



**UNIVERSIDADE SENAI CIMATEC**  
**GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

## **OTIMIZAÇÃO DE THRUSTERS PARA TOLERÂNCIA A FALHAS EM VEÍCULOS REMOTAMENTE OPERADOS (ROVS) EM 4 DOFS**

Júlia Maria Nascimento Ribeiro<sup>1</sup>

Orientador: Msc. Rebeca Tourinho Lima<sup>2</sup>

Coorientador: Msc. Lucas Marins Batista<sup>3</sup>

<sup>1</sup>Universidade Senai Cimatec, E-mail: julia.ribeiro@gmail.com;

<sup>2</sup>Universidade Senai Cimatec, E-mail: rebeca.lima@fieb.org.br;

<sup>3</sup>Universidade Senai Cimatec, E-mail: lucas.batista@fieb.org.br;

## **THRUSTER OPTIMIZATION FOR FAULT TOLERANCE IN REMOTELY OPERATED VEHICLES (ROVS) IN 4 DOFS**

**Resumo:** *A fazer* **Palavras-Chaves:** xxxxx

**Abstract:** *TODO*

**Keywords:** *xxxxx*

## 1 INTRODUÇÃO

Segundo Biazon (2017), os oceanos representam cerca de 70% da superfície terrestre na atualidade, mas apesar da sua vasta ocupação, somente 5% do território oceânico é conhecido pelo homem. A lacuna de conhecimento gera uma crescente demanda na indústria por recursos encontrados *offshore*, como petróleo, recursos minerais (óleos, gás e metais) e biodiversidades.

A indústria apresenta uma crescente expansão em relação ao oceano. No relatório anual de sustentabilidade (PETROBRAS, 2024) é relatado que 98% da produção do ano de 2024 veio de águas profundas ou ultraprofundas. Esse cenário de exploração crescente cria uma demanda cada vez maior por tecnologias e mão de obra especializada.

Mergulhadores são uma mão de obra especializada amplamente utilizada na indústria para trabalhos *offshore*, seja para manutenção ou instalação de equipamentos. Segundo (ADMINISTRAÇÃO DE SEGURANÇA E SAÚDE OCUPACIONAL, 2023), a organização de trabalho americana, os perigos enfrentados por profissionais do mergulho incluem afogamento, riscos respiratórios e cardiovasculares, hipotermia e riscos físicos devido ao manuseio de equipamentos pesados embaixo d'água, além dos riscos relacionados à descompressão e possíveis acidentes causados pelo ambiente de alto risco e nível elevado de estresse.

Dessa maneira, a indústria está constantemente em busca de melhorar a segurança no trabalho. Dentro desse contexto surgem os ROVs (*Remotely Operated Vehicles*), sendo esses uma alternativa estratégica para retirar o trabalhador de uma zona iminente de perigo.

ROVs são veículos subaquáticos que variam de tamanho, podendo ser de maior ou menor porte, e possuem a capacidade de se autopropelir abaixo da superfície da água, podendo, em alguns casos, também operar na superfície, dependendo de seu projeto e aplicação. Por serem veículos remotamente operados, podem ser utilizados em aplicações em alta profundidade de forma mais segura por não precisarem da presença de um ser humano na área de perigo.

Ao tirar o ser humano da zona de perigo são reduzidos os riscos fatais, além de oferecer vantagens operacionais como a possibilidade de um trabalho submerso contínuo e prolongado.

Apesar da utilização de ROVs ser vantajosa e funcional, a operação desses veículos é um desafio devido as perturbações geradas pelos oceanos, que contam com correntes marítimas imprevisíveis, baixa visibilidade, entre outras adversidades.

Assim, para lidar com as adversidades enfrentadas no ambiente subaquático, os ROVs necessitam de sistemas de controle capazes de compensar os ruídos gerados pelas perturbações ambientais, garantindo estabilidade, navegação autônoma e execução adequada das atividades para as quais foram projetados. Dessa forma, diversas técnicas de controle foram desenvolvidas, variando desde a modelagem do sistema até a utilização de inteligência artificial.

Em (XU; WANG, 2018) são citados alguns tipos de controladores, entre eles o controlador PID, caracterizado por ser um controlador simples e amplamente utilizado, é um controle de resposta de velocidade, com efeito linear. Apesar de ser amplamente utilizado, não é o ideal para processos complexos e não lineares ou variantes no tempo.

As redes neurais também são apontadas como um possível controlador para um ROV, sendo essas caracterizadas por serem capazes de aproximar qualquer função não linear na teoria. Apresentam uma alta capacidade de processamento paralelo, o que gera uma grande tolerância a falhas e eficiência no tratamento de múltiplos sinais de entrada e saída.

