Algoritmo híbrido para o roteamento de VANTs colaborativos com desvio de obstáculos ⋆

Antônio L. S. Aguiar * **Vandilberto P. Pinto** **

* *Universidade Federal do Ceará, Campus de Sobral R. Cel. Estanislau Frota, 563 - Centro, Sobral - CE, 62010-560 (e-mail: lucas.aguiar@alu.ufc.br).*
** *Universidade da Integração Internacional da Lusofonia Afro-Brasileira, Campus das Auroras R. José Franco de Oliveira, s/n - Zona Rural, Redenção - CE, 62790-970 (e-mail: vandilberto@unilab.edu.br)*

Abstract: This study addresses efficient task assignment for collaborative systems with a focus on route planning for multiple Unmanned Aerial Vehicles (UAVs). A hybrid algorithm based on the Ant Colony System is used with the application of the A* and DWA algorithms for the avoidance of fixed and mobile obstacles. The results show that the proposed method enables efficient route planning with the potential for more complex scenarios, providing guidance on mission time and distance traveled.

Resumo: Este estudo aborda a atribuição eficiente de tarefas para sistemas colaborativos com foco no planejamento de rotas para múltiplos Veículos Aéreos Não Tripulados (VANTs). É utilizado um algoritmo híbrido baseado no Sistema de Colônia de Formigas com a aplicação dos algoritmos A* com a Abordagem de Janela Dinâmica (do inglês *Dynamic Window Approach*, DWA) para o desvio de obstáculos fixos e móveis. Os resultados mostram que o método proposto permite o planejamento de rotas de forma eficiente com potencial para cenários mais complexos, fornecendo orientações sobre o tempo da missão e a distância percorrida.

Keywords: Multiple Traveling Salesmen; UAVs; Route planning; Ant colony System; DWA; Circumvent Obstacles.

Palavras-chaves: Múltiplos caixeiros viajantes; VANTs; Planejamento de rotas; Sistema de colônia de formigas; DWA; Desvio de obstáculos.

1. INTRODUÇÃO

O setor de transporte de cargas tem mostrado um interesse crescente e feito investimentos em aplicações de Veículos Aéreos Não Tripulados (VANTs). Este interesse tem sido impulsionado pelo aumento do comércio eletrônico, que tem gerado demanda por soluções inovadoras de logística. Os VANTs têm a capacidade de decolar e pousar com segurança em áreas próximas a edifícios e pessoas, o que melhora significativamente a qualidade dos serviços de transporte em regiões congestionadas ou de difícil acesso. Essa tecnologia tem o potencial de revolucionar a forma como as mercadorias são transportadas, oferecendo eficiência e agilidade. A utilização de VANTs no transporte de cargas pode representar uma mudança significativa no setor, proporcionando benefícios tanto para as empresas quanto para os consumidores (Floreano and Wood, 2015).

De acordo com os benefícios mencionados, a literatura ressalta a relevância da coordenação eficaz, especialmente em missões complexas que envolvem a otimização de trajetórias para múltiplos robôs colaborativos, conforme evidenciado em (Ju and Son, 2018; Liu et al., 2019). A alocação de tarefas nesse contexto requer não apenas a consideração das restrições energéticas, de capacidade e desvio de obstáculos, mas também a integração efetiva da navegação dos VANTs com seu sensoriamento para evitar colisões e alcançar um planejamento de rotas eficiente.

Desta forma, Wang et al. (2022) propôs uma abordagem baseada no Sistema de Colônias de Formigas Multiobjetivo para resolver o problema de atribuição de tarefas cooperativas com múltiplos robôs. Já Santana (2020), desenvolveu um algoritmo baseado em Otimização por Enxame de Partículas e Algoritmos Genéticos (AG) para resolver o problema de cálculo de rotas para robôs colaborativos com restrições energéticas. O modelo utilizado é uma variante do problema de orientação de equipes, que inclui características de coleta de recompensa ao passar pelos pontos de referência. Da mesma forma, Freitas and Carvalho (2015) utilizaram AG para resolver o problema de roteamento de múltiplos VANTs, obtendo resultados satisfatórios e tempos de execução razoáveis ao aplicar o modelo na solução de um problema envolvendo uma frota de VANTs.

Além disso, Pinto et al. (2020) propuseram um modelo de otimização usando uma abordagem de Programação Linear Inteira (PLI) para resolver um problema de pla-

⋆ Os autores agradecem à Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP) pelo apoio financeiro (processo BP5-0197-00058.01.00/22).

nejamento de missões múltiplas, considerando a influência do campo de vento e restrições de duração da missão. Já Aguiar et al. (2023), utiliza em seu trabalho a abordagem PLI com o algoritmo A* para o planejamento de rotas com múltiplos robôs colaborativos com aplicação de desvio de obstáculos. E Liu et al. (2021), propõe o uso da fusão do algoritmo A* com a Abordagem de Janela Dinâmica (do inglês *Dynamic Window Approach*, DWA) para o planejamento de rotas com desvio de obstáculos.

