

COORDENAÇÃO DE VEÍCULOS AÉREOS NÃO TRIPULADOS UTILIZANDO INTELIGÊNCIA DE ENXAMES

Trabalho de Conclusão de Curso
Engenharia de Computação

Aluno: Diego Marconi Pinheiro Ferreira Silva

Orientador: Prof. Dr. Carmelo José Albanez Bastos Filho

Diego Marconi Pinheiro Ferreira Silva

***Coordenação de veículos aéreos não
tripulados utilizando inteligência de enxames***

Monografia apresentada para obtenção do
Grau de Bacharel em Engenharia de Com-
putação pela Universidade de Pernambuco

Orientador:

Prof. Dr. Carmelo José Albanéz Bastos Filho

GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO
ESCOLA POLITÉCNICA DE PERNAMBUCO
UNIVERSIDADE DE PERNAMBUCO

Recife - PE, Brasil

Novembro de 2011

Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação
Projeto de Final de Curso

POLI
ESCOLA POLITÉCNICA
DE PERNAMBUCO

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 20 de Dezembro de 2011, às 13:00 horas, reuniu-se para deliberar a defesa da monografia de conclusão de curso do discente DIEGO MARCONI PINHEIRO FERREIRA SILVA, orientado pelo professor Carmelo Bastos Filho, sob título Coordenação de veículos aéreos não tripulados utilizando inteligência de enxames, a banca composta pelos professores:

Bruno José Torres Fernandes

Carmelo Bastos Filho

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada Aprovada com Restrições* Reprovada

e foi-lhe atribuída nota: 9,5 (Nove e meio)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O discente terá 5 dias para entrega da versão final da monografia a contar da data deste documento.

B. J. T. F.

BRUNO JOSÉ TORRES FERNANDES

Carmelo Bastos Filho

CARMELO BASTOS FILHO

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

Resumo

Este trabalho de conclusão de curso apresenta um simulador construído para possibilitar a visualização da coordenação de múltiplos veículos aéreos não tripulados (UAVs) utilizando inteligência de enxames. O objetivo do simulador construído foi utilizar conceitos de inteligência de enxames e enxames de robôs na modelagem de UAVs, tendo como objetivo a realização de atividades de patrulha territorial e rastreamento de alvos móveis. Foram utilizados mecanismos de locomoção utilizando dinâmica física, mecanismos sensoriais anti-colisão, de percepção ambiental, mecanismos para evitar a saída do espaço aéreo, retorno à base quando necessário, etc. Os UAVs iniciam suas atividades em ambientes desconhecidos e devem encontrar outros UAVS durante o estado de agrupamento e formar com eles uma rede de comunicações *ad hoc*, se tornando ponte de roteamento do enxame em que se encontram. Após conectados, os UAVS iniciam as atividades de patrulha e, ao encontrarem algum alvo, iniciam o rastreamento. Todas as atividades são executadas de forma coordenada. Os resultados evidenciaram que a comunicação existente determina a efetividade do sistema. Também foi posto em evidência que múltiplos UAVs coordenando suas atividades com as estratégias, mecanismos e abordagens propostas são mais efetivos que um único UAV ou múltiplos UAVs descoordenados. Houveram resultados visuais interessantes como, por exemplo, a formação natural, a partir das posições dos UAVs, de figuras geométricas simétricas, a saber, triângulos equiláteros, quadrados, pentágonos e hexágonos, dependendo do número de UAVs envolvidos no rastreamento. Outro resultado visual interessante foi a formação, também natural, de sub-enxames, tendo como consequência o aumento da área territorial coberta e a possibilidade do rastreamento de mais de um alvo paralelamente.

Abstract

This work presents a simulator built to enable the visualization of multiple unmanned aerial vehicles' (UAVs) coordination using swarm intelligence. The built simulator's purpose was to model UAVs using swarm intelligence and swarm robots concepts, in order to perform territorial patrolling and mobile targets tracking activities. Mechanisms were used for locomotion, such as physics dynamics, anti-collision sensory mechanisms, environment perception, avoids airspace's exit mechanisms, and, when needed, return to base, etc. All UAVs starts their activities in a unknown environment and must find other UAVs while in group state and create with them an ad hoc communication network, becoming a router bridge in the swarm. Once connected, the UAVs starts their patrolling activities and if a target was found, starts the targets tracking. All activities are performed well coordinated. The results showed that the communication's existence determines the system's effectiveness. It was also evidenced that multiple UAVs coordinating their activities are more effective than a single UAV or multiple uncoordinated UAVs. There were interesting visual results, such as natural formation, from the UAVs positions, of symmetric geometric figures, namely equilateral triangles, squares, pentagons and hexagons, depending on the number of UAVs involved in the tracking. Another result was the formation, also natural, of sub-swarms. The consequences were the increased covered area and track more than one target in parallel.

Dedicatória

Dedico este trabalho a todos, principalmente os brasileiros, que acreditam, e fazem outros acreditarem, que podem fazer a diferença, sendo agentes da mudança para uma nação melhor.

Agradecimentos

Primeiramente, agradeço a conclusão deste trabalho ao meu Senhor e meu Deus, o Cristo Jesus e a sua mãe, e minha também, Nossa Senhora, Maria Santíssima.

Agradeço a toda minha família, especialmente, a minha mãe Anita Márcia Pinheiro Ferreira Silva e ao meu pai Marconi José Ferreira Silva, aos meus irmãos Victor e Marcianita, pois, sempre estiveram ao meu lado em todas as ocasiões e me apoiaram em tudo.

Agradeço também à minha namorada Erika Lemos, pela compreensão e carinho durante quase todo o curso e, principalmente, no período deste trabalho.

Agradeço ao meu colega de pesquisa, Luiz Oliveira, pelo empenho e envolvimento para que este projeto se concretizasse.

Agradeço ao meu orientador, o professor Carmelo J. A. Bastos-Filho, por ter sido um ótimo orientador, fonte de motivação neste projeto e ser, atualmente, a minha referência de docência e pesquisa.

Agradeço ao professor Fernando Buarque de Lima Neto por ter sido o principal motivador, desde o primeiro período da Universidade na cadeira de Introdução à Engenharia, da minha decisão pela carreira acadêmica.

Agradeço também aos professores que contribuíram de forma significativa para a minha formação e, especialmente, aos professores Sérgio Campello, Denis Silveira, Gustavo Carvalho e Mêuser Valença.

Agradeço aos amigos que fiz na Universidade e que participaram de uma das fases mais importantes de minha vida, principalmente, Erick Barboza, David Alain, Marcelo Lacerda e Paulo Roger.

*“Amarás o Senhor teu Deus de todo o teu coração,
e de toda a tua alma, e de todo o teu pensamento.
Este é o primeiro e grande mandamento; e o segundo,
semelhante a este é: Amarás teu próximo como a ti mesmo.”*

Mateus 22, 37-39

Sumário

Lista de Figuras	xi
Lista de Algoritmos	xi
Lista de Abreviaturas e Siglas	xii
1 Introdução	1
1.1 Motivação e Caracterização do Problema	1
1.2 Hipóteses e Objetivos	3
1.2.1 Objetivos Gerais	3
1.3 Organização do Documento	3
2 Fundamentação teórica	5
2.1 Fundamentos de inteligência de enxames	5
2.2 Fundamentos de enxames de robôs	8
3 Descrição dos componentes do sistema	14
3.1 Modelo do ambiente	14
3.2 Modelo do agente	17
3.2.1 Mecanismo de localização	18
3.2.2 Mecanismo de locomoção	19
3.2.3 Mecanismo anti-colisão	22
3.2.4 Mecanismos de comunicação	25
3.2.5 Mecanismo de percepção do ambiente	27

3.2.6	Outros mecanismos	29
4	Descrição do comportamento do sistema	31
4.1	Máquina de estados do UAV	31
4.2	Estratégia de sincronismo	32
4.3	Estratégia de agrupamento	33
4.4	Estratégia de patrulha	34
4.5	Estratégia de rastreamento	34
4.6	Estratégia de retorno à base	36
4.7	Rotina principal	37
5	Apresentação do simulador proposto e suas funcionalidades	38
5.1	Introdução à ferramenta computacional	38
5.2	Parâmetros de simulação	40
5.2.1	Ambiente	41
5.2.2	UAV	41
5.2.3	Sensor anti-colisão	42
5.2.4	Sensor de percepção do ambiente	43
5.2.5	Sensor de percepção dos limites do ambiente	43
5.2.6	Sensor de comunicação	44
5.2.7	Alvos	44
5.2.8	Obstáculos	45
5.2.9	Simulação	46
5.3	Ações do simulador	46
5.4	Estatísticas da simulação	47
6	Experimentos e resultados das simulações	49
6.1	Resultados visuais	49

6.1.1	Formação de figuras geométricas simétricas	49
6.1.2	Formação de sub-enxames	53
6.2	Resultados numéricos	54
6.2.1	Conjunto experimental 1	55
6.2.2	Conjunto experimental 2	57
6.3	Discussões dos resultados obtidos	59
7	Conclusões e Trabalhos Futuros	62
	Referências Bibliográficas	63
	Apêndice A – Especificação de requisitos do simulador	67

Lista de Figuras

3.1	Ambiente virtual: Fronteira do Brasil com o Paraguai.	15
3.2	UAV: Veículo Aéreo Não Tripulado.	16
3.3	Alvo: Um invasor no ambiente.	16
3.4	Obstáculo: Qualquer estrutura fixa no ambiente.	16
3.5	Base terrestre.	17
3.6	Raio de atuação dos sensores do UAV.	22
3.7	Cálculo da intensidade de atuação do sensor anti-colisão.	24
3.8	Cálculo da intensidade de atuação do sensor de comunicação.	27
4.1	Máquina de estados do UAV.	32
5.1	Visão geral do simulador.	39
5.2	Painel de simulação.	40
5.3	Parâmetros do ambiente.	41
5.4	Parâmetros dos UAVs.	42
5.5	Parâmetros do sensor de colisão.	42
5.6	Parâmetros do sensor de percepção do ambiente.	43
5.7	Parâmetros do sensor de percepção dos limites do ambiente.	44
5.8	Parâmetros do sensor de comunicação.	44
5.9	Parâmetros dos alvos.	45
5.10	Parâmetros dos obstáculos.	45
5.11	Ações do simulador.	47
5.12	Exemplo das estatísticas de simulação.	47