Apesar das inúmeras vantagens, é necessário ter cautela, pois há a tendência de convergir para mínimos locais, além de sofrer com a lentidão na aprendizagem, limitação no número de camadas e riscos de overfitting. A limitação de camadas acontece quando a rede neural tem poucas camadas, o que impede que a mesma aprenda padrões complexos. Já o overfitting ocorre quando uma rede neural é treinada de forma excessiva ou por tempo prolongado, causando que a mesma decore os dados do teste, indo muito bem nos testes, mas apresentando resultados ruins com novos dados.

Por último é mencionado o controlador *fuzzy*, caracterizado por sua independência de uma modelagem matemática exata do objeto de controle, tendo sua essência em estratégias de autocontrole.

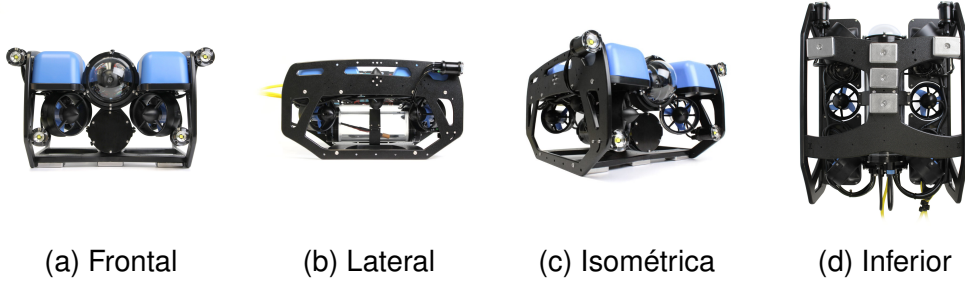
Segundo Christ e Wernli Sr (2013), o sistema propulsor impacta diretamente no design de um ROV, impactando assim nos tipos de thrusters escolhidos e na configuração dos mesmos. Thrusters são dispositivos responsáveis por impulsionar a embarcação, ou seja, são os propulsores que garantem sua movimentação.

Um dos desafios enfrentados por veículos que utilizam thrusters é a possibilidade de falha em um ou mais desses dispositivos durante a operação, o que levanta a questão de como recuperar o controle do veículo ou garantir que ele continue navegando mesmo diante dessas falhas.

Dessa forma, o objetivo principal deste trabalho é abordar o problema da falha de thrusters por meio da aplicação de um algoritmo de otimização, para gerar tolerância a falhas, através da otimização da redistribuição das forças restantes pós-falha, buscando manter a estabilidade e a funcionalidade do sistema mesmo em condições adversas.

Os estudos relacionados ao algoritmo de otimização serão aplicados ao robô BlueROV2, um veículo subaquático do tipo mini ROV desenvolvido pela empresa Blue Robotics. A Figura 1 apresenta uma representação visual do robô.

Figura 1: Vistas do robô BlueROV.



Fonte: Blue Robotics.

A modelagem do algoritmo otimizador será desenvolvida levando em consideração as dimensões, características e requisitos específicos do BlueROV2.

Todos os testes e simulações serão realizados em ambiente computacional, utilizando ferramentas como o Gazebo Sim, para simular a movimentação do ROV pré e pós-falha, para a observabilidade da realocação das forças otimizadas.

## 2 FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica fundamental para modelagem de ROV é descrita em Fossen (Fossen, 2021), onde é demonstrado o modelo matemático para uma embarcação marítima com 6 DOFs. As equações de movimento adotadas do modelo dinâmico apresentado por Fossen contam com a equação de cinemática, equação 1, e a equação de cinética, equação 2.

$$\dot{\eta} = J(\eta)v \quad (1)$$

$$M\dot{v} + C(v)v + D(v)v + g(\eta) = \tau \quad (2)$$

Onde:

- $J(\eta)$  – É a matriz de transformação entre Body e NED;
- $M$  – É a matriz de massa total;
- $C$  – É a matriz de Coriolis;
- $D$  – É a matriz de amortecimento hidrostático;
- $g$  – É o vetor geral de restauração de força;

As transformações em *Body* e *NED* (Norte, Leste, Descer) são essenciais para a modelagem de veículos subaquáticos, pois permitem a conversão entre diferentes sistemas de coordenadas utilizados na navegação e controle do ROV. Essas transformações são cruciais para garantir que os cálculos de movimento e controle sejam precisos e consistentes, independentemente do referencial adotado. Sendo *Body* o referencial atrelado ao veículo, e *NED* o referencial fixo ao solo. A equação 1 representa a cinemática do sistema, descrevendo assim os aspectos geométricos da movimentação do ROV em relação a diferentes coordenadas, enquanto a equação 2 representa

