

Projeto da unidade de voo de um aerogerador com aerofólio cabeado

Juliano da Silva* Marcos Schoenknecht* Igor Silvestre*
 Marcelo De Lellis* Felipe Cabral* Alexandre Trofino*
 Vinícius Wessler**

* *Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, SC, (e-mails: jrs96.jr@gmail.com, marcos.thurow@gmail.com, igorsilvestre2005@gmail.com, marcelo.lellis@ufsc.br, felipe.gomes.cabral@ufsc.br, alexandre.trofino@ufsc.br).*

** *Departamento de Engenharia Elétrica e Eletrônica, Universidade Federal de Santa Catarina, SC, (e-mail: viniciussbc@gmail.com).*

Abstract: This paper presents the design of a flight unit for a pumping kite wind power plant, based on the AWE (*Airborne Wind Energy*) technology, developed by the UFSCkite group. The system comprises a fabric-made wing anchored to the ground by means of a tether reeled around a drum, which drives an electric machine. These two components are housed in the ground unit. Mechanical power is provided to the electric machine (thus producing electricity) as the tether is reeled out under high traction force while the wing is carried away by the wind. The flight unit, suspended next to the wing, is responsible for the flight control. The main components of this unit are discussed, including a system of distributed software modules, a SCADA system and the main hardware elements. A radiofrequency infrastructure based on multiple beacons is presented, which allows for the wing position and velocity measurement, as well as for the communication between ground and flight unit computers. The design is validated through hardware-in-the-loop tests of the flight control using an open-source simulator of pumping kites.

Resumo: Este artigo apresenta o projeto de uma unidade de voo para um aerogerador com aerofólio cabeado do tipo *pumping kite* desenvolvido pelo grupo UFSCkite. Neste sistema, uma asa flexível (aerofólio) é presa por um cabo enrolado a um carretel conectado a uma máquina elétrica. Esses dois componentes são localizados na unidade de solo do aerogerador. Potência mecânica é fornecida à máquina elétrica (e eletricidade é gerada) enquanto o cabo é desenrolado sob alta tração à medida que o aerofólio é levado pelo vento. A unidade de voo, suspensa próximo ao aerofólio, é responsável pelo controle de voo. Os principais componentes desta unidade são discutidos, incluindo um sistema de módulos de *software* distribuídos, um sistema SCADA e os principais elementos de *hardware*. Uma infraestrutura de *beacons* de radiofrequência é apresentada, que permite a medição da posição e velocidade do aerofólio, além da comunicação entre os computadores das unidades de solo e de voo. O projeto é validado com testes *hardware-in-the-loop* do controle de voo utilizando um simulador *open-source* de *pumping kites*.

Keywords: Airborne wind energy; embedded systems; hardware-in-the-loop; communication protocols; flight control.

Palavras-chave: Energia eólica com aerofólios cabeados; sistemas embarcados; hardware-in-the-loop; protocolos de comunicação; controle de voo.

1. INTRODUÇÃO

O setor energético mundial vem experimentando uma fase na qual busca-se aprimorar e inventar novas abordagens de geração baseadas em fontes renováveis, as quais precisam obedecer a critérios cada vez mais restritos de sustentabilidade. Neste contexto a energia eólica é uma das fontes que mais crescem, devendo suprir cerca de 35% da demanda global de energia elétrica até 2050 (International Renewable Energy Agency, 2019). Historicamente, o uso de turbinas de eixo horizontal montadas sobre torres tornou-se a abordagem padrão no aproveitamento da energia cinética

do vento, mas acredita-se que este modelo tenha atingido uma certa saturação quanto à sua escalabilidade, pois o progressivo aumento dos custos estruturais não compensa os ganhos energéticos, limitando a viabilidade econômica (Diehl, 2013). Assim, os aerogeradores convencionais têm seu uso comercial limitado a uma altitude aproximada de 200 metros, incapazes, portanto, de explorar o enorme potencial energético presente em altitudes maiores, onde os ventos são mais fortes e constantes (de Oliveira, 2016).

Com o objetivo de explorar a energia eólica em elevadas altitudes e em localidades onde os aerogeradores convencionais não são viáveis, vários centros de pesquisa ao redor

do mundo têm voltado suas atenções para a tecnologia AWE (sigla em inglês para *Airborne Wind Energy*). Trata-se de uma tecnologia baseada em dispositivos que extraem a energia cinética do vento enquanto mantêm-se no ar por meio de forças aerodinâmicas ou aerostáticas, ancorados ao solo por meio de um ou mais cabos. Com isso, dispensa-se a enorme estrutura mecânica das torres eólicas, reduzindo custos e facilitando o transporte e instalação em localidades mais remotas ou de difícil acesso (de Oliveira, 2016). Apesar dos significativos desenvolvimentos já realizados, a tecnologia AWE ainda encontra-se em um estágio de P&D, com uma série de desafios técnicos e burocráticos a serem superados. Por exemplo, ainda falta avaliar, por meio de testes de longa duração, a robustez operacional, segurança, aspectos de manutenção e eficiência dos protótipos. Além disso, será necessária uma regulação do espaço aéreo para operação dos sistemas AWE.

Desde o ano 2000, vários grupos de pesquisa em AWE na academia e indústria se formaram ao redor do mundo. Entre esses grupos observa-se algumas variantes da tecnologia, principalmente quanto ao local de geração (em voo ou em solo), ao tipo de aerofólio (flexível ou rígido) e à característica de geração (contínua ou cíclica). Por exemplo, a empresa Makani Power optou pela abordagem de geração em voo com asa rígida, similar a um *drone* multirrotor. A energia gerada é transmitida através de condutores elétricos contidos no cabo que prende a aeronave a uma pequena torre de lançamento. Em 2019 a empresa realizou testes com seu protótipo de 600 kW instalado sobre uma plataforma flutuante no Mar do Norte, na costa norueguesa. Entretanto o projeto foi recentemente descontinuado devido a incertezas quanto à sua viabilidade econômica como produto de mercado (Makani Power, 2020). Isto serve de alerta aos desenvolvedores AWE para que sigam abordagens de baixo custo, evitando sistemas altamente sofisticados, como pode ter sido o caso da Makani Power.