Com isso, este estudo visa aplicar um algoritmo híbrido para o roteamento de VANTs colaborativos com desvio de obstáculos, sendo utilizado um modelo matemático com foco na Otimização por Colônia de Formigas (do inglês *Ant Colony Optimization*, ACO). O ACO, inspirado no comportamento cooperativo das formigas, é uma técnica de otimização amplamente utilizada em problemas de otimização combinatória. Nesse contexto, levando-se em consideração as restrições dos Múltiplos Caixeiros Viajantes (MCV) e a necessidade de desviar de obstáculos, os algoritmos A* e o DWA também foram aplicados para o desvio de obstáculos fixos e móveis, respectivamente.

As demais seções deste artigo estão organizadas como segue. A Seção 2 aborda o modelo matemático do problema. A Seção 3 apresenta o ACO multiobjetivo, bem como os algoritmos A* e DWA utilizados para evitar colisões entre os VANTs e os obstáculos. A Seção 4 detalha a metodologia empregada nos experimentos computacionais. A Seção 5 analisa os resultados obtidos nas simulações, enquanto a Seção 6 traz as principais conclusões e perspectivas para trabalhos futuros.

2. DESCRIÇÃO DO PROBLEMA

Dado um grupo de N_v VANTs $\mathcal{V} = \{1, 2, ..., N_v\}$, e um conjunto de N_t pontos de referência que podem ser explorados, representados por $\mathcal{T} = \{1, 2, ..., N_t\}$, onde 1 e N_t sinalizam o ponto de partida e de chegada, respectivamente, o objetivo do desafio de atribuição de tarefas é encontrar uma configuração livre de conflitos em que os pontos de referência sejam percorridos pelos VANTs e minimizem os custos globais. Onde o conjunto $P = \{P_1, P_2, ..., P_{N_t}\}$ indica a rota realizada por cada VANT.

Para isso, presume-se que cada ponto de referência seja visitado por um único VANT, e que cada VANT possa visitar múltiplos pontos seguindo uma ordem específica. Onde os VANTs não apresentam colisão entre si, o que significa que cada ponto de referência precisa ser designado a, no máximo, um VANT (Wang et al., 2022). O ponto de partida e de chegada é uma estação base, também conhecida como galpão. A alocação de tarefas é formulado como um problema MCV, no qual os objetivos são minimizar o custo total percorrido para executar a missão designada aos VANTs (Wang et al., 2022; Santana, 2020).

Desta maneira, o problema de otimização é representado pela Eq. (1) (Wang et al., 2022), onde x_{ijk} assume o valor 1 se o VANT i visitar o ponto de referência k, e assume o valor 0 caso contrário. A variável $c_{jk}^{(i)}$, representa o custo do VANT i ir do ponto de referência j até o ponto k em metros. Já as Eq. (2) e (3) indicam o custo global e o custo parcial máximo dos veículos.

$$\min F = \{f_1(P), f_2(P)\} \tag{1}$$

$$f_1(P) = \sum_{i=1}^{N_v}\sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} c_{jk}^{(i)}(P_i) \tag{2}$$

$$f_2(P) = \max_{1\le i\le N_v} \sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} c_{jk}^{(i)}(P_i) \tag{3}$$

Sujeito às restrições (Wang et al., 2022; Santana, 2020):

$$\sum_{i=1}^{N_v}\sum_{j=2}^{N_t-1} x_{ijk} = 1, \forall k \tag{4}$$

$$\sum_{i=1}^{N_v}\sum_{k=2}^{N_t-1} x_{ijk} = 1, \forall j \tag{5}$$

$$\sum_{i=1}^{N_v}\sum_{j=1}^{N_t} x_{ij1} = N_v \tag{6}$$

$$\sum_{i=1}^{N_v}\sum_{k=1}^{N_t} x_{ink} = N_v, n = N_t \tag{7}$$

$$\sum_{i=1}^{N_v}\sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} = N_t \tag{8}$$

$$f_1(P) \ge \lambda f_2(P) \tag{9}$$

As Eq. (4) e (5) são restrições que indicam que cada ponto de referência, exceto o depósito, pode ser visitado no máximo uma vez. As Eq. (6) e (7) são restrições que garantem que todos os veículos comecem suas rotas no galpão inicial e retornem ao galpão final após finalizarem suas respectivas tarefas. A Eq. (8) é a restrição que assegura que todos os pontos de referência possam ser visitados (Wang et al., 2022; Santana, 2020). E a Eq. (9) indica o equilibro da carga de trabalho no planejamento de rotas (Wang et al., 2022), sendo λ igual 2 neste trabalho.