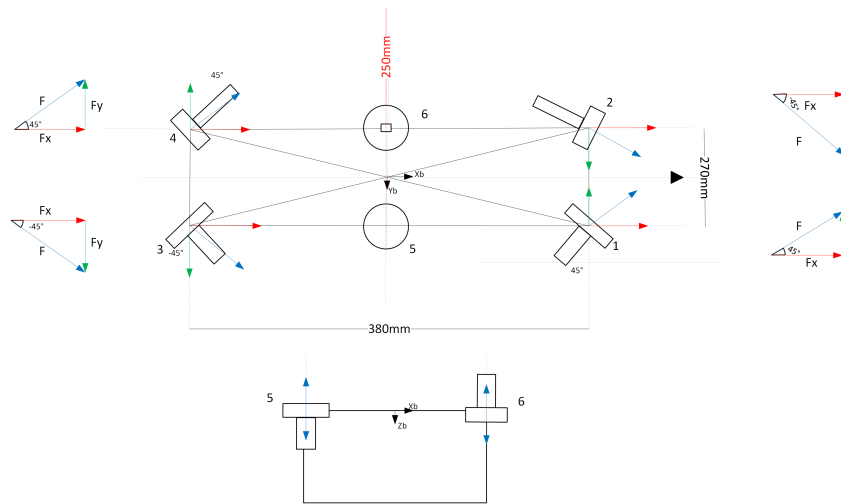
a cinética do modelo, que consiste na análise de forças e momentos incluídas no ROV durante o movimento.

Baseando-se nas equações de cinemática e cinética para 6 DOFs, é possível modelar o sistema do BlueROV light, que só é capaz de se movimentar em 4 DOFs. Dessa forma, os graus de liberdade que não são acessados pelo robô são zerados nas matrizes.

## 2.1 Modelo e Alocação de Thruster

O modelo de ROV utilizado (BlueROV2 standard) conta com 6 thrusters, como é possível visualizar na Figura 2, sendo esses 4 horizontais e 2 verticais, tendo sua configuração ilustrada também na Figura 2. Dessa forma, os thrusters dianteiros (1 e 2) giram no sentido anti-horário e os traseiros (3 e 4) giram no sentido horário.

Figura 2: Diagrama de thruster e forças.



Fonte: Autores.

Apesar da configuração física atípica dos thrusters, posicionados em direções opostas, a alocação de forças foi corretamente realizada graças à rotação em sentidos contrários. Caso todos os thrusters rotacionassem no mesmo sentido, essa disposição geraria forças opostas, comprometendo o controle do sistema. No entanto, como evidenciado na Figura 2, a rotação alternada dos thrusters soluciona esse problema, permitindo a geração das forças desejadas.

Conforme apresentado por (WU, 2018), o modelo de um thruster pode ser apresentado de forma linear pela Equação 1. Onde a força do thruster é representada pelo vetor 4, os inputs de controle são representados por 5, e os coeficientes de thruster são representados pela matriz diagonal 6.

$$F = Ku \quad (3)$$

$$F = \begin{bmatrix} F_1 & F_2 & \dots & F_n \end{bmatrix}^T \quad (4)$$

$$u = \begin{bmatrix} u_1 & u_2 & \dots & u_n \end{bmatrix}^T \quad (5)$$

$$K = [K_1 \ K_2 \ \dots \ K_n] \quad (6)$$

Considerando o vetor de força  $f$  em 7 e o vetor de momentos  $r$  em 8, podemos calcular as forças e momentos em 6 DOFs pela seguinte fórmula:

$$f = [f_x \ f_y \ f_z]^T \quad (7)$$

$$r = [l_x \ l_y \ l_z]^T \quad (8)$$

$$\tau = \begin{bmatrix} f \\ r \times f \end{bmatrix} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ F_z l_y - F_y l_z \\ F_x l_z - F_z l_x \\ F_y l_x - F_x l_y \end{bmatrix} \quad (9)$$

Considerando a limitação do BlueROV2 standard para somente 4 DOFs, é possível ajustar as Equações 8 e 9 para obter as forças em 4 DOFs como desejado. Sendo necessário zerar os momentos que não serão aplicados ao ROV, tendo como resultado então:

$$r = [0 \ 0 \ l_z]^T \quad (10)$$

$$\tau = \begin{bmatrix} f \\ r \times f \end{bmatrix} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ F_y l_x - F_x l_y \end{bmatrix} \quad (11)$$