Empresas como a alemã EnerKite e as holandesas AmpyxPower e KitePower (*startup* originada na universidade TU Delft), por outro lado, optaram pelo modo de operação cíclico do tipo *pumping kite*, ilustrado na Fig. 1. Este conceito de aerogerador possui três grandes componentes: uma unidade de solo, na qual é gerada a energia elétrica, o aerofólio cabeado e a sua unidade de (controle de) voo. A força aerodinâmica sobre o aerofólio é convertida em força de tração em um cabo enrolado em um carretel na unidade de solo. Este carretel, por sua vez, é conectado a uma máquina elétrica que pode operar como motor ou gerador, de acordo com a fase de operação (geração ou recolhimento). Na fase de geração o aerofólio descreve uma trajetória similar a uma lemniscata de Bernoulli (de forma a maximizar a tração) enquanto o cabo é desenrolado do carretel, gerando-se, assim, eletricidade. Em seguida, na fase de recolhimento, o padrão de voo é modificado visando reduzir a energia necessária para se recolher o cabo de tração (da Silva, 2014). Com isso, ao final do ciclo, o saldo energético da operação é positivo.

Entre as instituições que desenvolvem a tecnologia AWE encontra-se também a UFSC por meio do projeto UFSC-kite, estabelecido em 2012, pioneiro e único na América Latina. O grupo optou pela estratégia de *pumping kite* motivado pela maior facilidade de prototipagem, operacionalização e aos custos de desenvolvimento menores, princi-

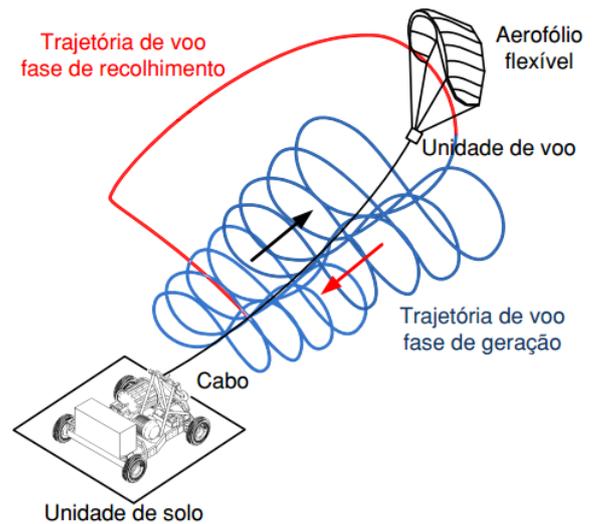


Figura 1. Conceito de aerogerador AWE do tipo *pumping kite*. Fonte: Araujo (2017).

palmente ao se empregar aerofólios flexíveis – “parapentes” de voo livre ou “pipas” de *kitesurf*.

No processo de desenvolvimento de protótipos, testes de campo permitem verificar a robustez e eficácia das soluções tecnológicas concebidas. Entretanto, a operacionalização de tais testes é bastante dificultada por fatores como a distância entre o local de testes e o laboratório, a disponibilidade de vento e a mobilização de muitas pessoas para a montagem, desmontagem e transporte de componentes. Para facilitar a execução de testes práticos, é possível realizar a integração dos componentes de *hardware* com um simulador em uma estrutura conhecida como *hardware-in-the-loop* (HIL). Nela, a dinâmica de voo do aerofólio, o qual não pode ser operado em laboratório, é simulada em computador, enquanto a parte restante do aerogerador é operada com seus reais componentes de *hardware* e *software*. Assim, pode-se configurar uma operação similar ao cenário de testes em campo, realizar testes práticos com uma frequência maior e, finalmente, acelerar o desenvolvimento dos protótipos.

Com base no exposto, o objetivo deste artigo é apresentar o projeto da unidade de voo do aerogerador com aerofólio cabeado desenvolvido pelo grupo UFSCkite. Para isto, o restante do artigo é organizado da seguinte forma: na Seção 2 são discutidos aspectos importantes de *software* e *hardware* do protótipo. Na Seção 3 discute-se o controle de voo, projetado em trabalhos anteriores. A Seção 4 é dedicada à infraestrutura de radiofrequência, utilizada para o sensoriamento da posição e velocidade do aerofólio e para a comunicação entre os microcomputadores das unidades de solo e de voo. Na Seção 5 são mostradas as funcionalidades do sistema SCADA desenvolvido para o controle supervisor e monitoramento da operação do aerogerador. O projeto realizado então é validado por meio de testes HIL em laboratório, cujo *setup* e resultados são discutidos na Seção 6. O artigo é concluído na Seção 7 com um resumo e perspectivas de trabalhos futuros.

2. UNIDADE DE VOO DO UFSCKITE

Para a unidade de controle de voo (UCV) do UFSCkite foram selecionados sensores, atuadores e componentes eletrônicos de baixo peso, visando à maximização da eficiência na extração de energia (da Silva, 2014). Em um estágio inicial de testes, dedicados ao controle de voo (sem a unidade de solo propriamente dita), optou-se por ancorar a UCV em uma pequena estrutura giroscópica no solo. Desta forma evita-se o risco de avaria dos componentes embarcados com quedas mais prováveis caso os primeiros testes já fossem com a UCV em voo. Conforme apresentado na Fig. 2, o cabo de tração é conectado a um olhal preso em uma célula de carga entre as duas metades da UCV. A montante da UCV está um pequeno mastro em cujo topo estão montados dois *encoders* rotativos. Estes sensores são movimentados por uma haste que segue o movimento do cabo de tração, de tal forma que os ângulos polar (θ) e azimutal (ϕ) do aerofólio, os quais definem a sua posição espacial, podem ser medidos.