3. FUNDAMENTAÇÃO TEÓRICA

Nesta seção, é apresentado as principais etapas para a aplicação do ACO multiobjetivo e o uso dos algoritmos A* e DWA para a correção de trajetórias visando o desvio de obstáculos. Os VANTs irão realizar o percurso definido pelo A* visitando cada ponto de referência definido pelo ACO, além disso, é realizado a correção das trajetórias em decorrência de obstáculos móveis usando o DWA. Esta abordagem combina a capacidade de encontrar rotas eficientes do ACO com a capacidade de evitar obstáculos dos algoritmos A* e DWA, proporcionando um planejamento de trajetórias mais seguro e eficiente para os VANTs.

3.1 Sistema de colônia de formigas multiobjetivo

O algoritmo ACO foi criado por Dorigo nos anos 90 com o objetivo de resolver o problema do caixeiro-viajante. A ideia é inspirada no comportamento das formigas, que deixam uma trilha de feromônios enquanto buscam por alimento. As formigas seguintes são atraídas por essa trilha e têm maior probabilidade de seguir o mesmo caminho. Com o passar do tempo, a trilha é reforçada pelas formigas que encontram caminhos mais curtos, resultando em uma solução ótima (Wang et al., 2022).

Para lidar com múltiplos objetivos e veículos, o processo de otimização envolve três etapas principais: configurações básicas e inicialização, construção das soluções e atualização dos feromônios, que são descritas a seguir:

Etapa 1: configurações básicas e inicialização

Nesta etapa são empregados N_g grupos de formigas, cada grupo constrói uma solução a cada iteração, e N_v formigas são empregadas em cada grupo, representando os VANTs utilizados. Para um link entre os pontos de referência r e s, existem dois feromônios $\tau_{rs}^{(1)}$ e $\tau_{rs}^{(2)}$, atribuídas as funções objetivo $f_1(P)$ e $f_2(P)$, respectivamente.

Para inicializar o algoritmo e se obter a matriz de feromônios, uma formiga é selecionada aleatoriamente, e as tarefas mais próximas a ela são atribuídas. O procedimento se repete até que todas as tarefas tenham sido atribuídas às formigas em um grupo. As informações heurísticas entre os pontos r e s para a formiga i em um grupo é inicializada como o inverso do custo, i.e. $\eta_{rs} = 1/c_{rs}$. A matriz de feromônios são atribuídas como $\tau_0^{(1)} = 1/f_1(P^{(0)})$ e $\tau_0^{(2)} = 1/N_v/f_2(P^{(0)})$ (Wang et al., 2022).

Etapa 2: construção das soluções

Na etapa da construção das soluções as tarefas são adicionadas iterativamente uma a uma, para isso, uma formiga precisa ser selecionada primeiro para adicionar uma tarefa não atribuída. A formiga i que será adicionada a próxima tarefa não atribuída é selecionado com base nos critérios da Eq. (10) (Wang et al., 2022; Necula et al., 2015).

Onde $\hat{P}_i$ é uma solução parcialmente construída pela formiga i no grupo, q é um número aleatório entre [0, 1], q_0 é a probabilidade de que a formiga com custo parcial mínimo seja selecionada, enquanto q_1 é a probabilidade de que a formiga com custo parcial máximo seja selecionada, e Int(1,N_v) é um inteiro aleatório entre 1 e N_v, e $C(\hat{P}_i)$ o custo parcial do grupo, é definido pela Eq. (11).

$$\hat{I} = \begin{cases} \arg \min\limits_{1 \le i \le N_v} C(\hat{P}_i), \text{ se } q < q_0 \\ \arg \max\limits_{1 \le i \le N_v} C(\hat{P}_i), \text{ se } q > 1 - q_1 \\ \qquad \text{Int}(1,N_v), \text{ outros casos} \end{cases} \tag{10}$$

$$C(\hat{P}_i) = \sum_{j=1}^{N_t} \sum_{k=1}^{N_t} c_{jk}^{(i)} x_{ijk} \tag{11}$$

Quando a formiga i em um grupo é escolhida, r se torna a última tarefa em $\hat{P}_i$, a tarefa s é selecionado conforme a Eq. (12). Onde C é o conjunto que contém as tarefas que ainda não foram atribuídas, α e β são parâmetros, usados para determinar a importância relativa do feromônio e da informação heurística, respectivamente, p é um número aleatório entre $[0, 1]$, e p_0 é a probabilidade de se escolher o caminho com máximo η_{rs}.

$$s = \begin{cases} \arg \max\limits_{u \in C} [\prod_{k=1}^{2} [\tau_{rs}^{(k)}]^{\alpha k}] \cdot [\eta_{rs}^{(i)}]^{\beta}], \text{ se } p < p_0 \\ \qquad \text{S, outros casos} \end{cases} \tag{12}$$

No termo $[\prod_{k=1}^{2} [\tau_{rs}^{(k)}]^{\alpha k}] \cdot [\eta_{rs}^{(i)}]^{\beta}]$ é indicado a informação de decisão da tarefa, usado para decidir a próxima tarefa a ser atribuída. Se $p < p_0$, a tarefa com informações de decisão máxima é designada, e S é uma variável aleatória selecionada. Após selecionada a tarefa, ela passa a ser atribuída a formiga i e será removido do conjunto não atribuído C. Os procedimentos se repetem até que todas as tarefas tenham sido atribuídas. Ao final, o galpão final, é atribuído a cada formiga (Necula et al., 2015).