6.1	Triângulo equilátero, contendo o alvo em seu centro, formado por três UAVs em estado de rastreamento.	50
6.2	Quadrado, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.	50
6.3	Pentágono regular, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.	51
6.4	Hexágono simétrico não regular, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.	51
6.5	Pentágono regular, contendo dois UAVs e o alvo em seu centro, formado por sete UAVs em estado de rastreamento.	52
6.6	Hexágono regular, contendo dois UAVs e o alvo em seu centro, formado por oito UAVs em estado de rastreamento.	52
6.7	Formação de sub-enxames: conectividade global existente.	53
6.8	Formação de sub-enxames: conectividade global baixa.	54
6.9	Formação de sub-enxames: conectividade global crítica.	55
6.10	Formação de sub-enxames: conectividade global inexistente.	56
6.11	Experimento 1: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvmax} = v_{uavmax}$	58
6.12	Experimento 1: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvmax} > v_{uavmax}$	59
6.13	Experimento 2: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvmax} = v_{uavmax}$	61
6.14	Experimento 2: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvmax} > v_{uavmax}$	61

Lista de Algoritmos

1	Pseudocódigo do PSO	8
2	Estratégia de sincronismo.	33
3	Estratégia de agrupamento.	34
4	Estratégia de patrulha	34
5	Estratégia de rastreamento	36
6	Estratégia de retorno à base	36
7	Rotina principal de cada UAV	37

Lista de Abreviaturas e Siglas

ACO – *Ant Colony Optimization*

BDI – *Belief-Desire-Intention*

GPS – *Global Point System*

PSO – *Particle Swarm Optimization*

RADAR – *Radio Detection And Ranging (Detecção e Telemetria pelo Rádio)*

SI – *Sistemas internacional*

SMA – *Sistemas multiagentes*

UAV – *Unmanned Aerial Vehicle*

1 *Introdução*

*“Enquanto mais me aprofundo nas ciências,
mais me aproximo de Deus.”*

– **Albert Einstein.**

Neste trabalho de conclusão de curso, foi construído um simulador para tornar possível observar a coordenação, projetar ações e desenvolver algoritmos para enxame de UAVs (Veículo Aéreo Não Tripulado) considerando os seguintes fatores: dinâmica dos UAVs, como por exemplo, aceleração e direcionamento; inteligência de enxames na coordenação dos UAVs (Veículos Aéreos não Tripulados), como por exemplo, afastamento, colisão, rastreamento de alvos e compartilhamento de informação; e comunicação entre os UAVs, utilizando o paradigma de comunicação *ad hoc*. O simulador pode auxiliar no projeto, simulação e implantação de um enxame real de UAVs em campo.

Este capítulo apresenta a introdução deste documento, e está organizado em 3 seções. Na Seção 1.1, é apresentada a motivação para a realização deste trabalho, bem como o problema abordado pelo mesmo. Em seguida, na Seção 1.2 são apresentados os objetivos gerais e específicos. Por fim, na Seção 1.3 é descrita a estrutura do restante do documento.

1.1 **Motivação e Caracterização do Problema**

Os UAVs tem recebido bastante interesse da comunidade científica nos últimos anos e graças aos investimentos em computação embarcada, comunicações, sensores e redução consumo de energia, os UAVs estão cada vez mais capazes de desempenhar tarefas bastante complexas e sofisticadas. Como não são ocupados por nenhum humano, os UAVs são geralmente mais simples e de um custo menor quando comparados com outros veículos aéreos, sendo adequados para missões longas e

perigosas, como monitoramento de tráfico urbano, operações militares de busca e salvamento e patrulha de fronteira [1]. Existem vários tipos de UAVs para diferentes propósitos, como os acima mencionados, utilizados e construídos por vários países, principalmente, Estados Unidos e Israel [2].

No entanto, implantar um enxame real de UAVs requer um alto investimento, visto que o hardware necessário para o enxame terá de ser implementado inicialmente. Um ambiente que pudesse ser capaz de simular o comportamento do enxame, poderia reduzir o custo envolvido para implantação e oferecer, previamente, informações acerca do comportamento do enxame de UAVs. Para permitir que os UAVs se engajem em missão de longa durações, não tendo que recarregar sua energia frequentemente, o consumo de energia não poderá ser alto. Dessa forma, uma comunicação de longo alcance, que por sua vez requer transmissores de alta potência, se torna inviável por conta do consumo de energia. Esse problema pode ser resolvido utilizando um paradigma de comunicação *ad hoc*, onde todos os UAVs podem funcionar como roteador. Ao invés de transmitir a informação direto para o destino, a informação é transmitida para o UAV imediatamente vizinho, e assim sucessivamente, até que a informação chegue ao destino.

Uma ferramenta computacional que simula a inspeção de dutos submarinos utilizando inteligência de enxames em robôs aquáticos foi proposto em [3], que por sua vez mostra que múltiplos robôs podem realizar a inspeção de forma mais efetiva que um único robô e que, dependendo das características do ambiente, a partir de um determinada quantidade de robôs, também foi mostrado em [3] que aumentar demasiadamente o número de robôs não gera benefícios adicionais relevantes, somente custos.

Um simulador de UAVs, utilizando inteligência de enxame, considerando os aspectos de comunicação, colisão, afastamento e rastreamento de alvos, ajudaria, também, a guiar o projeto e configuração do enxame necessário para implantação real em campo.

O relato de algumas experiências com UAVs, inclusive de implementação prática, podem ser verificados em [1] [4] [5]. Entretanto, nenhum destes trabalhos propôs um mecanismo de coordenação autônomo de um enxame de UAVs considerando colisão, afastamento, comunicação *ad hoc* e rastreamento de alvos móveis ao mesmo tempo.

1.2 Hipóteses e Objetivos

Neste simulador será analisado o desempenho dos UAVs mediante a variação no número de indivíduos no enxame, e qual o impacto na efetividade da tarefa que lhes foi designada.

1.2.1 Objetivos Gerais

O objetivo geral deste trabalho de conclusão de curso é construir um simulador que, fazendo uso de algoritmos de inteligência de enxames, seja capaz de prover um ambiente onde se possa observar o comportamento autônomo de UAVs, que compartilham informações entre si utilizando comunicação *ad hoc*.

São objetivos específicos:

- Aplicar técnicas de inteligência de enxame na coordenação dos UAVs;
- Construir algoritmos de coordenação de enxame de UAVs;
- Construir uma ferramenta computacional que simule o comportamento de um enxame de UAVs para patrulha territorial;
- Realizar simulações levando em conta a dinâmica dos UAVs, sensores de percepção ambiental, anti-colisão e comunicação;
- Analisar os resultados em relação às métricas de colisões, cobertura territorial e alvos rastreados.

1.3 Organização do Documento

O trabalho está organizado em 7 capítulos. No Capítulo 2 serão abordados os conceitos de inteligência de enxames bem como de enxames de robôs. Em seguida, no Capítulo 3, serão descritos os componentes do sistema, a saber, o modelo do ambiente e o modelo dos agentes e seus mecanismos. O Capítulo 4 especifica o comportamento do sistema, as estratégias de patrulha, rastreamento dos UAVs e os procedimentos de comunicação. No Capítulo 5 serão apresentados os componentes e funcionalidades do simulador. No Capítulo 6 serão descritos os experimentos realizados, os resultados obtidos nas simulações e as considerações finais sobre os

mesmos. Por fim, no Capítulo 7, serão apresentadas as principais conclusões desse trabalho, bem como propostas para trabalhos futuros.

2 *Fundamentação teórica*

“A menos que se admita a existência de Deus, a questão que se refere ao propósito para a vida não tem sentido.”

– **Bertrand Russel.**

Neste capítulo é apresentada a base necessária para o entendimento do simulador desenvolvido. A seção 2.1 apresenta os fundamentos de inteligência de enxames, de modo especial, otimização por enxame de partículas. A seção 2.2 apresenta os fundamentos relativos aos de enxames de robôs.

2.1 **Fundamentos de inteligência de enxames**

A metáfora de insetos sociais para solução de problemas se tornou uma linha de pesquisa em alta nos últimos 15 anos, visto que essa abordagem coloca em ênfase aspectos como distributividade, interação direta e indireta entre agentes relativamente simples, flexibilidade e robustez [6]. O número de aplicações de sucesso utilizando tal metáfora cresce exponencialmente em otimização combinatorial, redes de comunicação e robótica. Muitos pesquisadores estão interessados nessa nova maneira de alcançar tal forma de inteligência, cunhada como inteligência de enxames.

Inteligência de enxames (*Swarm Intelligence*) teve sua origem a partir de estudos em colônias, ou enxames de organismos simples. Os estudos em enxames de organismos sociais levaram à construção de algoritmos de otimização e agrupamento bastante eficientes [7]. Por exemplo, os estudos da elegante, mas imprevisível, coreografia de bandos de pássaros levou ao desenvolvimento do algoritmo de otimização baseado em enxame de partículas (PSO) e o estudo de colônias de formigas em suas atividades como construção de estruturas, proteção da rainha, busca por fontes de al-

imentos e estratégias de ataques, levou ao desenvolvimento do algoritmo de otimização baseado em colônia de formigas (ACO) [8].

Em 1995, James Kennedy e Russel Eberhart, criaram uma técnica de otimização estocástica chamada PSO para simular graficamente o comportamento de um bando de pássaros [9]. Um enxame pode ser descrito como uma coleção estruturada de organismos interativos (ou agentes), onde cada indivíduo é extremamente simples, no entanto, o comportamento coletivo apresentado se torna bastante complexo. O comportamento global do enxame de organismos sociais, portanto, emerge de maneira não linear através do comportamento dos indivíduos dentro do enxame, existindo assim um forte acoplamento entre o comportamento individual e o comportamento do enxame inteiro. Dessa forma, o comportamento coletivo dos indivíduos especifica o comportamento do enxame. No entanto, o comportamento do enxame não é determinado somente pelo comportamento dos indivíduos, independentemente de outros indivíduos. Em vez disso, a interação entre indivíduos desempenha o papel vital para a formação do comportamento do enxame [8].

Um aspecto que dirige o PSO é a interação social, onde indivíduos dentro do enxame aprendem uns com os outros e, baseado no conhecimento obtido, cada indivíduo, move-se com o objetivo de tornar-se mais próximo de seus melhores vizinhos. Tal estrutura social é determinada pela formação de vizinhanças, onde os indivíduos, dentro dos limites dessa vizinhança, comunicam-se uns com os outros.