A alocação de controle é então modelada por:

$$\tau = T(\alpha)F = T(\alpha)Ku \quad (12)$$

Sendo  $T$  a matriz de alocação, onde  $T \in \mathbb{R}^{4 \times 6}$  e  $\alpha$  o vetor de ângulos de rotação dos thrusters, onde  $\alpha \in \mathbb{R}^6$ . Como consequência, a matriz de configuração de thruster  $T$  pode ser calculada usando a Equação 11. Uma vez determinadas as forças e momentos dos thrusters, é possível formalizar o problema da alocação, cujo objetivo é distribuir corretamente os esforços para os propulsores, para realizar a ação de controle desejada.

## 2.2 Alocação de controle

(WU, 2018) define a alocação de controle como o processo que computa o sinal de entrada do controle  $u$  e o aplica nos thrusters, de forma que o controle geral de forças  $\tau$  possa ser gerado. Partindo da Equação 11, é possível calcular o vetor de entradas de controle através da seguinte equação:

$$u = K^{-1}T^{-1}\tau \quad (13)$$

Contudo, levando em conta que a matriz  $T$  é uma matriz não quadrática, é aplicada a Moore-Penrose pseudo-inversa  $T^+$ , dada por:

$$T^+ = T^T(TT^T)^{-1} \quad (14)$$

Assim, podemos obter o vetor de entradas do controle por:

$$u = K^{-1}T^+\tau \quad (15)$$

## 2.3 Formulação da Otimização

Em (JOHANSEN; FOSSEN; TONDEL, 2005) é sugerida uma formulação de otimização para o problema de alocação de controle. É considerado então o seguinte problema de otimização:

$$\min_{u,s,\tilde{u}}(s^T Q s + u^T W u + \beta \tilde{u}) \quad (16)$$

A Equação 16 está sujeita ao seguinte:

$$T u = \tau + s \quad (17)$$

$$u_{\min} \leq u \leq u_{\max} \quad (18)$$

$$-\tilde{u} \leq u_1, u_2, \dots, u_N \leq \tilde{u} \quad (19)$$

A variável  $s$  é o termo que garante a restrição proposta em 18, que faz com que o resultado da força generalizada  $Bu$  desvie das especificações de  $\tau$ , caso seja necessário. O segundo termo do critério corresponde ao critério de mínimos quadrados, enquanto o terceiro termo minimiza a maior força entre os atuadores, devido a 19. O parâmetro  $\beta \geq 0$  controla a ponderação relativa desses dois critérios, permitindo que sejam tratados os compromissos entre o uso médio e o pior caso do controle.

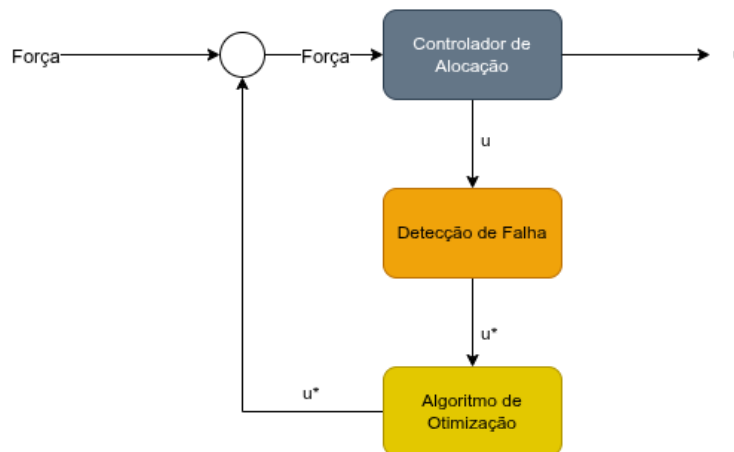
## 3 METODOLOGIA

A metodologia proposta neste trabalho consiste no desenvolvimento e validação de um algoritmo de otimização voltado à redistribuição de forças em veículos subaquáticos (*BlueROV2*), com o objetivo de manter a navegabilidade do sistema em caso de falha total de um ou mais *thrusters*.

Os experimentos foram conduzidos em ambiente de simulação, utilizando o sistema operacional Ubuntu 22.04, o simulador Gazebo Ignition e o framework ROS 2 Control. O modelo *BlueROV2 Standard* foi adotado como base para representar o comportamento dinâmico do veículo e verificar o desempenho do método proposto. O controle do ROV é feito por um controlador de alocação de esforços, responsável por distribuir as forças entre os *thrusters*. O algoritmo de otimização será integrado em paralelo com o controle de alocação, de modo a realizar a redistribuição adaptativa das forças quando houver falha em um ou mais propulsores. Vale destacar que o reconhecimento de falha não é abordado neste trabalho, por ser um desafio à parte da otimização das forças.