Etapa 3: atualização dos feromônios

Cada vez que uma tarefa r em um grupo seleciona s como a próxima tarefa, a regra de atualização de feromônio local é aplicada no link entre a tarefa r e s. O nível de feromônio é atualizado pela a Eq. (13), onde o parâmetro ρ assume valores entre $[0, 1]$ e é usado para equilibrar o feromônio recém-adicionado (Wang et al., 2020; Lu and Yue, 2017).

Já a atualização do feromônio de forma global, ocorre depois que todos os grupos construíram suas soluções. A matriz de feromônios é atualizado na Eq. (14), onde $\Delta\tau_{rs}^{(k)}$ é o feromônio recém-adicionado na ligação, sendo definido na Eq. (15), em que $n_k = 1$ se $k = 1$ e $n_k = N_v$ se $k = 2$ (Wang et al., 2020; Lu and Yue, 2017).

$$\tau_{rs}^{(k)} = (1 - \rho) \cdot \tau_{rs}^{(k)} + \rho \cdot \tau_0^{(k)}, k = 1, 2 \tag{13}$$

$$\tau_{rs}^{(k)} = (1 - \rho) \cdot \tau_{rs}^{(k)} + \rho \cdot \Delta\tau_{rs}^{(k)}, k = 1, 2 \tag{14}$$

$$\Delta\tau_{rs}^{(k)} = \begin{cases} \tau_0^{(k)} + \sum\limits_{l=1}^{|P|} \dfrac{1}{n_k \cdot f_k(P^{(l)})}, \text{ se } p < p_0 \\ \qquad \tau_0^{(k)}, \text{ outros casos} \end{cases} \tag{15}$$

Após a execução de todas as etapas e do fim do algoritmo, as rotas do conjunto P do grupo com melhor resultado na Eq. (1) são selecionadas.

*3.2 Desvios de obstáculos fixos com A**

O problema de encontrar um caminho de um ponto até um destino específico, evitando colisões com os obstáculos do ambiente, ainda é um desafio significativo. Este problema é conhecido como o problema do caminho mais curto com restrições de colisão. O objetivo é encontrar uma rota que minimize a distância percorrida pelo VANT e, ao mesmo tempo, evite colisões com os obstáculos (Gonçalves, 2013).

Algoritmos de busca heurística, como o A*, são capazes de encontrar rotas que evitam colisões, mas podem não garantir uma solução ótima. Ele usa uma função heurística para estimar o custo do caminho mais curto do nó atual até o nó de destino e, em seguida, expande o nó com o menor custo estimado. Isso permite que o algoritmo encontre rapidamente um caminho próximo ao ótimo, evitando a expansão de muitos nós desnecessários (Niu et al., 2021).

O A* tem sido amplamente estudado e aplicado em diversos contextos, incluindo o planejamento de movimento de veículos, especialmente em robótica e veículos autônomos. Este algoritmo permite a correção de trajetórias para desviar de obstáculos predefinidos, evitando colisões e perda de VANTs em missões (Gonçalves, 2013). Neste problema, o espaço de estados é representado como uma matriz, onde cada célula desta matriz é um estado (nó), conforme ilustra

a Fig. 1, as regras que podem ser aplicadas são as 8 direções que o robô pode se mover a partir da célula, conforme os obstáculos (Farlei J. Heinen, 2002).

Com isso, a ideia é executar uma busca no espaço de estados para encontrar a rota mais curta até o destino. É necessário que a posição do robô seja conhecida. A Eq. do algoritmo é apresentada em (16), onde $f(n)$ é o valor estimado do ponto inicial para o ponto de destino, $g(n)$ é o custo real do ponto inicial para o ponto n, ou seja, a distância já percorrida pelo robô desde sua partida, e $h(n)$ é o valor estimado do nó do estado n para o nó de destino.

$$f(n) = g(n) + h(n) \tag{16}$$

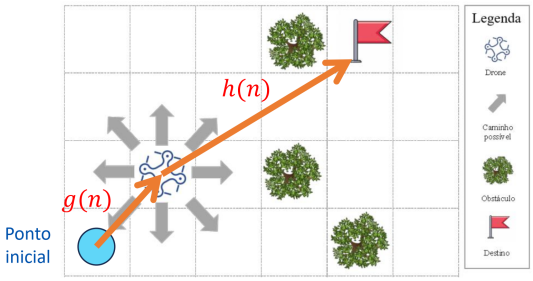

Figura 1. Representação dos caminhos possíveis do A*.