Os dados da posição e velocidade angular obtidos dos *encoders* são realimentados ao sistema de controle embarcado, o qual atua por meio de dois servomotores em cabos de comando conectados ao bordo de fuga (parte traseira) do aerofólio, resultando em dois possíveis comandos. O comando de *steering*, ou guinada, se dá pelo enrolamento/desenrolamento antissimétrico dos cabos de controle, o que faz o aerofólio rotacionar para um de seus lados e, desta forma, consegue-se controlar a trajetória de voo. Já o comando de *depower*, ou arfagem, enrola/desenrola os cabos dos dois lados de forma igual, alterando a eficiência aerodinâmica do aerofólio e, com isto, a sua velocidade e a força de tração gerada. O mecanismo de aplicação dos comando supracitados é ilustrado na Fig. 3.

2.1 Hardware

Além dos componentes já mencionados, na UCV é utilizado um *encoder* absoluto acoplado ao eixo de cada um dos servomotores, permitindo o referenciamento (com relação a uma posição zero) dos comandos de *steering* e *depower*. Em testes de campo utiliza-se também um anemômetro de copos para a medição da direção (ângulo de azimute) e velocidade do vento. As informações do anemômetro são fundamentais, respectivamente, para centralizar a trajetória de voo na direção predominante do vento e para a determinação do potencial de geração de energia, visando a uma futura otimização da velocidade ou tração de desenrolamento do cabo na operação do *pumping kite*. As informações de posição e velocidade do aerofólio, utilizadas para o controle de voo, são obtidas por meio de uma estrutura de *beacons* de radiofrequência. Mais detalhes desta estrutura serão fornecidos na Seção 4.

A unidade de processamento utilizada é uma Raspberry Pi Zero, escolhida por apresentar tamanho e preço reduzidos, além de possuir compatibilidade com diversos protocolos de comunicação, como o SPI (*Serial Peripheral Interface*) e o I2C (*Inter-Integrated Circuit*). Nativamente os *encoders* de ângulo polar e azimutal liberam informação em paralelo, os de *steering* e *depower* utilizam protocolo SSI (*Synchronous Serial Interface*) e a célula de carga fornece um sinal analógico. Devido à diversidade de sensores, circuitos eletrônicos adicionais foram projetados para tornar



(a) Versão preliminar, ancorada em estrutura giroscópica no solo. Fonte: original.



(b) CAD da versão projetada para voo. Fonte: original.

Figura 2. Protótipos da unidade de controle de voo (UCV) do projeto UFSCkite.

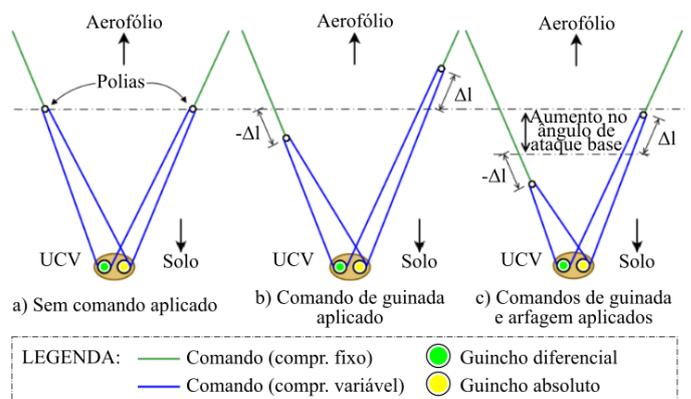


Figura 3. Comandos de guinada e arfagem do aerofólio. Fonte: original.

transparente a comunicação entre dispositivos e a unidade de processamento. Utilizando os ICs (*Integrated Circuits*) 74HC165, para serialização, MAX3294, para conversão de protocolos, e ADS1231, para digitalização, os sinais resultantes são multiplexados no barramento SPI da Raspberry.

Os servomotores são configurados nos seus *drivers*, modelos Faulhaber MCBL3006S, via comunicação serial com a unidade de processamento, sendo definidas características de proteção, limites de movimentação angular e entrada e saída de operação. A referência de posição dos servomotores é transmitida por sinais PWM em vez da interface serial, o que permitiu reduzir o período de amostragem da malha de controle de voo de aproximadamente 90 ms para cerca de 10 ms. Como a Raspberry Pi não possui interfaces PWM nativas, um circuito adicional com base no IC PCA9685 foi projetado para suprir essa demanda. A UCV é alimentada com 24 a 30 Vcc, fornecida em laboratório por uma fonte chaveada, posteriormente regulada em saídas de 24, 7, 5 e 3.3 Vcc, necessárias à alimentação dos diferentes sensores e atuadores. Para testes em campo, utiliza-se um banco de baterias de 26.4 Vcc formado por 8 células de LiFePO4, com um sistema de gerenciamento de carga. A infraestrutura de *hardware* é apresentada na Fig. 4.

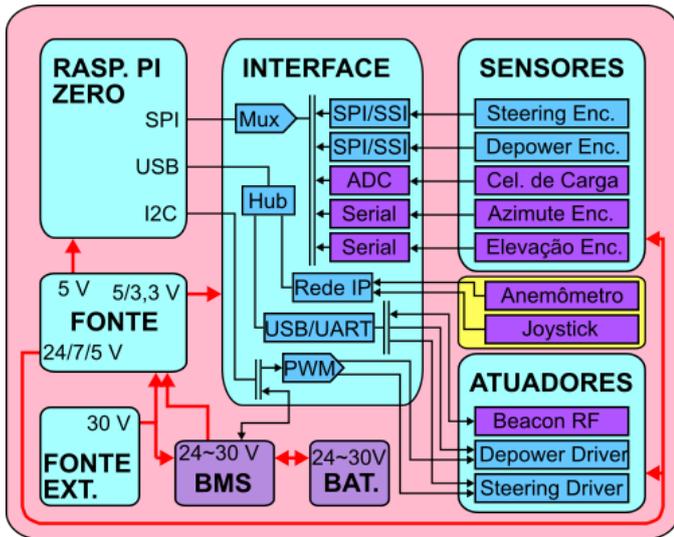


Figura 4. Estrutura modular de *hardware*. Em azul são os elementos usados no HIL; em roxo, elementos usados em campo; e em amarelo, elementos a serem acoplados à unidade de solo. Fonte: original.