Existem diferentes tipos de vizinhanças que foram definidas e estudadas, como por exemplo: topologia em estrela (*star topology*) [10], onde cada partícula pode comunicar-se com qualquer outra dentro do enxame, formando um rede social totalmente conectada; e a topologia em anel (*ring topology*) [10], onde cada partícula comunica-se com o seus n vizinhos imediatos e o valor de n é um inteiro não negativo menor que o tamanho do enxame.

Em um enxame, cada partícula representa uma solução potencial. As partículas voam pela hiperespaço, modificando seu posicionamento baseado em sua própria experiência e nas experiência de seus vizinhos. Cada partícula armazena as seguintes informações:

- $\vec{x}_i(t)$ (posição atual) – um vetor de valores reais que são os parâmetros para a solução do problema;
- $\vec{v}_i(t)$ (velocidade) – um vetor que determina como se dará a movimentação das

partículas;

- $\vec{P}_{besti}(t)$ – a melhor posição encontrada pela partícula;
- $\vec{G}_{besti}(t)$ – a melhor posição encontrada pelo enxame, ou seja, o $\vec{P}_{best}(t)$ da melhor partícula;
- *fitness* – o valor que define a partícula qualitativamente. Geralmente, é o retorno da função a ser otimizada.

A posição $\vec{x}_i(t)$ é modificada pela adição do vetor velocidade $\vec{v}_i(t)$ ao seu posicionamento. O comportamento do PSO consiste em, a cada iteração, alterar o vetor velocidade da partícula de acordo com as posições $\vec{P}_{best}(t)$ e o $\vec{G}_{best}(t)$. A velocidade das partículas, no PSO, é regida pela seguinte equação [9]

$$\vec{v}_i(t+1) = \omega \vec{v}_i(t) + c_1 r_1 [\vec{P}_{besti} - \vec{x}_i(t)] + c_2 r_2 [\vec{G}_{best} - \vec{x}_i(t)], \quad (2.1)$$

onde $\vec{v}_i(t+1)$ será a nova velocidade da partícula \vec{P}_i , enquanto $\vec{v}_i(t)$ e $\vec{x}_i(t)$ são, respectivamente, a velocidade atual e posição atual da partícula. Existem dois componentes que influenciam o comportamento do PSO: o primeiro é o componente cognitivo c_1 , que denota a influência cognitiva individual da partícula em relação a melhor posição encontrada a partir de sua experiência pessoal, \vec{P}_{best} , no cálculo de sua velocidade; o segundo é o componente social c_2 , que especifica a influência social em relação à melhor posição encontrada pelo grupo, \vec{G}_{best} , no cálculo de sua velocidade. Para conceder a característica contínua ao PSO, no cálculo da velocidade, são escolhidos dois coeficientes aleatórios r_1 e r_2 no intervalo $[0,1]$, associados respectivamente ao componente cognitivo e social. Por fim, o fator de inércia ω provê o balanceamento entre a busca em amplitude e profundidade tendo aumentado a eficácia das soluções encontradas em um grande número de aplicações [11].

A velocidade da partícula, após efetuado seu cálculo, é utilizada no cálculo da nova posição da partícula. A posição atual é somada à nova velocidade, gerando assim a posição final da partícula, como apresentado na Equação (2.2):

$$\vec{x}_i(t+1) = \vec{v}_i(t+1) + \vec{x}_i(t). \quad (2.2)$$

Como o tempo é a unidade de iteração e possui valor unitário no PSO, não é levado em consideração como um fator que multiplica a velocidade.

A atualização das velocidades e posições no PSO é realizada de tal maneira que

esse tipo de técnica mostra-se eficiente na resolução de problemas em que o espaço de variável é contínuo.

Dessa maneira, as principais propriedades identificadas na inteligência coletiva são:

- Adaptabilidade: habilidade de adaptação à variações ambientais;
- Estabilidade: nem todas as variações afetam o comportamento de um agente;
- Diversidade: habilidade de reação à situações inesperadas;
- Qualidade: capacidade de avaliação comportamental por parte dos agentes;
- Proximidade: capacidade de interação por parte dos agentes.

Em resumo, o PSO funciona de acordo com o pseudocódigo apresentado no Algoritmo 1.

Algorithm 1: Pseudocódigo do PSO .

- 1 Inicialize aleatoriamente as partículas no espaço e busca;
 - 2 Avalie o *fitness* de cada partícula;
 - 3 Determine o \vec{G}_{best} ;
 - 4 **enquanto** critério de parada não for alcançado **faça**
 - 5 **para cada partícula faça**
 - 6 Atualize velocidade e posição (Equação 2.1 e 2.2);
 - 7 Avalie o *fitness* da partícula;
 - 8 Atualize o \vec{P}_{besti} ;
 - 9 Atualize o \vec{G}_{best} ;
 - 10 Retorne o \vec{G}_{best} .
-

2.2 Fundamentos de enxames de robôs

Enxames de robôs é uma nova abordagem para coordenação de um grande número de robôs relativamente simples, autônomos, não controlados de maneira centralizada, capazes de se comunicar localmente e que operam baseados em inspiração biológica [12]. Enxames de robôs pode ser definido como o estudo de como um grande número de agentes corporificados relativamente simples fisicamente podendo ser projetados de forma que o comportamento coletivo desejado venha emergir através das

interações locais entre os agentes e entre os agentes e o ambiente. Inteligência de enxames pode ser entendida como uma propriedade de um sistema de robôs não inteligentes exibindo coletivamente um comportamento inteligente [13]. A aplicação dos princípios de inteligência de enxames em uma coleção de robôs pode ser denominada como enxame de robôs (*Swarm robotics*) [12].

Segundo [14], um enxame de robôs deve apresentar três propriedades funcionais que são observadas em enxames naturais e permanecem como propriedades desejadas em um sistema multi-robôs. São elas:

1. Robustez

A propriedade robustez apresenta a capacidade de continuar suas operações apesar de distúrbios no ambiente ou falha de algum indivíduo. Primeiro, os enxames são inerentemente sistemas redundantes; a perda de um indivíduo não causa uma falha no sistema, podendo aquele ser substituído por outro. Segundo, a coordenação é descentralizada, então é improvável que a destruição de uma parte particular do enxame comprometa sua operação. Terceiro, os indivíduos que compõem o enxame são relativamente simples, tornando-os menos propensos à falhas. Quarto, a percepção é distribuída, por isso, o sistema é robusto contra perturbações locais no ambiente.

2. Flexibilidade

A propriedade flexibilidade apresenta indivíduos com a habilidade de coordenar seus comportamentos de modo que sejam capazes de enfrentar tarefas de diferentes naturezas. Por exemplo, a capacidade que os indivíduos em uma colônia de formigas possuem de encontrar o menor caminho entre o ninho e a fonte de alimento ou transportar uma presa grande através da utilização de diferentes estratégias de coordenação.

3. Escalabilidade

A propriedade escalabilidade apresenta a capacidade de operar com grupos de tamanho variados e em um amplo intervalo, suportando também um número demasiado grande de indivíduos sem impactar o desempenho consideravelmente. Sendo assim, os mecanismos e estratégias de coordenação a serem desenvolvidos em um sistema de enxame de robôs devem assegurar a operação do enxame sob variados tamanhos de enxames.

Dentro dessas propriedades, o maior interesse dos pesquisadores é a resolução

de problemas de cobertura e exploração de ambientes de maneira eficiente, controle distribuído, difusão e interação. As áreas de pesquisa em inteligência de enxames foram organizadas em [15], levando em consideração os níveis de pesquisa relacionados. São elas:

- Inspiração biológica;
- Comunicação;
- Controle;
- Mapeamento e localização;
- Transporte e manipulação de objetos;
- Robôs reconfiguráveis;
- Coordenação de movimento;
- Aprendizado;
- Alocação de tarefas.

Devido a limites de espaço, serão apenas brevemente apresentadas algumas pesquisas nestas áreas. Os mecanismos mais importantes em enxame de robôs contidos nessas pesquisas são os de agregação, dispersão, forragem, auto-montagem, movimentação conectada, transporte cooperativo e formação de padrões. [14]

1. Inspiração biológica

Enxame de robôs e, o conceito relacionado, inteligência de enxames, estão fundamentados nos mecanismos descentralizados que permeiam os enxames naturais de abelhas, formigas, pássaros, peixes, lobos e até mesmo humanos. Os insetos sociais possuem os melhores exemplos de comportamento biológicos auto-organizável. Por meio de comunicação local e limitada, são capazes de realizar tarefas impressionantes como, por exemplo, manutenção da saúde da colônia, cuidado com os menores e reação a invasões [16]. Em [17], é apresentado um grupo de robôs com o objetivo de localização de alvos em um ambiente desconhecidos utilizando como inspiração biológica a quimiotaxia, que é o processo de locomoção em direção a um gradiente químico. Alguns tipos de bactérias como, por exemplo, *E. coli*, exibem esse comportamento através da

mudança da direção de rotação de seus flagelos. A quimiotaxia inspirou estratégias de busca com robôs efetivamente no passado [18].

A adoção do feromônio como inspiração biológica foi utilizada em [19] em um exame de robôs distribuídos sendo proposto um mecanismo de feromônio virtual como um esquema de coordenação da troca de mensagem entre os robôs.

2. Comunicação

Em qualquer tarefa que requeira cooperação, existirá uma necessidade de comunicação entre os agentes participantes. Embora existam debates em torno do nível de comunicação permitida para tais sistemas, a maior parte das referências faz distinção entre comunicação implícita/indireta e explícita/direta. Comunicação implícita, algumas vezes chamada de estigmergia (*stigmergy*), é um método de comunicação através do ambiente. A comunicação através da utilização do feromônio é um tipo de comunicação implícita [7] [19].

Comunicação explícita é o tipo de comunicação na qual os robôs trocam mensagens diretamente uns com os outros [7]. A comunicação entre os robôs pode multiplicar suas capacidades e aumentar sua eficiência. Embora não existam conclusões claras acerca de qual tipo de comunicação seja melhor para um exame de robôs, a maioria das pesquisas visam a comunicação indireta devido à sua robustez.

