Resultados e Discussão

Os experimentos foram realizados testando-se individualmente o ângulo Pitch e Roll. Para isso, o quadricóptero foi fixado de maneira que só era possível a rotação em um dos eixos.

Para o operador movimentar o quadricóptero, o setpoint era alterado através do controle por radiofrequência. O sinal recebido pelo controle era um valor que possuía uma variação de 1000 a 2000. Para eliminar problemas de ruído quando o stick era solto pelo operador, aplicou-se uma faixa morta de ± 8 no valor central, além de converter essa faixa de 1000 a 2000 para -500 a +500. Portanto, após a aplicação da faixa morta, a variação do sinal era de -492 até +492.

Para o quadricóptero realizar os movimentos horizontais determinou-se um ângulo máximo de $\pm 10^\circ$. No entanto, a faixa de valores medidos pelos sensores inerciais precisava se igualar ao mesmo range do setpoint, logo foi necessária a multiplicação do ângulo por 49,2. Por fim, com os dois valores na mesma faixa, foi possível obter o sinal do setpoint propriamente dito, subtraindo-os e dividindo por três para se obter um valor máximo de velocidade angular de $164^\circ/\text{s}$, escolhido experimentalmente. Desta forma quando o stick era movido para o máximo positivo, por exemplo, o valor do setpoint crescia, e à medida que a plataforma se aproximava dos 10° , o valor do setpoint retornava à zero para conseguir a manutenção deste ângulo. A sintonização dos ganhos do controlador PID foi realizada através de uma série de passos. Primeiro a constante proporcional foi elevada até que o sistema assumisse uma pequena oscilação, então a constante derivativa era aumentada possibilitando

uma melhora significativa na estabilização, porém ainda apresentando um pequeno offset entre o setpoint e a posição real do sistema. Então, a constante integrativa era gradativamente testada, até a eliminação do offset. Os ganhos encontrados foram $K_p = 0,15$; $K_i = 0,003$ e $K_d = 0,01$. Uma vez encontradas as constantes que apresentaram o melhor comportamento, foram realizados dois experimentos, onde foi testado o sistema em duas situações distintas. No primeiro, com o sistema estável era aplicado um distúrbio, e no segundo, o operador aplicava uma mudança de setpoint através do controle remoto.

Resposta ao Distúrbio Externo

Quando exposto a uma perturbação, o sistema apresentou o comportamento apresentado na Fig. (7). Neste experimento, o sistema foi submetido a dois distúrbios, onde foi induzido ao desequilíbrio nas duas direções. Na Fig. (7), é possível observar que o setpoint varia rapidamente de forma a compensar o erro causado pelo distúrbio. É possível observar o mesmo comportamento para o Pitch, conforme apresentado na Fig. (8). Nestes testes, o controlador apresentou uma ação eficiente contra distúrbios externos, retornando sempre ao setpoint inicial rapidamente.

Figura 7. Experimento com aplicação de distúrbio no ângulo Roll. Fonte: Autoria própria.

Figura 8. Experimento com aplicação de distúrbio no ângulo Pitch. Fonte: Autoria própria.

Resposta à Mudança de Setpoint

O gráfico da Fig. (9) apresenta a alteração do setpoint do ângulo Roll através do Stick até o ângulo máximo (10°) e posteriormente o retorno até o estado inicial (ângulo de 0°). Pode-se perceber o funcionamento do controlador por meio da velocidade angular do sistema. Primeiramente o setpoint cresce no sentido negativo e na medida em que o ângulo se aproxima dos 10° o setpoint volta à zero, para que ele se mantenha estável. Logo depois o Stick retorna à posição inicial, fazendo com que o setpoint cresça no sentido positivo. E na medida em que o ângulo se aproxima de 0° , o setpoint retorna à zero, retornando ao seu estado inicial. Pode-se observar o mesmo comportamento no Pitch, como apresentado na Fig. (10).

Figura 9. Experimento com mudança de setpoint no ângulo Roll. Fonte: Autoria própria.

Figura 10. Experimento com mudança de setpoint no ângulo Pitch. Fonte: Autoria própria.

Os ensaios realizados também comprovaram a eficiência do controlador na variação do setpoint, onde o sistema respondeu rapidamente e se manteve estável. Nos ensaios com variação de setpoint, a entrada do sistema apresenta o mesmo sentido do setpoint, diferentemente da aplicação do distúrbio, onde a entrada cresce em sentido contrário.

Conclusões

Neste trabalho foi apresentado o desenvolvimento de um controlador híbrido Fuzzy-PID, assim como o desenvolvimento de um supervisor para monitoramento do sistema e sintonização dos ganhos do controlador. A implementação do controlador híbrido dispensou o modelamento matemático do sistema além de mostrar uma grande eficiência nas ações de estabilização e na resposta a distúrbios. O supervisor possibilitou a análise precisa do sistema durante e após operação e trouxe praticidade para a estratégia de sintonização dos ganhos. Como trabalho futuro pretende-se integrar um barômetro e uma bússola digital ao sistema e através da interpretação de seus sinais, realizar o controle de altitude e o controle do ângulo Yaw do quadricóptero.

Referências

- CAMPOS, M. e SAITO, K. (2004). **Sistemas Inteligentes em Controle e Automação de Processos**. Rio de Janeiro, Editora Ciência Moderna.
- CARVALHO, M.A. (2010). Controlador Neurofuzzy com Aprendizado On-line: Teoria e Aplicação na Indústria de Petróleo. **Dissertação** (Mestrado em Engenharia Elétrica). Universidade Federal de Minas Gerais.
- MIRANDA, C.S. (2014). Sintonia Automática dos Parâmetros de um Controlador para um Quadricóptero de Modelo Desconhecido em Voo Pairado. **Dissertação** (Mestrado em Engenharia Mecânica) Universidade Estadual de Campinas. Campinas, São Paulo.
- MORAIS, E.R. (2017). Desenvolvimento de um Controlador para Estabilização de um VANT do Tipo Quadricóptero na Realização de um Voo Autônomo Vertical. **Dissertação** (Mestrado em Sistemas de Comunicação e Automação), Universidade Federal Rural do Semi-árido, Rio Grande do Norte.
- OGATA, K. (2011). **Engenharia de Controle Moderno**. 5. ed. Pearson.
- SÁ, R.C. (2012). Construção, Modelagem Dinâmica e Controle PID para Estabilidade de um Veículo Aéreo Não Tripulado do Tipo Quadricóptero. **Dissertação** (Mestrado em Engenharia de Teleinformática), Universidade Federal do Ceará, Fortaleza.
- SIMÕES, M.G. e SHAW, I.S. (2007). **Controle e Modelagem Fuzzy**. Editora Blucher, 2ª ed. São Paulo.
- SUZUKI, N.K.G. (2009). Proposta de uma arquitetura de controle híbrida Fuzzy-PID para a realização de manobras em VANTs. **Dissertação** (Mestrado em Computação Aplicada), Universidade do Vale do Itajaí, Santa Catarina.

XAVIER FILHO, A. (2008). Controlador Fuzzy Aplicado a Um Sistema de Controle de Posição. **Dissertação** (Mestrado em Engenharia Mecânica), UFPB, João Pessoa – PB, Brasil.