2.2 Software

Sistemas AWE consistem basicamente de uma unidade de solo e uma de voo, as quais precisam trocar informações entre si para uma operação segura e eficiente. Trata-se, portanto, de sistemas de natureza intrinsecamente distribuída, com componentes fisicamente separados e acoplados por meio de conexões mecânicas e, às vezes, elétricas (Winter, 2017). Baseado nisso, o grupo UFSCkite desenvolveu uma plataforma de *software* modularizada, com módulos que se comunicam entre si através de mensagens utilizando *multicast UDP*, que permite uma menor latência e maior eficiência no uso dos recursos da rede. Os módulos são escritos em C e executados sob a forma de processos em um

sistema operacional Linux, seguindo uma estrutura padronizada. Os tipos de módulos criados são: (a) *drivers*, que fazem a comunicação direta com o *hardware*, recebendo dados dos sensores ou enviando comandos aos atuadores; (b) *processadores*, incluindo controladores, filtros de sinais e estimadores de parâmetros; (c) e *gateways*, que fazem a integração com os sistemas externos, como os ambientes de simulação e a interface gráfica.

Para a comunicação entre os módulos foi utilizada a biblioteca LCM (*Lightweight Communications and Marshalling*), a qual possibilita uma infraestrutura de troca de mensagens baseada no padrão *publish/subscribe*, com um mecanismo automático de *marshalling/unmarshalling* e geração de código sobre uma linguagem de tipo formal. Esta biblioteca oferece uma baixa latência na comunicação, característica fundamental para sistemas de controle. No processo de comunicação, a estrutura-base, formada por arquivos de configuração e bibliotecas compartilhadas por todos os módulos, implementa o padrão de comunicação empregado pela biblioteca LCM e disponibiliza dois macros: uma para o envio (*publish*) e outra para o recebimento de mensagens (*subscribe*). Assim, somente os módulos interessados em determinadas mensagens as recebem em uma *thread* dedicada. Isto permite o envio e recebimento de mensagens de forma independente, dispensando a necessidade de uma conexão direta entre módulos (Winter, 2017).

A estrutura de módulos de *software* é apresentada na Fig. 5. Os módulos específicos da unidade de voo são descritos na Tab. 1, onde os módulos processadores recebem dados de submódulos que utilizam protocolos de comunicação de baixo nível, como é o caso do módulo *SPI-interface*, pelo qual se dá a comunicação com a célula de carga e com os *encoders*. Além disso, os *drivers* dos servomotores *depower* e *steering* utilizam uma linguagem própria em *assembly*. Já na Tab. 2, são descritos os módulos da unidade de solo que foram utilizados apenas para controlar o aerofólio e supervisionar, com auxílio de um sistema supervisorio (SCADA), a operação do sistema, não tendo relação com a geração elétrica. O controle e monitoramento dos módulos é feito por meio do *Supervisor*, um sistema cliente/servidor para gerenciamento de processos e serviços de sistemas operacionais baseados em *Unix*. Ademais, é possível um monitoramento local ou remoto com interface *web*, muito conveniente para sistemas embarcados.

3. CONTROLE DE VOO

O controle de voo é estruturado com duas malhas em cascata, conforme a Fig. 6. Na malha externa (cinemática) é calculada uma referência para a direção de voo com base em duas possíveis estratégias de geração de trajetória, enquanto a malha interna (dinâmica) atua sobre o motor de *steering* buscando minimizar o erro de controle na direção do voo.

Uma das estratégias da malha externa é a lemniscata de Bernoulli, apresentada na Fig. 7. Trata-se de uma referência de trajetória contínua, que possui como parâmetros a distância focal a_l , na direção azimutal, e é centrada em um ângulo polar θ_c . A posição (θ_k, ϕ_k) do aerofólio define o seu vetor posição, \mathbf{r}_k . A posição do aerofólio é considerada como origem para os vetores velocidade tangencial do

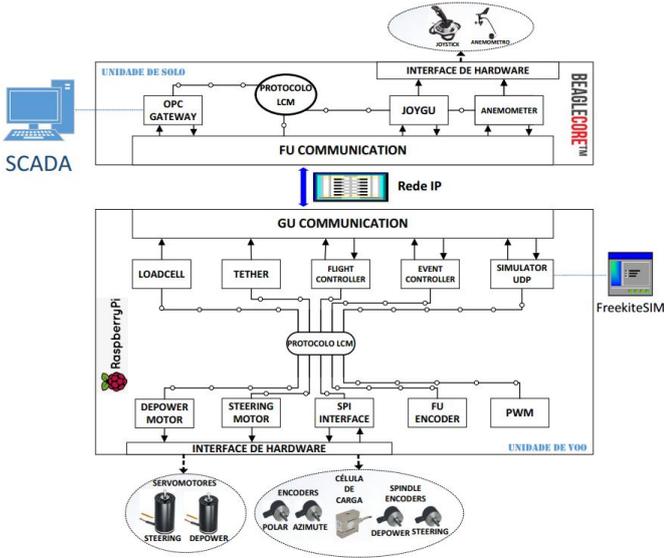


Figura 5. Estrutura modular de *software* para o aerogerador do UFSCkite. Fonte: original.