3. Controle

Existem duas formas de controle: centralizado e distribuído. O controle centralizado é caracterizado pela presença de um agente líder que está no comando da organização do trabalho entre os outros robôs. O líder está envolvido no processo de tomada de decisão para todo o grupo, enquanto o resto do grupo age de modo a concordar com as decisões tomadas pelo líder. O controle descentralizado, pelo contrário, é caracterizado pela completa autonomia presente em todos os agentes no processo de tomada de decisão respeitando uns aos outros. Nesse tipo de controle, não existe um líder.

4. Mapeamento e localização

Mapeamento é a representação do ambiente físico através das informações sensoriais do robô móvel em modelos espaciais. Localização é definido como encontrar a localização absoluta ou racional do robô nos modelos espaciais gerados. Um dos maiores problemas nesta área, enquanto obtendo um mapa do ambiente desconhecido através de um robô móvel, é localizar este mesmo robô

relativamente ao mapa obtido. Em [20] foi proposto, simulado e implantado com bons resultados, um algoritmo de exploração de mapas baseado em fronteiras (*frontier-based*) utilizando o PSO para coordenação entre os robôs. Um algoritmo utilizado na coordenação de robôs móveis para encontrar alvos desejados sem depender de uma informação global de posicionamento precisa foi proposto por [21]. O algoritmo inspira-se no PSO para guiar os robôs através das missões de busca e é baseado em geometria cartesiana para unificar um sistema de coordenadas relativas.

5. Transporte e manipulação de objetos

Habilitar os robôs a transportar e manipular objetos cooperativamente foi uma das primeiras metas estabelecidas. No entanto, esta se mostra muito difícil de se alcançar pois, embora muitos projetos lidaram com aspectos como esses, poucos tiveram resultados plausíveis em robôs reais. Aspectos relevantes nesta área estão relacionados com a capacidade que os robôs tem de realizar a movimentação com ou sem restrições, movimentação conectada (isto é, locomoção com os robôs conectados uns aos outros), modelos baseados em sistemas globais versus sistemas distribuídos, dupla de robôs versus enxame de robôs, etc.

As pesquisas nesta área classificaram em três tipos os métodos para manipulação de objetos: *grasping* (agarrar), *pushing* (empurrar) e *caging* (prender). A técnica de *grasping* incorpora técnicas de *form closure* e *force closure*. Força de fechamento é uma condição que implica que o agarramento é capaz de resistir a qualquer força externa aplicado ao objeto. A técnica de *pushing*, por sua vez, não garante *form closure* ou *force closure*, mas necessita que forças externas, tais como peso e atrito, sejam aplicadas ao objeto. A técnica de *pushing* possui as vantagens de não necessitar agarrar o objeto e mover múltiplos objetos ao mesmo tempo. A técnica de *caging* introduz um área móvel limitada (ou de fronteira), tornando o planejamento de locomoção e controle simplificado e robusto. Essa forma é chamada de *object closure* (fechamento do objeto). Em [22] foi utilizada a técnica de *object closure* para locomover objetos utilizando a cooperação entre robôs móveis e em [23] é apresentado um grupo de robôs móveis transportando cooperativamente um objeto pesado utilizando a técnica de *grasping*.

6. Robôs reconfiguráveis

As pesquisas nesta área visam capacitar os robôs à alterarem sua forma através

do rearranjo da conectividade de suas partes, com o objetivo de adaptar-se à novas circunstâncias, desempenhar novas tarefas ou recuperar-se de alguma avaria. Em [24] é apresentado, simulado e implantado, a habilidade de auto-montagem em um enxame de robôs.

7. Coordenação de movimento

Dentro desta área, planejamento de trajetória é o tema com o maior número de pesquisas. Vários algoritmos foram desenvolvidos para enfrentar os problemas de planejamento de trajetória como, por exemplo, lógica difusa [25], PSO. [26]

Existe, também, o problema de geração de formação, que é definido como a coordenação de um grupo de robôs para entrar e manter uma formação com determinada configuração como, por exemplo, uma linha, uma circunferência, etc. [27]

8. Aprendizado

O aprendizado pode ser supervisionado ou não supervisionado. O aprendizado supervisionado requer a utilização de um supervisor externo. No aprendizado supervisionado, os robôs tem o conhecimento de quais são as melhores saídas em uma dada situação através da correção fornecida pelo supervisor, utilizando, por exemplo, árvores de decisão, redes neurais artificiais, etc. O aprendizado não supervisionado é um método que requer um mínimo, ou mesmo nenhum, *feedback* corretivo do ambiente. Para tal forma de aprendizado, pode-se utilizar, por exemplo, computação evolucionária, algoritmos genéticos, etc. Esse tipo de método é útil por permitir que os robôs se adaptem à situações onde o ambiente ou as tarefas não são conhecidos ou estão em constante mudança. Em [28] é apresentada uma aplicação utilizando redes neurais.

9. Alocação de tarefas

Consiste na atribuição de tarefas entre os robôs de maneira produtiva e eficiente. A alocação realizada não deve assegurar apenas o cumprimento da missão global, mas também que as tarefas sejam bem distribuídas entre os robôs. Uma abordagem efetiva na alocação de tarefas deve considerar os recursos disponíveis, os atributos que serão otimizados (tempo, energia, etc), as capacidades dos robôs, etc. O problema de alocação de tarefas multi-robôs tem sido investigado utilizando diferentes técnicas como, por exemplo, planejamento distribuído [29].

3 *Descrição dos componentes do sistema*

*“Atingimos a perfeição não quando nada
pode acrescentar-se a um projeto,
mas quando nada pode retirar-se.”*

– Saint-Exupery

Neste capítulo serão apresentados o modelo do ambiente e o modelo de agentes e seus mecanismos, a saber, o mecanismo de localização, o mecanismo de locomoção, o mecanismo anti-colisão, o mecanismo de comunicação e o mecanismo de percepção do ambiente.

3.1 **Modelo do ambiente**

O mapeamento de ambientes é uma atividade de alta importância entre as atividades de um robô. Sua capacidade de exploração ambiental, geralmente, determina o sucesso de seus objetivos. Como mencionado anteriormente em 2.2, os robôs podem, ou não, possuir uma representação interna do ambiente. Todo UAV apenas possui conhecimento a respeito de sua localização atual e não possui representação interna do ambiente em que se encontra neste simulador. Dessa forma, o modelo utilizado pode ser considerado como um modelo de robô reativo, ou seja, o UAV não raciocina sobre a representação interna do ambiente para planejar trajetórias ou definir suas ações, mas apenas sobre informações adquiridas de seus sensores de percepção, comunicação, obstáculos, etc.

O ambiente virtual é determinado pelos seguintes parâmetros:

- Área de extensão ($A_{ambiente}$)

Esse parâmetro determina em metros quadrados, a área do ambiente virtual, cujos lados possuem tamanho $\sqrt{A_{ambiente}}$.

- Quantidade de UAVs (Q_{uavs})
 Esse parâmetro determina o número de UAVs que serão colocados no ambiente em posições aleatórias.
- Quantidade de alvos (Q_{alvos})
 Esse parâmetro determina o número de alvos que serão colocados no ambiente em posições aleatórias.
- Quantidade de obstáculos ($Q_{obstaculos}$)
 Esse parâmetro determina o número de obstáculos que serão colocados no ambiente em posições aleatórias.

O ambiente virtual consiste em um ambiente de 2 (duas) dimensões. As dimensões usadas nos eixos são parâmetros do simulador. Foi utilizado uma imagem que representa a fronteira do Brasil com o Paraguai apenas como ilustração. Tal fronteira estende-se desde Foz do Iguaçu, no Paraná, até Corumbá, em Mato Grosso do Sul, num total de 1.365,4 Km. O modelo do ambiente virtual é visualizado em 600x600 pixels independente de seu tamanho. A escala visual do ambiente, nos dois eixos, é $1:\sqrt{A_{ambiente}}$. A escala visual dos outros elementos como, alvos, obstáculos, UAVs, base terrestre, por questões de visualização, não são alteradas com a modificação da área ($A_{ambiente}$) e pode apresentar, visualmente, incoerência de dimensões.

A figura 3.1 ilustra o ambiente virtual.



Figura 3.1: Ambiente virtual: Fronteira do Brasil com o Paraguai.

Tal ambiente possui:

1. UAVs

O UAVs são os agentes principais do simulador, pois é a coordenação entre os UAVs que permitirá a realização da patrulha e rastreamento de forma eficiente. A figura 3.2 ilustra um UAV no ambiente.



Figura 3.2: UAV: Veículo Aéreo Não Tripulado.

2. Alvos

É considerado um alvo toda entidade que trafega através do ambiente sem permissão. Esse alvos são percebidos pelo sensor de percepção ambiental do UAV e representam invasões que devem ser comunicadas aos UAVs vizinhos e rastreadas durante toda sua permanência no ambiente. A figura 3.3 ilustra um alvo no ambiente.



Figura 3.3: Alvo: Um invasor no ambiente.

3. Obstáculos

É considerado um obstáculo qualquer estrutura fixa no ambiente como, por exemplo, uma antena de comunicação, que esteja nas proximidades do UAV. Esses obstáculos são percebidos pelo sensor anti-colisão do UAV e devem ser contornados. A figura 3.4 ilustra um obstáculo no ambiente.



Figura 3.4: Obstáculo: Qualquer estrutura fixa no ambiente.

4. Base terrestre

A base terrestre está presente no ambiente para dar suporte quando possível, devido as restrições de comunicação, e necessário. Todo UAV possui o conhecimento da localização, \vec{x}_{base} , de sua base terrestre. Tal informação é utilizada para retornar à base quando se fizer necessário. O UAV deseja retornar à base quando, por exemplo, invadir o espaço aéreo vizinho e perder a comunicação com seus vizinhos. A figura 3.5 ilustra a base terrestre dos UAVs.



Figura 3.5: Base terrestre.

5. Limites

Por fim, o modelo do ambiente possui limites nos sentidos norte, sul, leste e oeste. Tal propriedade é importante para configurar uma saída do UAV de seu espaço aéreo e consequente invasão do espaço aéreo vizinho.

3.2 Modelo do agente

O simulador foi construído utilizando o modelo de agente. Um agente é uma entidade que percebe o ambiente a partir de sensores, processa e raciocina sobre as informações obtidas desses mesmos sensores e conhecimentos *a priori* e atua no ambiente. Os UAVs foram modelados no simulador como agentes com suas respectivas propriedades.