3.3 Desvios de obstáculos móveis com DWA

O DWA é amplamente utilizado para resolver o problema do planejamento local da trajetória de robôs móveis. Este algoritmo é capaz de determinar a cada intervalo de tempo a velocidade do robô, o que permite inferir a trajetória de movimento viável para o período de tempo seguinte. Além disso, o DWA é capaz de determinar a trajetória do veículo por meio da utilização de múltiplas funções de avaliação (Wang et al., 2023; Jin and Wang, 2023).

Essa estratégia permite que o robô escolha a trajetória mais adequada com base em critérios específicos, levando em conta as restrições dinâmicas do ambiente. Esse algoritmo tem se mostrado eficiente na resolução de problemas de planejamento de trajetória para robôs móveis, contribuindo significativamente para aprimorar a navegação e o desempenho desses sistemas. O DWA abrange a direção, velocidade e a segurança do robô na função de avaliação definida na Eq. (17) (Wang et al., 2023).

$$G(v,\omega) = \alpha \cdot Head(v,\omega) + \beta \cdot Vel(v,\omega) + \gamma \cdot Dist(v,\omega) \tag{17}$$

A função $Head(v,\omega)$ é a função de avaliação do azimute, que indica o ângulo entre a direção do final da trajetória e o ponto alvo, quanto menor o ângulo, maior o seu peso. $Vel(v,\omega)$ é uma função de avaliação da velocidade, que representa a velocidade de operação, quanto maior a velocidade, maior o seu peso. $Dist(v,\omega)$ é uma função de avaliação da distância que representa a distância entre a trajetória e o obstáculo mais próximo, um valor muito grande fará com que o desvio da trajetória do caminho planejado aumente (Li et al., 2020; Wang et al., 2023).

A função de avaliação $G(v,\omega)$ indica que o robô está se movendo em direção ao destino a uma grande velocidade enquanto mantém uma distância segura do obstáculo. Os parâmetros α, β, e γ são os coeficientes de ponderação das funções de avaliação, tendo valores respectivamente iguais a 0,3, 0,8 e 0,1(Wang et al., 2023).

O modelo cinemático do robô é definido pela Eq. (18), onde supõe-se que sua trajetória é composta por uma pequena trajetória em arco, cada uma das quais corresponde a um único vetor de velocidade (v_t, ω_t). Dado um intervalo de tempo Δt pequeno, cada trajetória de arco pode ser considerada aproximadamente como uma trajetória em linha reta. Portanto, assumi-se um movimento em linha reta a uma velocidade constante dentro de um intervalo de tempo Δt, e o modelo cinemático do robô é obtido da seguinte forma (Liu et al., 2021):

$$\begin{cases} x_{t+1} = x_t + v_t \cdot \Delta t \cdot \cos\theta_t \\ y_{t+1} = y_t + v_t \cdot \Delta t \cdot \sin\theta_t \\ \theta_{t+1} = \theta_t + \omega_t \cdot \Delta t \end{cases} \tag{18}$$

4. METODOLOGIA

Com os conceitos da Seção 3, os VANTs irão realizar o percurso definido pelo A*, realizando o desvio de obstáculos móveis com o DWA, visitando cada ponto de referência definido pelo ACO, conforme ilustra a Fig. 2.

Um mapa fictício com obstáculos foi utilizado para os experimentos, uma vez que atualmente não há exemplos de referência para MCV a serem utilizados na avaliação de desempenho das abordagens para comparação com a solução ótima. A disposição dos pontos de referência se apresenta de forma esparsada. A Fig. 3 exibe os pontos de referência representados por quadrados amarelos, os obstáculos pré-definidos nas simulações são representados como retângulos de cor cinza, tendo efeito de barreira linear, e também serão aplicados 10 obstáculos móveis, que terão movimento aleatório no mapa.

Os VANTs voam a uma altura de 15m, com velocidade média de 0,43m/s, a unidade de distância entre os pontos de referência será em metro. Já as figuras dos experimentos demonstram os deslocamentos dos VANTs, nos eixos x e y, apresentando o comportamento semelhante a uma partícula, onde o VANTs não presentam colisão entre si.

O ACO utilizou os parâmetros apresentados na Tabela 1 (Wang et al., 2022). Foram realizadas 10 simulações com 10^3 iterações em cada experimento. Isso foi feito devido à natureza dos métodos não-heurísticos, que podem produzir resultados diferentes mesmo com a mesma entrada em diferentes execuções. Dessa forma, foi possível analisar os valores médios, desvio padrão, valores mínimos e máximos das métricas de desempenho.