<Nome> (<categoria>): <descrição>
Event-controller (processador): a partir do monitoramento das variáveis do sistema, gera eventos discretos para controlar os estados dos demais módulos.
Flight-controller (processador): faz o cálculo das leis de controle que regem a trajetória de voo do aerofólio.
PWM (processador): converte em PWM os valores das referências de posição para os servomotores de <i>steering</i> e <i>depower</i> .
Tether (processador): trata os valores de posição e velocidade angular do cabo de tração do aerofólio pela composição e transformação das medições dos submódulos <i>Azimuth encoder</i> e <i>Elevation encoder</i> ao eixo de referência do protótipo.
Loadcell (processador): trata o valor lido pela célula de carga responsável pela medição da força de tração no cabo principal que conecta o aerofólio ao protótipo.
Depower-motor (<i>driver</i>): a partir do valor de referência recebido e do estado atual, controla a posição do motor responsável pelo comando de <i>depower</i> .
Steering-motor (<i>driver</i>): a partir do valor de referência recebido e do estado atual, controla a posição do motor responsável pelo comando de <i>steering</i> (guinada).
SPI-interface (<i>driver</i>): Configura a <i>Raspberry PI</i> para leitura dos <i>encoders</i> e célula de carga acoplados a unidade de voo.
FU-encoder (<i>driver</i>): Faz a conversão dos valores lidos em código <i>Gray</i> pelos <i>encoders</i> dos servomotores para binário, transformando-os em valores de posição, número de voltas e ângulo.
Simulator-UDP (<i>gateway</i>): Estabelece a comunicação <i>multicast UDP</i> entre o simulador <i>FreeKiteSim</i> com os módulos embarcados.
GU-communication (<i>gateway</i>): cria um pacote de mensagens com as variáveis da unidade de voo, enviando-o para a unidade de solo.

Tabela 1. Módulos da unidade de voo. Fonte: original.

aerofólio, $\mathbf{v}_{k,\tau}$, e para o vetor \mathbf{e}_θ , que aponta na direção do ângulo polar. Entre estes dois vetores é definido o ângulo de curso do aerofólio (γ), que representa a direção de voo. Busca-se então um ponto sobre a lemniscata mais próximo à posição do aerofólio, entre os quais define-se o vetor \mathbf{r}_d^* . Neste ponto particular sobre a lemniscata toma-se o vetor tangente normalizado no sentido de voo e multiplica-se a sua norma por um parâmetro (de controle) δ , resultando

<Nome> (<categoria>): <descrição>
Joygu (<i>driver</i>): faz a leitura do <i>joystick</i> responsável pelo comando manual do aerofólio.
Anemometer (<i>driver</i>): realiza a comunicação com o anemômetro responsável pela medição da velocidade e direção do vento.
FU-communication (<i>gateway</i>): cria um pacote de mensagens com as variáveis da unidade solo, enviando-o para a unidade de voo.
OPC (<i>gateway</i>): faz a comunicação com o SCADA, enviando informações de todos os módulos, mas também recebendo dados do supervisor e os encaminhando para os módulos.

Tabela 2. Módulos da unidade de solo utilizados para o controle de voo. Fonte: original.

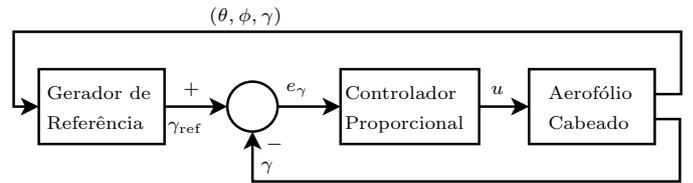


Figura 6. Controle de voo em cascata. Fonte: de Oliveira (2016).

no vetor $\delta \mathbf{r}_{t,n}^*$. Calcula-se então um vetor de referência para a direção de voo: $\mathbf{r}_r = \mathbf{r}_d^* + \delta \mathbf{r}_{t,n}^*$. O erro de controle passado à malha interna (e_γ) é então determinado como aquele entre \mathbf{r}_r e $\mathbf{v}_{k,\tau}$. Nela o erro é multiplicado por um ganho K_p (controle proporcional), resultando no sinal de controle (*steering*) aplicado ao aerofólio.

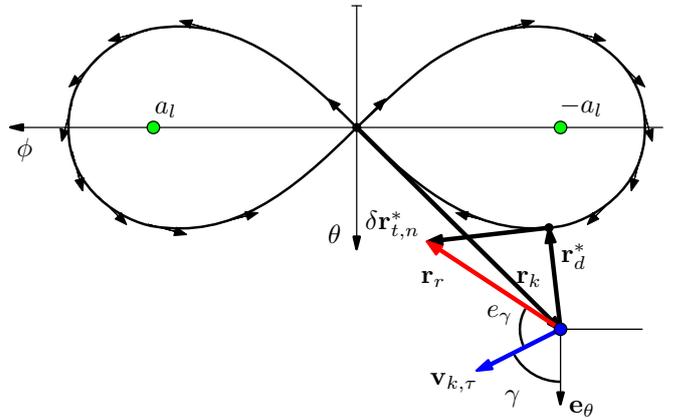


Figura 7. Trajetória de voo com a lemniscata de Bernoulli. Fonte: de Oliveira (2016).

A lemniscata tem a vantagem de produzir um sinal de controle mais suave, porém apresentava um custo computacional maior devido à necessidade de resolver um problema de otimização numérica a cada período de amostragem (80 ms ou menos) para se encontrar o vetor \mathbf{r}_d^* . Porém, recentemente este custo computacional foi drasticamente reduzido graças a uma solução algébrica para o problema de determinação de \mathbf{r}_d^* (Saraiva et al., 2019).

Pode-se utilizar também uma estratégia de geração de trajetória discreta na malha externa. Ela consiste em definir

dois pontos atratores simétricos, com coordenadas azimutais a_l e $-a_l$. Com um deles ativo por vez, calcula-se o vetor de referência \mathbf{r}_r que aponta na direção do atrator ativo partindo da posição do aerofólio, até que a coordenada azimutal do aerofólio ultrapasse a do atrator. Quando isso ocorre, o atrator atual é desativado e o outro é ativado, resultando em uma curva de aproximadamente $\pm 180^\circ$ no ângulo de curso. A estratégia com pontos atratores tem a vantagem de ser bastante simples de se implementar e computacionalmente leve, entretanto resulta em uma descontinuidade no sinal de controle na permuta de atratores, aumentando o desgaste mecânico do atuador.