Para realizar as atividades de patrulha, os UAVs devem possuir algumas habilidades de vital importância. Primeiramente, eles devem possuir a habilidade de locomoção dentro do ambiente e possuir algum conhecimento de sua localização. Geralmente, os UAVs apresentam limitações de locomoção como, por exemplo, o ângulo de rotação, velocidade máxima, aceleração máxima, etc. Os mecanismos de localização e locomoção estão descritos nas seções 3.2.1 e 3.2.2, respectivamente.

No entanto, tal locomoção deve ser realizada de maneira segura. O fator segu-

rança pode ser entendido como evitar colisões durante suas operações. O mecanismo anti-colisão está descrito na seção 3.2.3.

Para realizar a patrulha e rastreamento de maneira efetiva, os UAVs precisam comunicar-se uns com os outros. O mecanismo de comunicação é descrito na seção 3.2.4.

São propostos em [30] os seguintes critérios para distinguir uma arquitetura de um enxame de robôs:

- Quantidade de indivíduos relativamente alta, dependendo da aplicação;
- Homogeneidade entre os indivíduos;
- Cada indivíduo possui autonomia;
- Indivíduos simples, ou seja, relativamente incapazes ou ineficientes para realizar suas tarefas sozinhos;
- Indivíduos possuem sensoriamento local e capacidade de comunicação.

A partir desses critérios, foi elaborado um modelo para os UAVs desse simulador e as seções a seguir descrevem os principais mecanismos de tal modelo considerado neste simulador.

3.2.1 Mecanismo de localização

Como mencionado anteriormente, umas das principais atividades dos robôs móveis e, conseqüentemente, dos UAVs, é a exploração do ambiente em que se encontram. No caso específico dos UAVs, os mesmos possuem o objetivo de patrulhar o ambiente em que se encontram e rastrear eventuais alvos. Para tal, os UAVs não armazenam internamente informações a respeito do ambiente como, por exemplo, um mapa. No entanto, os UAVs tem o conhecimento de sua localização no ambiente, \vec{x}_{uav} . Tal informação possibilita o UAV locomover-se no ambiente, tomar decisões, retornar à base, evitar colisões e sair de seu espaço aéreo, rastrear possíveis alvos e atuar de maneira a manter-se conectado aos seus vizinhos.

É importante observar que nenhum UAV possui conhecimento *a priori* do ambiente. A conseqüência dessa característica é a adaptabilidade dos UAVs a qualquer ambiente em que estão inseridos.

Para implementação em UAVs reais, tal premissa pode ser alcançada a partir da utilização de um módulo GPS em cada UAV.

3.2.2 Mecanismo de locomoção

Embora o mecanismo de locomoção dos UAVs esteja relacionado com aerodinâmica, serão abstraídos diversos fatores ligados com aerodinâmica, no entanto, podem ser encontrados maiores detalhes em [31].

Não serão considerados os seguintes itens:

- Física da força de sustentação e seus respectivos princípios físicos como a terceira lei de Newton e o princípio de Bernulli, pressões dinâmicas de escoamento de fluido;
- Número de Reynolds, utilizado em mecânica dos fluidos para cálculo regime de escoamento de determinado fluido sobre uma superfície;
- Teoria do perfil aerodinâmico, ângulo de ataque, ângulo de incidência, etc.;
- Forças aerodinâmicas e momentos em perfis, bem como a capacidade que um perfil aerodinâmico possui de gerar essas forças;
- Centro de pressão e centro aerodinâmico do perfil;
- Formas geométricas e localizações dos componentes do UAV;
- Arrasto em aeronaves;

Serão considerados os seguintes atributos para dinâmica dos UAVs:

- Aceleração horizontal (a), expressa em metros/unidade de tempo ao quadrado;
- Aceleração horizontal máxima (a_{max}), expressa em metros/unidade de tempo ao quadrado;
- Velocidade horizontal (v), expressa em metros/unidade de tempo;
- Velocidade máxima, (v_{max}), expressa em metros/unidade de tempo;
- Direção, expressa em graus. Em UAVs reais pode-se utilizar uma bússola.

Em suas operações, os UAVs apresentam:

- Sincronismo;
 - Atuando de forma a afastar-se se sua distância ultrapassar um limite mínimo de aproximação (ver 3.2.3);
 - Atuando de forma a se aproximar se sua distância ultrapassar um limite máximo de afastamento (ver 3.2.4).
- Cognição;
 - De maneira estocástica quando em estado de Patrulha;
 - Perseguindo e comunicando as posições dos alvos quando em estado de Rastreamento.
- Social;

Considerando o direcionamento para o qual o grupo como um todo está se movendo.

Pode-se definir o vetor de sincronismo como o vetor calculado pela equação (3.1)

$$\vec{a}_{sincronismo} = \vec{a}_{antiColisao} + \vec{a}_{comunicacao} \quad (3.1)$$

Pode-se definir o vetor do enxame como o vetor calculado pela equação 3.2

$$\vec{a}_{enxame} = \vec{a}_{cognitivo} + \vec{a}_{social} \quad (3.2)$$

A aceleração resultante pode ser entendida de acordo com a equação (3.3).

$$a(t+1) = \vec{a}_{sincronismo} + \vec{a}_{enxame} \quad (3.3)$$

O vetor $\vec{a}(t+1)$ deve respeitar a inequação (3.4)

$$|\vec{a}(t+1)| \leq a_{max} \quad (3.4)$$

A velocidade resultante pode ser entendida de acordo com a equação (3.5)

$$\vec{v}(t+1) = \vec{v}(t) * \omega + \vec{a}(t+1) \quad (3.5)$$

onde:

- $\vec{v}(t + 1)$ – nova velocidade;
- $\vec{v}(t)$ – velocidade anterior;
- ω – fator de inércia;
- $\vec{a}_{antiColisao}$ – calculado em função do sensor anti-colisão descrito em 3.2.3;
- $\vec{a}_{comunicacao}$ – calculado em função do do sensor de comunicação descrito em 3.2.4;
- $\vec{a}_{cognitivo}$ – calculado em função do sensor de percepção do ambiente do UAV;
- \vec{a}_{social} – calculado em função do direcionamento dos vizinhos para o melhor local.

O vetor $\vec{v}(t + 1)$ deve respeitar a inequação 3.6

$$|\vec{v}(t + 1)| \leq v_{max} \quad (3.6)$$

Além desses direcionamentos, podem surgir, eventualmente, outros dois. São eles:

- $\vec{a}_{limiteCruzado}$ – surge quando o UAV está prestes a sair de seus espaço aéreo e invadir o espaço aéreo vizinho;
- \vec{a}_{base} – calculado em função da localização da base e surge quando o UAV invade o espaço aéreo vizinho, se perde, ou qualquer outra situação adversa ocorre.

Uma vez definida o vetor velocidade, $\vec{v}(t + 1)$, a posição do UAV é atualizada conforme a equação (3.7).

$$\vec{x}(t + 1) = \vec{x}(t) + \vec{v}(t + 1) \quad (3.7)$$

Como a unidade de iteração é unitária, o fator tempo é omitido nas equações.

Dependendo da intensidade de cada direcionamento, a ordem de prioridade é estabelecida de acordo com a Equação (3.8).

$$prioridade(\vec{a}_{antiColisao}) > prioridade(\vec{a}_{limiteCruzado}) > prioridade(\vec{a}_{comunicacao}) \quad (3.8)$$

Sendo $prioridade(\vec{a})$ um função que determina a prioridade de um direcionamento \vec{a} e um maior valor configura uma maior prioridade.

Quando o sensor anti-colisão fornece um direcionamento utilizando como intensidade o valor da velocidade máxima, v_{max} , provavelmente, uma colisão está prestes a ocorrer. Dessa forma, todos os outros direcionamentos são desconsiderados e o UAV atua considerando apenas $\vec{a}_{antiColisao}$.

Os raios de atuação dos três mecanismos, descritos adiante, podem ser visualizados no simulador de acordo com a figura 3.6. A cor vermelha representa o sensor anti-colisão, a verde representa o sensor de comunicação e, por fim, a cor azul representa o sensor de percepção do ambiente. Os valores de cada raio, como apresentado no Capítulo 5, são parâmetros do simulador. No exemplo da figura 3.6, foram utilizados os valores 1500m para o sensor anti-colisão, 1000m para o sensor de comunicação e 500m para o sensor de percepção.

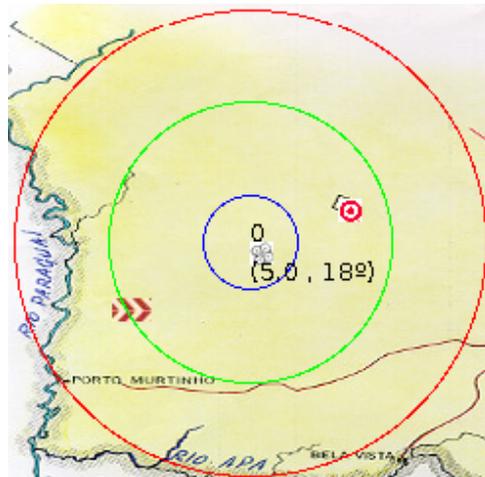


Figura 3.6: Raio de atuação dos sensores do UAV.

3.2.3 Mecanismo anti-colisão

O UAV possui um sensor anti-colisão para possibilitar uma locomoção segura através do ambiente. Obstáculos podem ser entendidos como, por exemplo, estruturas fixas e outros UAVs.

O sensor anti-colisão possui os seguintes parâmetros:

1. Raio de alcance ($r_{antiColisao}$);

Esse parâmetro determina o raio de alcance do sensor anti-colisão expresso em metros. A distância entre qualquer obstáculo e o UAV é calculada pelo sensor

anti-colisão de acordo com a equação

$$d_{antiColisao} = \|\vec{x}_{uav} - \vec{x}_{obstaculo}\| \quad (3.9)$$

O sensor anti-colisão é sensibilizado quando a inequação (3.10) é satisfeita.

$$d_{antiColisao} \leq r_{antiColisao} \quad (3.10)$$

onde \vec{x}_{uav} é a posição atual do UAV obtida através do mecanismo de localização descrito em 3.2.1 e $\vec{x}_{obstaculo}$ é a posição de qualquer obstáculo presente no ambiente.