Tabela 1. Parâmetros utilizados no ACO.

Param.	Descrição	Valor
N_g	Num. de grupos de formiga	24
q_0	Prob. de escolher a formiga com CP_{mn}	0,9
q_1	Prob. de escolher a formiga com CP_{mx}	0,05
α	Grau de importância do feromônio	1
β	Grau de importância da inf. heurística	2
p_0	Prob. de escolher o caminho com máx. η_{rs}	0,9
ρ	Taxa de evaporação do feromônio	0,5

5. RESULTADOS

Nesta seção, apresentamos as simulações realizados ilustrando a aplicação do modelo proposto para o problema

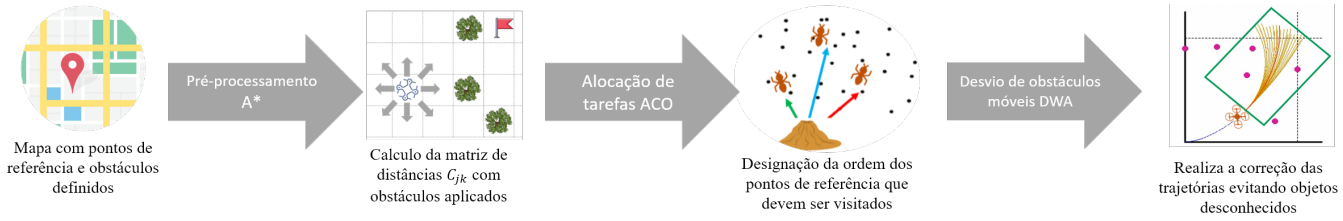

Figura 2. Etapas para alocação de tarefas.

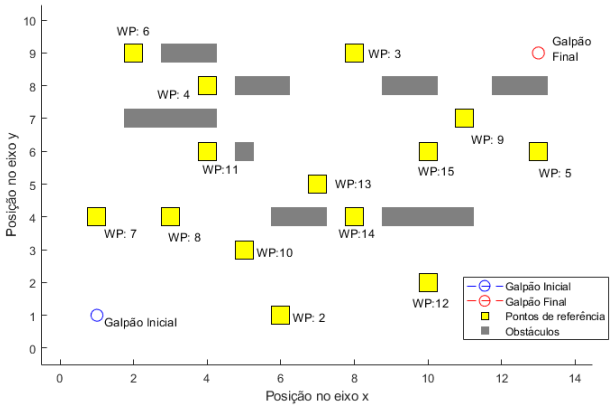

Figura 3. Cenário utilizado nos experimentos.

de planejamento de rotas de VANTs colaborativos. As simulações dos experimentos foram realizadas no *software* Matlab® em um *notebook* com processador Intel(R) Core(TM) i7-7500U @2.70GHz-2.90 GHz com 16Gb RAM. As simulações tiveram variação no número de VANTs, com experimentos de 2 a 6 VANTs. Em todos os experimentos, os VANTs colaborativos iniciam as tarefas partindo do galpão inicial e finalizam a missão visitando o galpão final.

Tabela 2. Comparação dos experimentos

N_v	exp.	média	máx.	mín.	desvio pad.
2	$Dist_{(m)}$	57,54	63,67	54,89	2,60
3	$Dist_{(m)}$	64,26	69,44	61,61	2,26
4	$Dist_{(m)}$	78,67	81,75	75,32	1,91
5	$Dist_{(m)}$	93,84	96,67	91,78	1,68
6	$Dist_{(m)}$	109,30	110,54	108,04	0,86
2	$CPU_{(s)}$	1150,23	1509,76	947,26	205,73
3	$CPU_{(s)}$	1612,28	1952,23	1229,70	213,05
4	$CPU_{(s)}$	1876,20	2162,09	1588,54	170,72
5	$CPU_{(s)}$	2155,46	2700,55	1711,98	342,64
6	$CPU_{(s)}$	1855,28	2112,09	1599,81	170,56
2	$Tmax_{(s)}$	69,76	77,52	64,05	4,71
3	$Tmax_{(s)}$	56,35	64,17	49,87	3,47
4	$Tmax_{(s)}$	54,48	57,45	48,94	2,18
5	$Tmax_{(s)}$	51,95	55,45	46,77	3,63
6	$Tmax_{(s)}$	47,00	47,23	46,70	0,21

A Tabela 2, ilustra os resultados e comparativos dos experimentos, demonstrando que o aumento no número de VANTs leva a uma maior eficiência na execução das missões, com uma redução significativa no tempo total da missão, sendo um resultado já esperado devido à quantidade de agentes envolvidos na missão. No entanto, isso também resulta em um aumento no tempo de processamento e na distância total percorrida, o que pode ser um fator limitante em cenários com recursos computacionais restritos e com limitações de distância de voo nos VANTs.