4. INFRAESTRUTURA DE RADIOFREQUÊNCIA

Para o controle de voo é necessário realimentar a posição e velocidade do aerofólio. No aerogerador do UFSCkite, o sensoriamento destas grandezas é realizado através de *beacons* (transceivers) de radiofrequência (RF): cinco são instalados em solo, enquanto um é fixado na unidade de voo que acompanha o aerofólio cabeado. Esta configuração é apresentada na Fig. 8, em que quatro *beacons* são posicionados no solo em distâncias conhecidas \mathbf{r}_i , $i = 1, \dots, 4$, da unidade de solo, a qual também é equipada com um *beacon*. Cada um destes cinco *beacons* em solo trocam mensagens com o *beacon* instalado na unidade de voo e, baseado no “tempo de viagem” (*time of flight*) das ondas de RF, calcula-se as distâncias d_i . Com estes valores é executado um algoritmo de posicionamento (lateração) por meio de um *extended Kalman filter*, o que permite reduzir o efeito de ruídos na medição da posição e velocidade do aerofólio, além de implementar uma fusão de sensores com as informações angulares fornecidas pelos *encoders*.

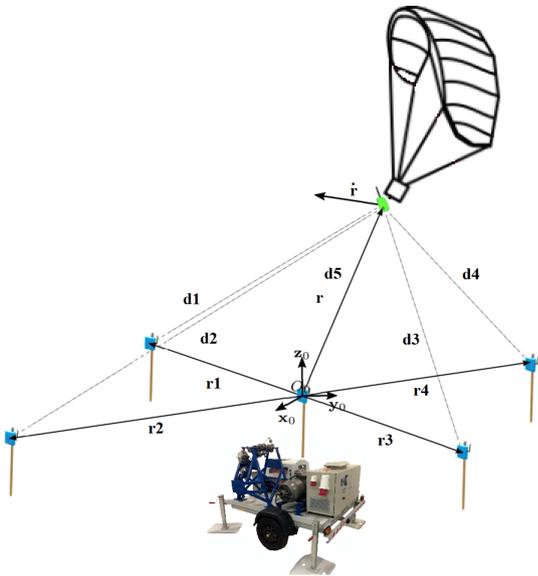


Figura 8. *Beacons* de RF para o posicionamento do aerofólio e comunicação entre unidades. Fonte: original.

Na troca de mensagens entre os *beacons* da unidade de solo e da unidade de voo para determinação da distância d_5 é possível incluir um *payload*. Isto permite a implementação de um protocolo de comunicação (camada de aplicação) ponto-a-ponto entre os computadores embarcados de ambas as unidades, o que é fundamental para a operação

sincronizada, passagem de referências de controle, ajuste de ganhos e parâmetros por parte do operador, telemetria e até mesmo controle manual do voo. Este protocolo, desenvolvido em Python, tem o comportamento de cada um dos nodos regido pela máquina de estados na Fig. 9

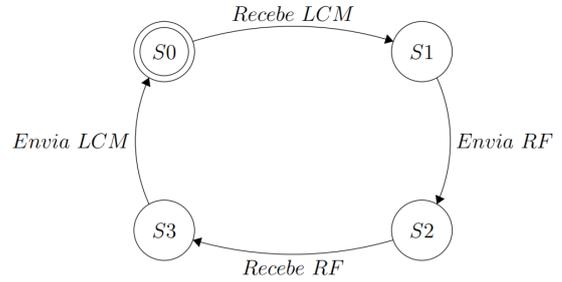


Figura 9. Máquina de estados de um nodo do protocolo de comunicação ponto-a-ponto via RF. Fonte: original.

Com base no tamanho máximo do *payload* de RF, limitado a 128 *bytes*, foram definidos quais dados seriam transportados entre as unidades e com quais resoluções. Foi estabelecido um *frame* para cada sentido de envio, conforme apresentado na Fig. 10. Por exemplo, no *frame* “gu_to_fu” (solo para voo) envia-se: *setpoints* de posição dos servomotores de *steering* e *depower*, gerados a partir de um *joystick* utilizado para o controle manual do aerofólio; coordenada azimutal ϕ_{atr} dos atratores da malha externa e ganho proporcional K_p da malha interna, etc.

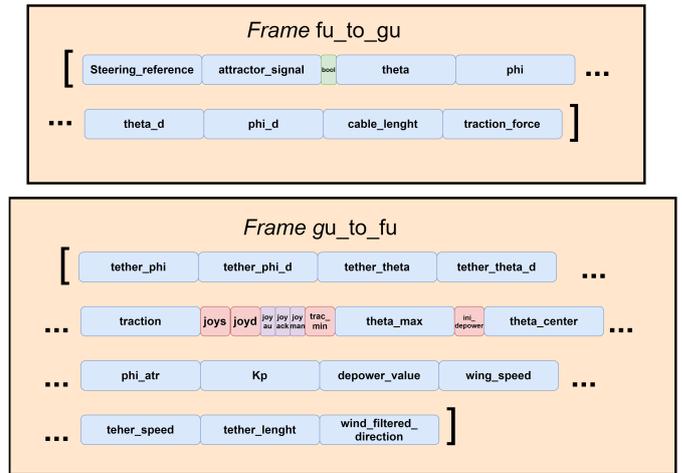


Figura 10. *Frames* “fu_to_gu” (65 Bytes, voo para solo) e “gu_to_fu” (123 Bytes, solo para voo). Fonte: original.

5. SISTEMA SCADA

Tal qual em diversas aplicações industriais, aerogeradores AWE requerem um sistema SCADA, acrônimo em inglês para “Controle Supervisório e Aquisição de Dados”. No âmbito de supervisão, o SCADA possibilita apresentar ao operador os valores de variáveis por meio de gráficos e/ou *displays*, além de permitir a configuração de alarmes e o registro de variáveis ao longo do tempo (arquivos de *log*). No que tange ao controle (supervisório), o SCADA permite ao operador definir *setpoints*, ativar e escolher entre controladores, alterar o valor de seus ganhos e atuar manualmente em certos atuadores. O *software* de SCADA

utilizado no projeto é o WinCC (fabricante Siemens), por meio do qual foram programadas diferentes telas de operação com campos interativos, a exemplo da Fig. 11. Nessas telas é possível também visualizar a tração do cabo e os ângulos que determinam a posição instantânea do aerofólio. A direção e velocidade do vento são mostradas no *faceplate*¹ do anemômetro. Na tela da Fig. 11 é possível visualizar uma máquina de estados, também representada em *faceplate* próprio, que define a situação atual do controle de voo por meio de quatro estados.