2. Limiar ($l_{colisao}$);

Esse parâmetro determina o início da atuação efetiva do sensor anti-colisão. Embora o sensor de colisão possua um raio de atuação $r_{antiColisao}$, pode ser desejado iniciar a atuação a partir de $l_{colisao}$ metros, onde

$$l_{antiColisao} \leq r_{antiColisao} \quad (3.11)$$

3. Segurança ($s_{antiColisao}$);

Determina uma distância mínima de segurança na qual é desejado atuar com intensidade igual a a_{max} e

$$s_{antiColisao} \leq l_{antiColisao} \quad (3.12)$$

A intensidade de atuação do sensor anti-colisão é calculada de acordo com as equações (3.13), (3.14), (3.15), (3.16) e (3.17). A intensidade de atuação pode ser verificada no gráfico presente na figura 3.7.

$$a_{antiColisao} = A * d_{antiColisao} + B \quad (3.13)$$

no intervalo ($s_{antiColisao}, l_{antiColisao}$), onde

$$A = \frac{a_{max}}{s_{antiColisao} - l_{antiColisao}} \quad (3.14)$$

e

$$B = -A * l_{antiColisao} \quad (3.15)$$

$$a_{antiColisao} = a_{max} \quad (3.16)$$

no intervalo $[0, s_{antiColisao}]$ e

$$a_{antiColisao} = 0 \quad (3.17)$$

no intervalo $[l_{antiColisao}, +\infty]$.

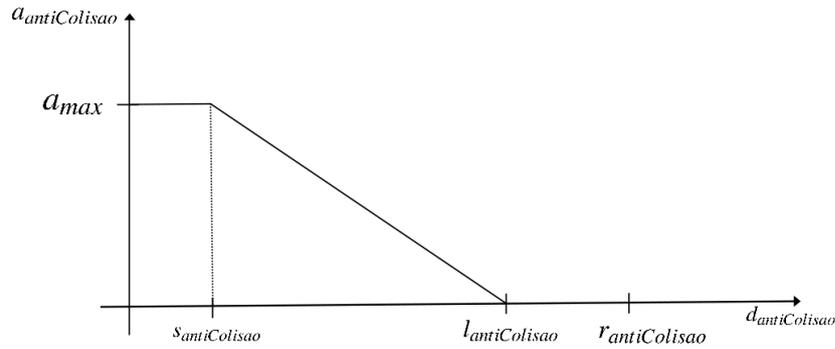


Figura 3.7: Cálculo da intensidade de atuação do sensor anti-colisão.

Dado que $n_{obstaculos}$ é o número de obstáculos no raio do sensor anti-colisão, a direção e o sentido da atuação do sensor anti-colisão é a mesma do vetor resultante da equação (3.18)

$$\vec{a}_{antiColisao} = \sum_{i=1}^{n_{obstaculos}} \vec{x}_{uav} - \vec{x}_{obstaculoi} \quad (3.18)$$

Devido a sua simplicidade e menor consumo de processamento, foi utilizada uma função linear como modelo para calcular a intensidade de atuação do sensor anti-colisão no intervalo de $(s_{antiColisao}, l_{antiColisao})$, outras funções como, por exemplo, a sigmoidal logística, tangente hiperbólica, ou aproximações destas, podem ser utilizadas. Dado que $e_{obstaculo}$ é o raio da extensão do obstáculo e e_{uav} é o raio da extensão do UAV considerando uma modelagem circular, uma colisão fica estabelecida quando a inequação (3.19) é satisfeita.

$$d_{antiColisao} \leq e_{obstaculo} + e_{uav} \quad (3.19)$$

3.2.4 Mecanismos de comunicação

A comunicação utilizada em robôs móveis, geralmente, é sem fio devido, principalmente, à característica dinâmica do ambiente. Embora exista UAV que utiliza um sistema de comunicação satélite [32], neste simulador foi proposto um sistema de comunicação sem fio com uma topologia *ad hoc*, ou seja, cada UAV é ponte de roteamento, baseado no padrão IEEE 802.15.4 [33]. É importante observar que o sensor de comunicação sensibiliza os atuadores do UAV com o objetivo de evitar a perda de comunicação que ocorre quando a inequação (3.32) é satisfeita.

Todo UAV possui um dispositivo de comunicação sem fio com os seguintes parâmetros:

1. Raio de alcance ($r_{comunicacao}$);

Esse parâmetro determina o raio de alcance de comunicação expresso em metros. A distância entre qualquer vizinho e o UAV é calculada pelo sensor de comunicação de acordo com a equação

$$d_{comunicacao} = ||\vec{x}_{uav} - \vec{x}_{vizinho}|| \quad (3.20)$$

Um UAV está no raio de comunicação quando a inequação (3.21),

$$d_{comunicacao} \leq r_{comunicacao} \quad (3.21)$$

é satisfeita, onde \vec{x}_{uav} é a posição atual do UAV obtida através do mecanismo de localização descrito em 3.2.1 e $\vec{x}_{vizinho}$ é a posição de qualquer vizinho no raio de comunicação.

2. Número máximo de vizinhos ($n_{vizinhosmax}$)

Esse parâmetro determina o número máximo de vizinhos que um UAV possui. É importante observar que para garantir o roteamento

$$n_{vizinhosmax} > 1 \quad (3.22)$$

Um UAV é considerado n-conectado, quando

$$n_{vizinhos} = n_{vizinhosmax} \quad (3.23)$$

dado que $n_{vizinhos}$ é o número de vizinhos conectados no momento.

3. Limiar ($l_{comunicacao}$);

Esse parâmetro determina o início da atuação efetiva do sensor de comunicação. Embora o sensor de comunicação possua um raio de atuação $r_{comunicacao}$, pode ser desejado iniciar a atuação com o objetivo de manter a conexão a partir de $l_{comunicacao}$ metros, onde

$$l_{comunicacao} \leq r_{comunicacao} \quad (3.24)$$

4. Segurança ($s_{comunicacao}$);

Determina uma distância mínima de segurança na qual é desejado atuar com intensidade igual a a_{max} e

$$s_{comunicacao} \geq l_{comunicacao} \quad (3.25)$$

A intensidade de atuação do sensor de comunicação é calculada de acordo com as equações 3.26, 3.27, 3.28, 3.29 e 3.30. A intensidade de atuação pode ser verificada no gráfico presente na figura 3.8.

$$a_{comunicacao} = A * d_{comunicacao} + B \quad (3.26)$$

no intervalo ($l_{comunicacao}, s_{comunicacao}$), onde

$$A = \frac{a_{max}}{l_{comunicacao} - s_{comunicacao}} \quad (3.27)$$

e

$$B = -A * s_{comunicacao} \quad (3.28)$$

$$a_{comunicacao} = a_{max} \quad (3.29)$$

no intervalo [$s_{comunicacao}, r_{comunicacao}$] e

$$a_{comunicacao} = 0 \quad (3.30)$$

no intervalo $[-\infty, l_{comunicacao}]$.

A direção e o sentido da atuação do sensor de comunicação é a mesma do vetor resultante da equação 3.31

$$\vec{a}_{comunicacao} = \sum_{i=1}^{n_{vizinhos}} \vec{x}_{vizinhoi} - \vec{x}_{uav} \quad (3.31)$$

Devido a sua simplicidade e menor consumo de processamento, foi utilizada uma

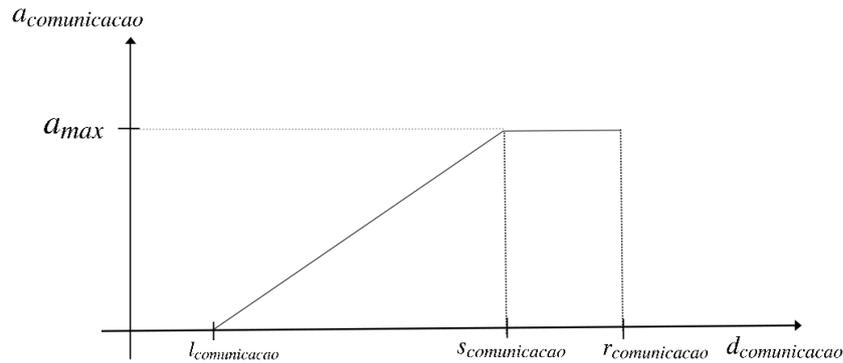


Figura 3.8: Cálculo da intensidade de atuação do sensor de comunicação.

função linear como modelo para calcular a intensidade de atuação do sensor de comunicação no intervalo de $(l_{comunicacao}, s_{comunicacao})$, outras funções como, por exemplo, a sigmoideal logística, tangente hiperbólica, ou aproximações destas, podem ser utilizadas.

Uma conexão é considerada perdida e, conseqüentemente, o UAV vizinho é excluído do conjunto de vizinhos, quando

$$d_{comunicacao} > r_{comunicacao} \quad (3.32)$$

3.2.5 Mecanismo de percepção do ambiente

Várias técnicas podem ser utilizadas na percepção ambiental como, por exemplo, a utilização de RADAR ou processamento digital de imagens, embora essa última técnica necessite de uma maior poder de processamento.

O sensor de percepção ambiental é determinado pelos seguintes parâmetros:

- Raio de percepção ($r_{percepcao}$)

Esse parâmetro especifica o alcance do sensor de percepção do ambiente e determina a visibilidade de patrulha e rastreamento dos alvos. Um alvo é detectado quando a inequação (3.33) é satisfeita.

$$d_{alvo} \leq r_{percepcao} \quad (3.33)$$

dado que

$$d_{alvo} = |\vec{x}_{uav} - x_{alvo}| \quad (3.34)$$

- $l_{percepcao}$

Esse parâmetro determina a distância mínima que se deseja chegar do alvo em metros. A presença desse limiar evita colisões nas proximidades do alvo, pois é utilizado no rastreamento, quando a equação (3.35) é satisfeita, para evitar que o UAV atue no sentido do alvo quando já se encontrar nas proximidades do mesmo.

$$d_{alvo} \leq l_{percepcao} \quad (3.35)$$

Para o cálculo da equação (3.2), presente no Algoritmo 1, é necessária a avaliação de uma função chamada *fitness*. Uma das funções *fitness* mais simples é a distância euclidiana. Para o problema em questão, é desejado calcular a distância euclidiana entre o UAV e o alvo detectado. Tal cálculo é dado pela equação (3.36)

$$fitness_{uav}(t) = d_{alvo}(t) = \sqrt{[x_{uav}(t) - x_{alvo}(t)]^2 + [y_{uav}(t) - y_{alvo}(t)]^2} \quad (3.36)$$

onde as informações a respeito do posicionamento do alvo são fornecidas pelo sensor de percepção do ambiente.

