

O presente trabalho abordou o desenvolvimento e a validação de um algoritmo de otimização para tolerância à falha em veículos subaquáticos, mais especificamente no *BlueROV2 Standard*, que possui 4 DOFs e 6 *thrusters*. O objetivo principal foi conservar a navegabilidade do veículo em casos de falha total de um ou mais *thrusters*. A metodologia proposta envolveu a implementação do algoritmo em um ambiente de simulação utilizando, com os testes foi possível constatar a eficiência do algoritmo proposto, o mesmo sofre com limitações causadas pela configuração de *thrusters* do veículo. Os resultados obtidos demonstram que o algoritmo de otimização proposto é eficaz na redistribuição das forças entre os *thrusters* remanescentes, permitindo que o ROV mantenha sua navegabilidade mesmo em situações de falha, desde que as condições mínimas de navegabilidade sejam mantidas.

O otimizador demonstra sua eficiência principalmente no âmbito da velocidade média e distância percorrida, mesmo em casos onde a combinação mínima de *thrusters* não é atendida. Como o *BlueROV2 Standard* possui uma configuração de 6 *thrusters*, o otimizador encontra limitações na redistribuição das forças quando múltiplos *thrusters* falham, o que causa uma dificuldade em manter a trajetória ideal, sendo possível visualizar erros significativos em *Yaw* nos casos que não atendem às combinações mínimas para aquela direção. Nos casos que atendem às combinações mínimas, o otimizador é capaz de diminuir o erro médio em *yaw*, que é causado pelas falhas dos *thrusters*, mantendo assim a capacidade do ROV de seguir a trajetória desejada.

O caso 3 se destaca por apresentar erros muito próximos entre os cenários com e sem otimização em qualquer trajetória, indicando que a falha diagonal representada pelo caso três tem a capacidade de se autocompensar devido ao posicionamento dos *thrusters* no ROV, o que minimiza o impacto da falha na capacidade de movimentação do veículo. O maior fator de melhoria observado para o algoritmo de otimização é observado tanto na distância percorrida quanto na velocidade média, mantendo a capacidade do ROV de manter a velocidade o mais próxima do ideal possível e preservando também a capacidade de percorrer distâncias similares ao caso ideal, mesmo com a falha de *thrusters*, para todos os casos na maioria das trajetórias, enfrentando limitações somente na trajetória em *yaw*.

Apesar das dificuldades enfrentadas pelo otimizador em manter a trajetória ideal em todos os casos, os resultados obtidos indicam que o algoritmo proposto é eficaz na redistribuição das forças entre os *thrusters* remanescentes, permitindo que o ROV mantenha sua navegabilidade mesmo em situações de falha, desde que as condições mínimas de navegabilidade sejam mantidas.

Sendo assim, o trabalho contribui para o avanço da segurança operacional de ROVs em aplicações industriais submarinas, oferecendo uma solução viável para mitigar os efeitos de falhas em *thrusters* e garantindo a continuidade das operações em ambientes desafiadores. Para trabalhos futuros, visa-se a implementação do algoritmo no ROV em sua configuração *heavy*, que conta com 8 *thrusters*, o que pode proporcionar uma maior flexibilidade na redistribuição das forças e potencialmente melhorar o desempenho do algoritmo em cenários de falha múltipla, permitindo também que a otimização sem *heave*, seja realizada. Também é possível realizar a implementação do algoritmo no *Bluerov2* real, para validação dos resultados obtidos em simulação, e observação dos efeitos do algoritmo no mundo real, sendo possível combinar a existência do algoritmo de otimização com controladores que permitirão ao ROV compensar os desvios de trajetória causados pela não correspondência dos *thrusters* mínimos de operação.