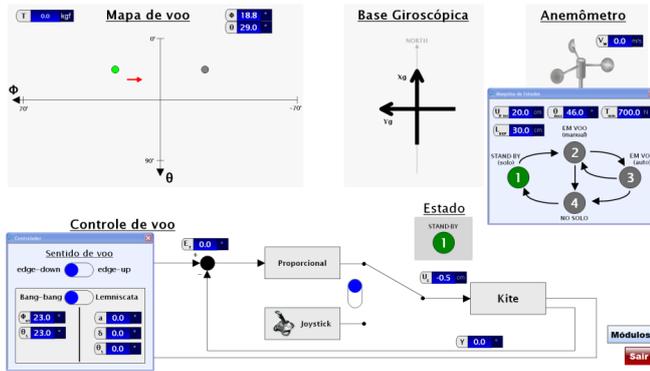


Figura 11. Exemplo de tela do sistema SCADA. Fonte: original.

- **Stand-by:** o aerofólio está no solo pronto para lançamento e todos os módulos estão devidamente inicializados e parametrizados.
- **Voo manual:** o aerofólio já decolou e seu voo é controlado pelo operador por meio de um *joystick*.
- **Voo automático:** o voo é guiado pelas malhas de controle embarcadas no módulo *flight-controller*.
- **No solo:** a tração do cabo e ângulo de elevação (complementar do ângulo polar) caíram abaixo de certos valores-limite, caracterizando a queda ou pouso proposital do aerofólio. Neste estado, as referências de posição dos servomotores são mantidas constantes para evitar desgaste mecânico desnecessário.

A comunicação do SCADA com o *software* embarcado utiliza o protocolo OPC (*OLE for Process Control*), que permite a interconectividade entre dispositivos de fabricantes diferentes. Na estrutura de *software*, o módulo *OPC-Gateway*, utilizando o padrão *publish/subscribe*, adquire e disponibiliza ao servidor OPC os dados do SCADA. Desta forma é possível atualizar e visualizar o valor das variáveis em tempo real, apesar de uma pequena latência na comunicação, de aproximadamente um segundo.

6. HARDWARE-IN-THE-LOOP

Testes do tipo HIL consistem em operar alguns elementos de *hardware* de um determinado sistema enquanto o comportamento de outros elementos é simulado em computador. Geralmente a parte simulada apresenta dificuldades ou perigos em se operar na prática, ou o seu *hardware* ainda não está funcional. No caso do UFSCkite, testes de campo apresentam uma série de inconveniências, como

¹ *Faceplate* é uma pequena tela, de caráter específico a algum equipamento/subsistema, que aparece ao se clicar em algum elemento gráfico (tela *pop-up*) de uma tela principal do SCADA.

a distância significativa entre o laboratório e o local de testes, a necessidade de mobilizar várias pessoas para a desmontagem, transporte e montagem do protótipo, além da necessidade de condições climáticas adequadas. Assim, para acelerar o processo de desenvolvimento do aerogerador, os protótipos são testados em laboratório, em que a dinâmica de voo (parte do sistema não operável em laboratório) é simulada computacionalmente. No estado atual do protótipo da UCV, as variáveis informadas pelo simulador são os ângulos e velocidades angulares que definem a posição e velocidade do aerofólio, o comprimento de cabo desenrolado e a força de tração. Já a Raspberry Pi fornece os valores de referência de *steering* e *depower* e a velocidade de desenrolamento do cabo.

6.1 FreeKiteSim

O *FreeKiteSim* é uma ferramenta *open-source* que usa o modelo de Fechner et al. (2015) para simular o comportamento dinâmico do voo de um aerofólio cabeado (Fechner, 2015). Desenvolvido em Python, o simulador permite o controle do aerofólio tanto pelos botões do teclado de um PC, por um *joystick* conectado ao PC ou ainda por meio de algum outro dispositivo externo. O código original do *FreeKiteSim* foi alterado para incluir uma interface de comunicação de rede UDP que conecta o simulador ao *software* embarcado na unidade de voo. Desta forma, as informações de posição e velocidade do aerofólio são enviadas do simulador para o *software* embarcado na Raspberry Pi, que as processa como se fossem provenientes dos sensores reais do aerogerador, calcula o sinal de controle e transmite o comando de *steering* ao simulador, que o aplica no modelo dinâmico de voo.

6.2 Resultados

Os resultados apresentados a seguir foram obtidos usando a estratégia de pontos atratores para a malha externa do controle de voo, além das configurações detalhadas na Tab. 3. Na Fig. 12 é mostrada a trajetória de voo sobre o plano (θ, ϕ) . Nota-se que a trajetória é deformada com relação à lemniscata de Bernoulli, uma vez que o ponto de referência permanece constante no atrator ativo em vez de ser atualizado sobre a lemniscata a cada período de amostragem. É importante ter em mente que o ângulo θ é zero quando o aerofólio está no zênite, e 90° quando o aerofólio está no chão. Na Fig. 13 é mostrado o sinal de controle (*steering*): percebe-se um pico repentino no momento da comutação entre atratores. Já a potência mecânica extraída do vento pelo desenrolamento do cabo é apresentada na Fig. 14. A potência gerada é de fato baixa para as condições de operação, dado que a velocidade de desenrolamento ótima é $v_w^* = (1/3) \sin \theta \cos \phi$ (de Oliveira, 2016). Considerando valores médios das variáveis envolvidas, tem-se $v_w^* \approx 2.5$ m/s, em vez dos 0.5 m/s utilizados na simulação.