É importante observar que a função $fitness_{uav}(t)$ não é, necessariamente, invariante no tempo, como ocorre, geralmente, nas utilizações clássicas do PSO. Dado que o ambiente de um problema dinâmico pode sofrer alterações de diversas formas [34], o ambiente em questão, por exemplo, em relação ao sensor de percepção, já se configura como um ambiente dinâmico, visto que o ótimo da função é a distância entre o UAV e os alvos e que esses mesmos alvos possuem a capacidade de locomoção. Dessa forma, é muito provável que $fitness(t)$ não seja invariante no tempo para todo t , salvo nas situações em que o alvo possuir velocidade nula ($|\vec{v}_{alvo}| = 0$).

No ambiente em questão, as alterações que ocorrem são mudanças não dimensionais [35], onde tanto a posição quanto o valor do ótimo local sofrem modificações com o tempo [36].

Quando em rastreamento, dado que o ambiente é dinâmico, os UAVs conectados devem possuir as seguintes habilidades :

- Habilidade de detectar a mudança do ótimo e encontrar sua nova posição [37]:

essa habilidade é alcançada através do compartilhamento de informação entre os vizinhos;

- Habilidade de seguir o ótimo do problema [38]: essa habilidade é alcançada utilizando a Equação (3.3).

Wicker [39] propõe três objetivos principais para os algoritmos de otimização de problemas dinâmicos. São eles:

- Precisão: mede o quão próxima a solução encontrada está da solução global. Para o problema da coordenação entre os UAVs, a solução é local aos UAVs conectados de forma *ad hoc*;
- Estabilidade: um algoritmo é considerado estável se mudanças no ambiente não impactam seriamente na sua precisão;
- Reação: mede a capacidade de reação do algoritmo às mudanças que ocorrem no ambiente.

As três propriedades acima puderam ser verificadas nas simulações realizadas e apresentadas no Capítulo 6, tanto de forma visual, como apresentado na seção 6.1, quanto de forma numérica e estatística, como apresentado na seção 6.2.

3.2.6 Outros mecanismos

Existem outros dois mecanismos que compõe o modelo dos UAVs, a saber, o mecanismo responsável por garantir que o UAV retorne à base e o mecanismo responsável por não permitir a invasão do espaço aéreo vizinho por parte do UAV.

O primeiro mecanismo atua com o objetivo de locomover o UAV na direção e sentido da base, dado pela equação (3.37).

$$\vec{a}_{base} = \vec{x}_{base} - \vec{x}_{uav} \quad (3.37)$$

Um UAV entende que chegou na base quando a inequação (3.38) é satisfeita.

$$|\vec{x}_{base} - \vec{x}_{uav}| < l_{percepcao} \quad (3.38)$$

E o segundo mecanismo é utilizado para evitar que o UAV ultrapasse seu espaço aéreo. Para tal, é gerada uma repulsão $\vec{a}_{limiteCruzado}$.

Embora pudesse ter sido utilizada, para tal mecanismo, uma modelagem circular em relação à base, para não criar o direcionamento de vários UAVs para uma mesma posição, foi utilizada outra abordagem.

O UAV entende a sua direção e sentido atuais como sendo os responsáveis por estarem o colocando para fora de seu espaço aéreo. Sendo assim, a aceleração gerada para impedir que o UAV saia de seu espaço aéreo é dada pela equação (3.39) utilizando coordenadas polares.

$$\vec{a}_{limiteCruzado} = (a_{limiteCruzado}, \theta_{\vec{v}(t)} + 180^\circ) \quad (3.39)$$

onde $\theta_{\vec{v}(t)}$ é o ângulo, em coordenadas polares, da velocidade atual ($\vec{v}(t)$) do UAV.

A intensidade $a_{limiteCruzado}$ é calculada de forma semelhante às equações (3.13), (3.14), (3.15), (3.16) e (3.17). No entanto, são utilizados os parâmetros $l_{limiteCruzado}$ e $S_{limiteCruzado}$.

4 *Descrição do comportamento do sistema*

“...quem despreza o que se faz e se atém ao que deveria ser feito aprenderá a maneira de se arruinar, e não a defender-se.”

– Maquiavel

Neste capítulo serão abordados os comportamentos do simulador. A seção 4.1 apresenta detalhes dos estados nos quais o UAV pode se encontrar. A seção 4.2 apresenta a estratégia de sincronização dos UAVs para evitar colisão e perda de comunicação durante todas as operações. A seção 4.3 apresenta a estratégia de agrupamento. A seção 4.4 apresenta a estratégia de patrulha. A seção 4.5 apresenta a estratégia de rastreamento utilizando inteligência de enxame. Por fim, a seção 4.7 apresenta a rotina de coordenação distribuída executada por cada UAV.

4.1 **Máquina de estados do UAV**

O UAV pode assumir quatro estados durante toda a missão:

1. Estado de agrupamento (*GroupState*);
2. Estado de patrulha (*PatrolState*);
3. Estado de rastreamento (*TrackingState*);
4. Estado de retorno à base (*BaseReturnState*).

Todos os UAVs iniciam suas operações no estado de agrupamento e encerram quando a missão é finalizada ou quando se envolvem em colisões. Neste simulador, o fim da missão é estabelecido pelo número máximo de iterações.

A condição de guarda da transição do estado de agrupamento para o estado de patrulha é a restrição imposta pela equação (3.23). Ao entrar no estado de rastreamento, o UAV passa a comunicar a presença do alvo e, enquanto houver alvo detectado, o mesmo é comunicado aos UAVs vizinhos.

É possível visualizar os estados, ações, e condições de guarda do UAV na figura 4.1.

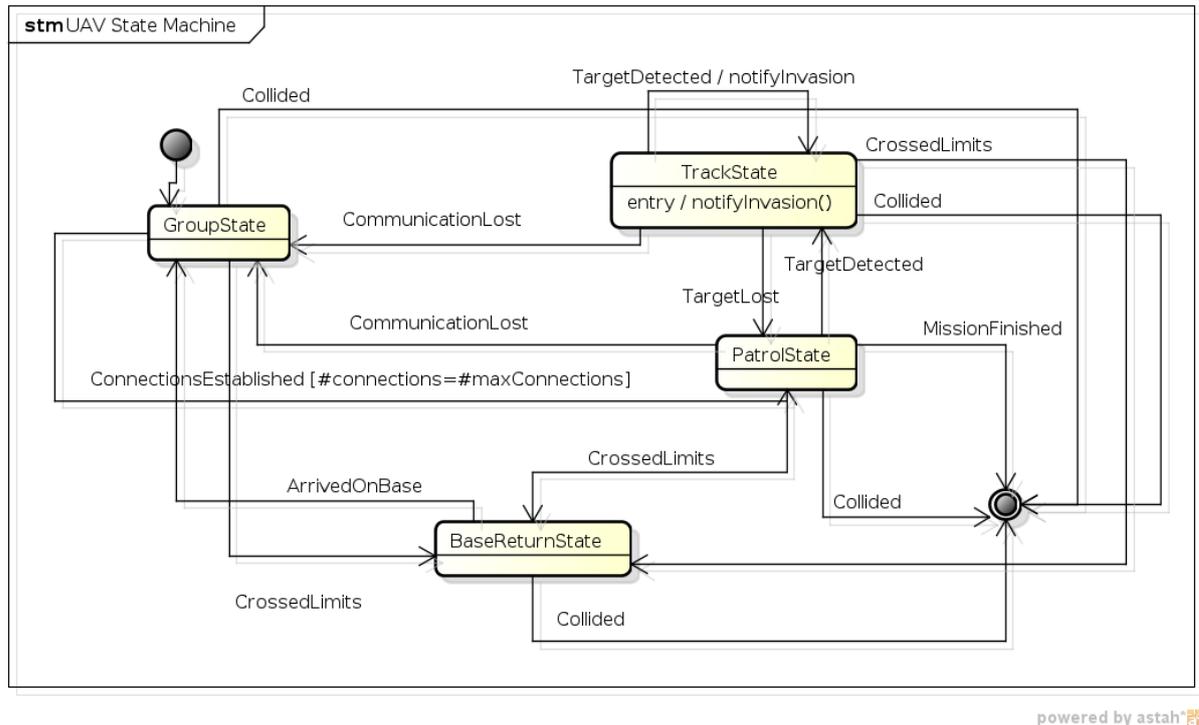


Figura 4.1: Máquina de estados do UAV.

4.2 Estratégia de sincronismo

A locomoção dos UAVs é realizada de forma sincronizada, no que diz respeito a evitar colisões, perda de comunicações e invasão do espaço aéreo vizinho.

A estratégia de sincronismo é executada em todos os estados do UAV e possui três objetivos:

- Afastar-se de obstáculos, inclusive UAVs, de forma à evitar colisões;
- Aproximar-se de UAVs conectados, de forma à evitar perda de conexões;
- Atuar de forma à evitar sair do espaço aéreo.

Além dos três objetivos acima mencionados, existe a priorização destes de acordo com a Equação (3.8).

A estratégia de sincronismo é apresentada no Algoritmo 2

Algorithm 2: Estratégia de sincronismo.

- 1 Calcular o vetor anti-colisão $\vec{a}_{antiColiso}$ (Equação (3.18))
 - 2 Calcular o vetor de comunicação $\vec{a}_{comunicacao}$ (Equação (3.31))
 - 3 Calcular o vetor de percepção dos limites do ambiente $\vec{a}_{limiteCruzado}$
 - 4 **se** $|\vec{a}_{antiColiso}| = a_{max}$ **então**
 - 5 Retornar $\vec{a}_{antiColiso}$
 - 6 **se** $|\vec{a}_{limiteCruzado}| = a_{max}$ **então**
 - 7 Retornar $\vec{a}_{limiteCruzado}$
 - 8 **se** $|\vec{a}_{comunicacao}| = a_{max}$ **então**
 - 9 Retornar $\vec{a}_{comunicacao}$
 - 10 Retornar $\vec{a}_{antiColiso} + \vec{a}_{comunicacao} + \vec{a}_{limiteCruzado}$
-

4.3 Estratégia de agrupamento

O objetivo dos UAVs neste simulador é patrulhar o ambiente e rastrear possíveis alvos. No entanto, antes de iniciar as atividades de patrulha e rastreamento, os UAVs devem encontrar, estabelecer e manter comunicações com outros $n_{vizinhomax}$ UAVs. Essas atividades iniciais são chamadas de atividades de agrupamento.