Já a Fig. 4 apresenta a movimentação de 2 VANTs, desviando os obstáculos fixos e móveis, onde os pontos de referência marcados em vermelho e azul são os pontos visitados pelos VANTs 1 e 2, respectivamente. Devido o comportamento dinâmico dos veículos, os resultados dos melhores experimentos de 2 a 6 VANTs foram gravados e disponibilizados no *YouTube* em: https://encr.pw/DWA-cba-resultados2024. Onde os asteriscos na Fig. 4 representam os obstáculos que se movimentam de forma aleatória.

Ademais, a junção dos algoritmos ACO, A* e DWA, tornaram os VANTs capazes de desviar de obstáculos e de adaptar sua velocidade e trajetória em tempo real. Sendo especialmente importante em cenários dinâmicos, onde os obstáculos podem se mover ou surgir inesperadamente. Nos casos onde um obstáculo móvel era detectado, o DWA agia no ajuste da velocidade e a direção para evitar colisões através da sua função de avaliação, garantindo a segurança e a eficiência das rotas planejadas pelo ACO e A*.

6. CONCLUSÃO

Neste trabalho foi discutido o planejamento de rotas para múltiplos VANTs com desvio de obstáculos fixos e móveis. O problema foi formulado como um MCV multiobjetivo, visando minimizar a distância total do percurso. Simulações foram realizadas no Matlab® com um algoritmo híbrido baseado em ACO, A* e DWA, seguindo as restrições do problema proposto. Para a validação da abordagem foram feitos 10 experimentos com 10^3 iterações com variações no número de VANTs de 2 a 6.

O modelo proposto apresenta bons resultados no mapa fictício, concluindo as tarefas de visitas os pontos de referência designados com diferentes números de VANTs, uma vez que atualmente não há exemplos de referência para MCV a serem usados para avaliar o desempenho das abordagens. Observou-se através da Tabela 2 que o tempo de processamento para o planejamento das rotas tende a aumentar com o número maior de VANTs, entretanto o número maior de VANTs garante um tempo de missão menor. Neste caso o algoritmo poderá ser utilizado para análise de viabilidade para a realização de missões.

Além disso, os algoritmos A* e DWA se mostraram eficientes no desvio de obstáculo fixos e móveis, apresentando potencial para cenários mais complexos. Podendo ser aplicado em ambientes controlados no cálculo de rotas para veículos e robôs autônomos, estendendo seu uso às aplicações relacionadas ao conceito de indústria 4.0. Para trabalhos futuros, planeja-se abordar o planejamento de rotas para múltiplos VANTs colaborativos com restrições energéticas, tornando o problema o mais próximo do real, além de verificar o método com VANTs reais.