O diagrama do fluxo de controle proposto é apresentado na Figura 3, destacando a interação entre o controlador de alocação e o algoritmo de otimização, bem como as entradas e saídas do sistema. O desempenho do algoritmo será avaliado com base no erro de Pearson e o coeficiente de determinação  $r^2$  e no desvio padrão do erro de cada repetição.

Figura 3: Fluxo de controle.



Fonte: Autores.

A partir dessas análises é esperado verificar a robustez e a eficiência do método de otimização proposto em comparação ao controle convencional sem a tolerância a falhas.

### 3.1 Metodologia experimental

A validação do algoritmo de otimização será realizada por meio do seguinte procedimento experimental:

Foram estabelecidos três cenários de teste, variando conforme a presença ou ausência do algoritmo de otimização e a ocorrência de falhas nos *thrusters*. Os cenários foram organizados da seguinte forma:

- Cenário 1: Trajetória sem falhas e sem o algoritmo de otimização;
- Cenário 2: Trajetória com falhas e sem o algoritmo de otimização;
- Cenário 3: Trajetória com falhas e com o algoritmo de otimização.

Cada cenário serão aplicados em cinco casos de falhas, sendo esses:

- Caso 1: Falha em dois *thrusters* horizontais paralelos;
- Caso 2: Falha em dois *thrusters* horizontais coincidentes;
- Caso 3: Falha em dois *thrusters* horizontais diagonais;
- Caso 4: Falha em um *thruster* horizontal e um vertical;
- Caso 5: Falha em um *thruster* horizontal.

Os testes foram realizados em três intervalos de tempo (30, 60 e 90 segundos) para cada tipo de trajetória, a fim de observar o comportamento do sistema ao longo do tempo. Cada combinação de cenário, caso e tempo foi repetida cinco vezes, totalizando 225 execuções experimentais. Foram delimitadas 2 trajetórias para o ROV seguir, sendo elas:



- Trajetória 1: Movimento circular;
- Trajetória 2: Movimento quadrático.

As trajetórias são realizadas a partir de uma força de *wrench* aplicada nos *thrusters* do ROV, o que gera um movimento relacionado ao tipo de trajétorio, ou seja, a trajetória circular não se trata de um movimento perfeito, mas sim de uma força aplicada que gere um movimento aproximado ao circular, o mesmo é aplicado na trajetória quadrática.

Após as 225 execuções, os dados coletados serão analisados estatisticamente, utilizando métricas como o erro de Pearson e  $r^2$ , para a correlação e similaridade entre os casos com falhas e o caso original de pleno funcionamento, e o desvio padrão do erro para cada repetição. Essas análises permitirão avaliar a eficácia do algoritmo de otimização na manutenção da navegabilidade do ROV em diferentes cenários de falhas.

## 4 RESULTADOS E DISCUSSÃO

## 5 CONSIDERAÇÕES FINAIS

### AGRADECIMENTOS

### REFERÊNCIAS

- BIAZON, Tássia. **Nas profundezas dos oceanos**. *Revista Marítima Brasileira*, v. 137, n. 7-9, p. 66–82, 2017.
- CHRIST, Robert D.; WERNLI SR, Robert L. **The ROV manual: a user guide for remotely operated vehicles**. Oxford, Reino Unido: Butterworth-Heinemann, 2013.
- JOHANSEN, Tor A.; FOSSEN, Thor I.; TONDEL, Petter. **Efficient optimal constrained control allocation via multiparametric programming**. *Journal of Guidance, Control, and Dynamics*, v. 28, n. 3, p. 506–515, 2005.
- OCCUPATIONAL SAFETY AND HEALTH ADMINISTRATION. **Commercial Diving: Hazards and Solutions**. Washington, D.C., EUA: U.S. Department of Labor, 2023.
- PETROBRAS. **Relatório de sustentabilidade Petrobras 2024**. Rio de Janeiro, Brasil: Petrobras, 2024.
- WU, Chu-Jou. **6-DOF modelling and control of a remotely operated vehicle**. 2018. – Flinders University, College of Science e Engineering, Adelaide, Australia.
- XU, Jiankang; WANG, Nan. **Optimization of ROV control based on genetic algorithm**. In: OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO). Kobe, Japão: IEEEExplore, 2018. p. 1–4.