7. CONSIDERAÇÕES FINAIS

Neste artigo foi apresentado o projeto de uma unidade de controle de voo de um aerogerador baseado em aerofólio cabeado do tipo *pumping kite*. Os principais sensores e atuadores do protótipo desenvolvido pelo grupo UFSCkite

Parâmetro	Valor
Ângulo azimutal inicial (ϕ_0)	0°
Ângulo do atrator (α)	35.4°
Ângulo polar central (θ_c)	45°
Ângulo polar inicial (θ_0)	30°
Área do aerofólio	10.18 m ²
Comprimento de cabo desenrolado inicial	150 m
Ganho proporcional do controlador (K_p)	0.2 rad ⁻¹
Massa da unidade de controle de voo (UCV)	8.4 kg
Massa do aerofólio	6.24 kg
Período de amostragem da simulação	0.05 s
Período de amostragem do controle de voo	0.03 s
Velocidade de desenrolamento do cabo	0.5 m/s
Velocidade do vento em 6 m de altura	9.39 m/s

Tabela 3. Parâmetros de configuração do teste HIL. Fonte: original.

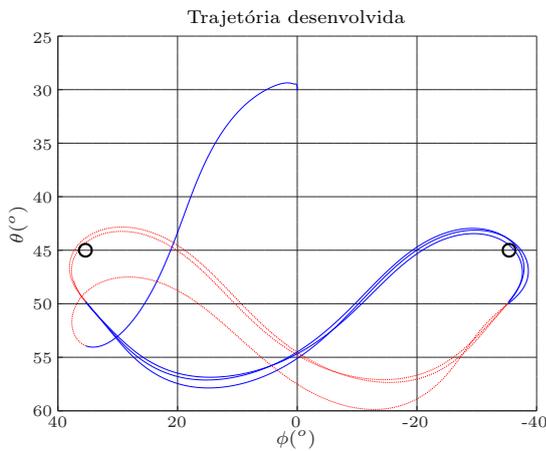


Figura 12. Trajetória de voo do aerofólio. Fonte: original.

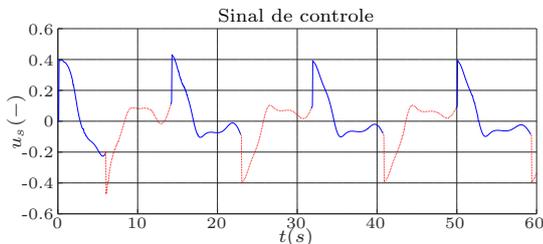


Figura 13. Sinal de controle (*steering*). Fonte: original.

foram elencados, e foi apresentada uma arquitetura distribuída de módulos de *software*. Os módulos comunicam-se entre si via pacotes LCM se localizados na mesma plataforma computacional, e entre mensagens de RF se localizados em plataformas distintas do aerogerador. Estas mensagens são trocadas entre *beacons* que compõem um sistema de posicionamento do aerofólio por lateração. Algoritmos de controle de voo também foram discutidos, os quais foram validados com a realização de ensaios do tipo HIL utilizando um simulador *open-source* de *pumping kites*. Na continuação deste projeto estão previstos ensaios de campo da unidade de voo, seguidos por testes HIL da unidade de solo. Somente então as duas unidades serão testadas em campo conjuntamente na operação do *pumping kite* completo, inicialmente em modo manual (pilotado pelo operador) e, mais adiante, com os controles automáticos de voo e do desenrolamento de cabo.

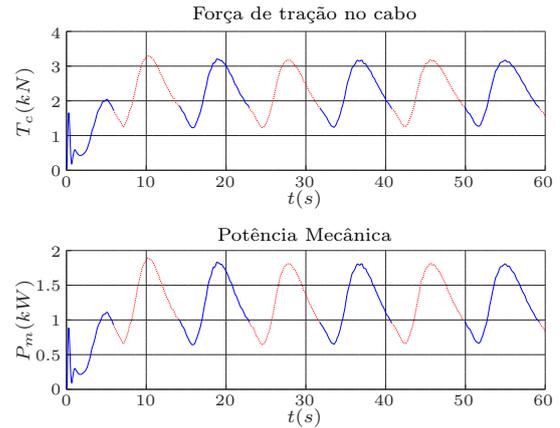


Figura 14. Potência mecânica extraída do vento pelo desenrolamento do cabo. Fonte: original.

AGRADECIMENTOS

Este trabalho foi apoiado pela CAPES e CNPq na forma de bolsas de estudo, além de recursos via Chamada MC-TIC/CNPq n° 28/2018, processo 433113/2018-9.

REFERÊNCIAS

- Araujo, H. (2017). Controle da máquina elétrica de um aerogerador com aerofólio cabeado na configuração pumping-kite. Diss. de Mest., PPGEAS/UFSC.
- da Silva, R.S. (2014). Aerofólios cabeados para geração de energia elétrica. Diss. de Mest., PPGEAS/UFSC.
- de Oliveira, M.D.L.C. (2016). Airborne wind energy with tethered wings: modeling, analysis and control. Tese de dout., PPGEAS/UFSC.
- Diehl, M. (2013). Airborne wind energy: Basic concepts and physical foundations. In *Airborne Wind Energy*, 3–22. Springer Berlin Heidelberg. doi:10.1007/978-3-642-39965-7_1. URL 1.
- Fechner, U., van der Vlugt, R., Schreuder, E., and Schmehl, R. (2015). Dynamic model of a pumping kite power system. *Renewable Energy*, 83, 705–716. doi: 10.1016/j.renene.2015.04.028.
- Fechner, U. (2015). Freekitesim - open source kite power system simulator in python. URL <https://bitbucket.org/ufechner/freekitesim/src/master/>.
- International Renewable Energy Agency (2019). *Future of wind: Deployment, investment, technology, grid integration and socio-economic aspects*.
- Makani Power (2020). A long and windy road. URL <https://medium.com/makani-blog/a-long-and-windy-road-3d83b9b78328>.
- Saraiva, R., Lellis, M.D., Bastos, M.B., and Trofino, A. (2019). An algebraic solution for tracking bernoulli's lemniscate flight trajectory in airborne wind energy systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE. doi:10.1109/cdc40024.2019.9029406.
- Winter, M.K. (2017). Desenvolvimento de uma plataforma de software embarcado para aerogeradores com aerofólios cabeados. Mon. de grad., UFSC.