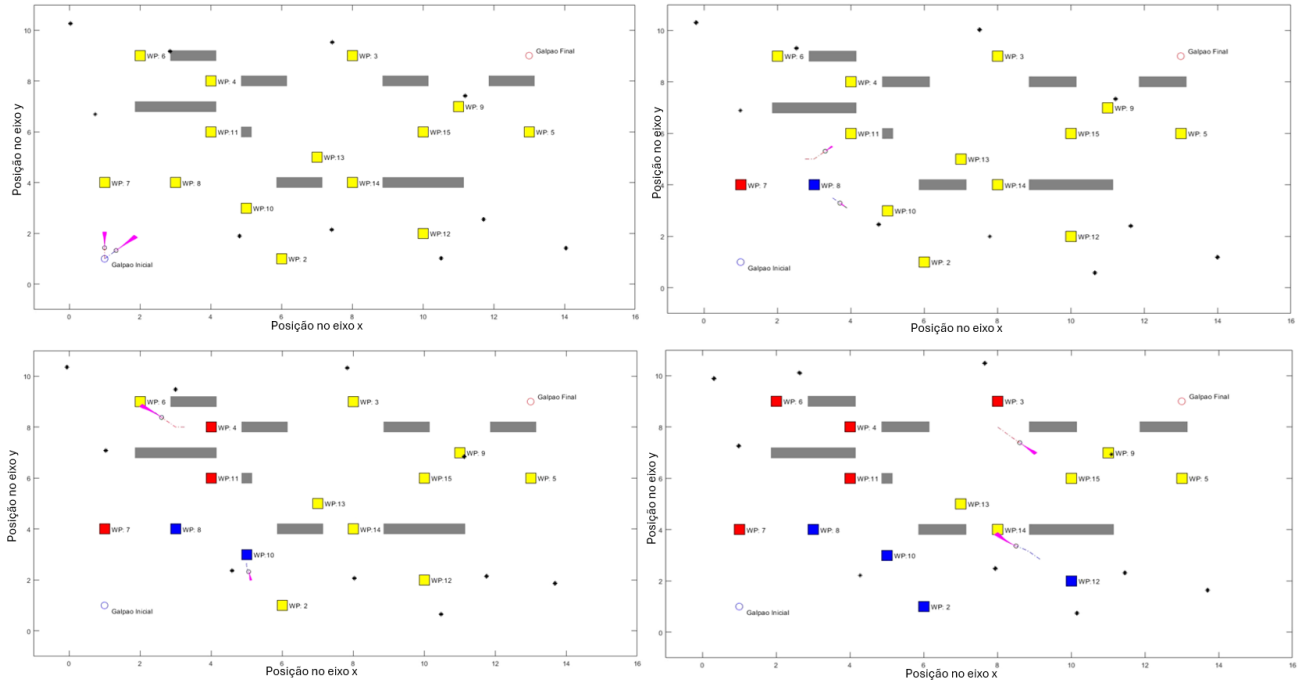

Figura 4. Movimentação dos VANTs.

REFERÊNCIAS

Aguiar, A.L.S., Pinto, V.P., and Rodrigues, L.R. (2023). Método multiobjetivo para alocação de tarefas de múltiplos robôs colaborativos. In *Anais do Encontro Nacional de Modelagem Computacional e Encontro de Ciência e Tecnologia dos Materiais*. Nova Friburgo, RJ, Brasil.

Farlei J. Heinen, F.S.O. (2002). Sistema de controle híbrido para robôs móveis autônomos.

Floreano, D. and Wood, R.J. (2015). Science, technology and the future of small autonomous drones. *nature*, 521(7553), 460–466.

Freitas, E. and Carvalho, J. (2015). Genetic algorithm approach for a class of multi-criteria, multi-vehicle planner of uavs. volume 9019, 234–248.

Gonçalves, M.A. (2013). Algoritmo a-estrela de estado híbrido aplicado à navegação autônoma de veículos.

Jin, M. and Wang, H. (2023). Robot path planning by integrating improved a* algorithm and dwa algorithm. In *Journal of Physics: Conference Series*, volume 2492, 012017. IOP Publishing.

Ju, C. and Son, H.I. (2018). Multiple uav systems for agricultural applications: Control, implementation, and evaluation. *Electronics*, 7(9), 162.

Li, X., Hu, X., Wang, Z., and Du, Z. (2020). Path planning based on combinaion of improved a-star algorithm and dwa algorithm. In *2020 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)*, 99–103.

Liu, L., Yao, J., He, D., Chen, J., Huang, J., Xu, H., Wang, B., and Guo, J. (2021). Global dynamic path planning fusion algorithm combining jump-a* algorithm and dynamic window approach. *IEEE Access*, 9.

Liu, Y., Zhang, Y., Zhang, W., Zhang, Y., and Zhang, J. (2019). Reinforcement learning in multiple-uav networks: Deployment and movement design. *IEEE Transactions on Vehicular Technology*, 8037–8049.

Lu, L.C. and Yue, T.W. (2017). Mission-oriented ant-team aco for min-max mtsp. In *2017 International Conference on Information, Communication and Engineering (ICICE)*, 522–525. doi:10.1109/ICICE.2017.8478886.

Necula, R., Breaban, M., and Raschip, M. (2015). Tackling the bi-criteria facet of multiple traveling salesman problem with ant colony systems. In *2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*, 873–880.

Niu, C., Li, A., Huang, X., Li, W., and Xu, C. (2021). Research on global dynamic path planning method based on improved a algorithm. *Mathematical Problems in Engineering*, 2021. doi:10.1155/2021/4977041.

Pinto, V., Galvão, R., Rodrigues, L., and Gomes, J.P. (2020). Mission planning for multiple uavs in a wind field with flight time constraints. *Journal of Control, Automation and Electrical Systems*, 31. doi:10.1007/s40313-020-00609-5.

Santana, K. (2020). Método metaheurístico multi-objetivo para o cálculo de rotas de robôs colaborativos com restrições energéticas.

Wang, G., Jiang, C., Tao, G., and Ye, C. (2023). Dynamic path planning based on the fusion of improved rrt and dwa algorithms. In *2023 4th International Conference on Mechatronics Technology and Intelligent Manufacturing (ICMTIM)*, 534–538.

Wang, M., Ma, T., Li, G., Zhai, X., and Qiao, S. (2020). Ant colony optimization with an improved pheromone model for solving mtsp with capacity and time window constraint. *IEEE Access*, 8, 106872–106879.

Wang, S., Liu, Y., Qiu, Y., Zhang, Q., Huo, F., Huangfu, Y., Yang, C., and Zhou, J. (2022). Cooperative task allocation for multi-robot systems based on multi-objective ant colony system. *IEEE Access*, 10, 56375–56387. doi:10.1109/ACCESS.2022.3165198.