Como os UAVs são inicializados no ambiente em posições e com velocidades aleatórias, para realizar o agrupamento, o UAV realiza sua locomoção, quando possível, sempre em frente.

A estratégia de agrupamento é importante, pois possibilita a formação de redes de comunicação *ad hoc* em qualquer ambiente desconhecido, desde que os UAVs vizinhos implementem o mesmo padrão de comunicação.

A estratégia de agrupamento é apresentada no Algoritmo 3.

Neste simulador, foi adotada a priorização do agrupamento em relação ao rastreamento. Ou seja, quando um UAV perceber um alvo em suas proximidades e estiver no estado de agrupamento, o alvo será ignorado, pois o UAV só iniciará as atividades de rastreamento quando estiver conectado à outros $n_{vizinhomax}$ UAVs e estiver no estado de patrulha.

Algorithm 3: Estratégia de agrupamento.

- 1 Calcular vetor de sincronismo $\vec{a}_{sincronismo}$ (Algoritmo 2)
 - 2 **se** $n_{vizinhos} = n_{vizinhos_{max}}$ (Equação 3.23) **então**
 - 3 └ Transicionar para o estado de patrulha (Algoritmo 4)
 - 4 **senão**
 - 5 └ Verificar a presença de UAVs (Equação (3.21))
 - 6 Calcular $\vec{a}_{agrupamento} = \vec{v}(t) * \frac{a_{max}}{|\vec{v}(t)|}$
 - 7 Retornar $\vec{a}_{sincronismo} + \vec{a}_{agrupamento}$
-

4.4 Estratégia de patrulha

Quando um UAV está conectado de acordo com a Inequação (3.23), seu estado é transicionado para o estado de patrulha.

Conforme mencionado em 3.2.1, os UAVs modelados nesse simulador não armazenam internamente representações do ambiente no qual se encontram para realizar planejamento de rotas, evitar patrulha redundante, etc. A estratégia de patrulha é locomover-se sempre em uma direção aleatória. Embora tal estratégia seja simples e em conformidade com o modelo de agentes reativos, pelos componentes aleatórios introduzidos, como posições e velocidades iniciais, é possível estabelecer coberturas territoriais satisfatórias conforme apresentado na seção 6.2 do Capítulo 6.

A estratégia de patrulha é apresentada no Algoritmo 4 e $\vec{a}_{patrulha}$ representa a estratégia de patrulha.

Algorithm 4: Estratégia de patrulha .

- 1 Calcular vetor de sincronismo $\vec{a}_{sincronismo}$ (Algoritmo 2)
 - 2 **se** $d_{alvo} \leq r_{percepcao}$ (Equação 3.33) **então**
 - 3 └ Transicionar para o estado de rastreamento
 - 4 Calcular $\vec{a}_{patrulha} = \vec{v}(t) * \frac{a_{max}}{|\vec{v}(t)|}$
 - 5 Retornar $\vec{a}_{sincronismo} + \vec{a}_{patrulha}$
-

4.5 Estratégia de rastreamento

A execução da estratégia de rastreamento tem início quando, de acordo com a Equação (3.33), um alvo é detectado. Nesse estado, o UAV que detectou e está rastreando algum alvo, comunica constantemente este alvo para seus $n_{vizinhos_{max}}$ vizinhos.

Estes também o comunicam e dessa forma o alvo é comunicado por toda a rede de maneira *ad hoc*.

O UAV neste simulador, prioriza o alvo mais próximo, visto que a probabilidade de perdê-lo é menor.

Mesmo que o UAV esteja presente em um rede onde está sendo comunicado o rastreamento de um alvo, mas este mesmo alvo está fora de seu alcance, conforme a Equação (3.33), não se faz necessário a execução da estratégia de rastreamento, pois a mesma é garantida por *a_sincronismo*. Ou seja, os UAVs atuam no sentido de ficar conectados ao enxame e, conseqüentemente, acompanhar o rastreamento de um possível alvo.

Contudo, a consequência da utilização desta estratégia é a formação de sub-enxames, como será apresentado na seção 6.1 do Capítulo 6. Tal formação tem origem quando um segundo alvo entra no raio de percepção de UAVs em estado de patrulha e localizados, geralmente, nas periferias do enxame formado pela rede de comunicações *ad hoc*. Estes detectam o alvo, iniciam seu rastreamento, se desligam do enxame original e formam entre si um novo enxame. Este segundo enxame é formado por UAVs que executam a estratégia de rastreamento para o segundo alvo e, geralmente, é menor que o enxame original.

Então, embora a estratégia utilizada reduza o número total de comunicações em todo sistema, devido a formação de sub-enxames, a utilização da mesma produz dois comportamentos desejados:

1. Aumento da cobertura territorial patrulhada;
2. Rastreamento de mais de um alvo paralelamente.

Todos os dois comportamentos acima mencionados e considerados inteligentes ocorrem com pouca cognição cooperativa, principalmente, quando comparada com a cognição presente, geralmente, nos agentes *Belief-Desire-Intention* (BDI) em sistemas multiagentes (SMA). Esses comportamentos inteligentes emergem da interação e compartilhamento de informação entre os UAVs: inteligência de enxames.

A estratégia de rastreamento é apresentada no Algoritmo 5.

Algorithm 5: Estratégia de rastreamento .

```

1 Calcular vetor de coordenação  $\vec{a}_{sincronismo}$  (Algoritmo 2)
2 se  $d_{alvo} \geq r_{percepcao}$  (Equação 3.33) então
3   | Transicionar para o estado de patrulha
4 senão
5   | Atualizar fitness
6   | Comunicar alvo aos vizinhos
7   | Calcular  $\vec{a}_{exame}$  (Equação 3.2)
8   se  $d_{alvo} \leq l_{percepcao}$  então
9     |  $\vec{a}_{exame} = \vec{0}$ 
10  | Retornar  $\vec{a}_{sincronismo} + \vec{a}_{exame}$  (Equação (3.3))

```

4.6 Estratégia de retorno à base

A última estratégia é executado quando o UAV se encontra em situações adversas. Neste simulador, a única situação adversa é quando o UAV ultrapasse seu espaço aéreo, conseqüentemente, invadindo o espaço aéreo vizinho.

A partir do conhecimento da localização de sua base terrestre, \vec{x}_{base} , o UAV atua para retornar à base.

A estratégia de retorno à base é apresentada no Algoritmo 6.

Algorithm 6: Estratégia de retorno à base .

```

1 Calcular vetor de coordenação  $\vec{a}_{sincronismo}$  (Algoritmo 2)
2 se  $|\vec{x}_{base} - \vec{x}_{uav}| < l_{percepcao}$  (Equação (3.38)) então
3   | Transicionar para o estado de agrupamento
4 senão
5   | Calcular vetor para retorno à base  $\vec{a}_{base}$  (Equação (3.37))
6 Retornar  $\vec{a}_{sincronismo} + \vec{a}_{base}$  (Equação (3.3))

```

Health Monitoring é o mecanismo que cuida da saúde do UAV. O termo saúde deve ser entendido como possíveis avarias, mal funcionamento de algum componente ou quantidade de energia abaixo do seguro para retornar à base. O mecanismo acima não foi implementado neste simulador por razões de escopo. No entanto, a estratégia de retorno à base em questão será útil quando utilizado tal mecanismo.

4.7 Rotina principal

A composição de todas as estratégias anteriormente mencionadas formam a rotina principal, que é uma rotina de coordenação distribuída de UAVs.

A rotina principal é apresentada no Algoritmo 7. Como consequência da utilização de máquina de estados para modelar o comportamento do UAV, a aceleração do UAV, $\vec{a}(t + 1)$, é calculada em função de seu estado.

Algorithm 7: Rotina principal de cada UAV

- 1 Calcular $\vec{a}(t + 1)$ a partir do *estado* atual (Algoritmos 3, 4, 5 e 6)
 - 2 **se** $|\vec{a}(t + 1)| > a_{max}$ (Equação (3.4)) **então**
 - 3 $\vec{a}(t + 1) = \vec{a}(t + 1) * (\frac{a_{max}}{|\vec{a}(t+1)|})$
 - 4 Atualizar $\vec{v}(t + 1)$ utilizando $\vec{a}(t + 1)$
 - 5 **se** $|\vec{v}(t + 1)| > v_{max}$ (Equação (3.6)) **então**
 - 6 $\vec{v}(t + 1) = \vec{v}(t + 1) * (\frac{v_{max}}{|\vec{v}(t+1)|})$
 - 7 Atualizar $\vec{x}(t + 1)$ (Equação 3.7)
-

5 *Apresentação do simulador proposto e suas funcionalidades*

*“Se você quiser educar um homem,
comece pela avó dele.”*

– Victor Hugo

Este capítulo apresenta a ferramenta computacional construída, os módulos que a compõem e explica detalhes de sua utilização.

5.1 **Introdução à ferramenta computacional**

O presente simulador, denominado *Swarm Unmanned Aerial Vehicle Simulator* (SUAVS), foi construído utilizando a plataforma JAVA e os conceitos de Orientação à Objetos. A figura 5.1 apresenta uma visão geral do simulador.

Basicamente, o usuário deve informar os parâmetros, presentes nos campos de textos ao lado esquerdo, e acionar a simulação. É possível visualizar os resultados numéricos e estatísticas da simulação conforme apresentado na seção 5.4.

Quando o número de simulações for igual a 1 (um), é possível, também, visualizar graficamente a simulação no lado direito do simulador conforme apresentado na figura 5.2, incluindo o raios dos sensores e velocidade em coordenadas polares de cada UAV.

O simulador possui uma interface gráfica que permite ao usuário entrar com os parâmetros desejados, acionar a simulação, observar a simulação visualmente e obter os resultados numéricos e estatísticos da simulação. Na construção da interface gráfica foram utilizados os seguintes componentes:

1. Biblioteca *Swing*: essa biblioteca foi utilizada para criar os campos de texto, rótulos, botões, desenho dos UAVs, alvos, obstáculos, raio dos sensores e do

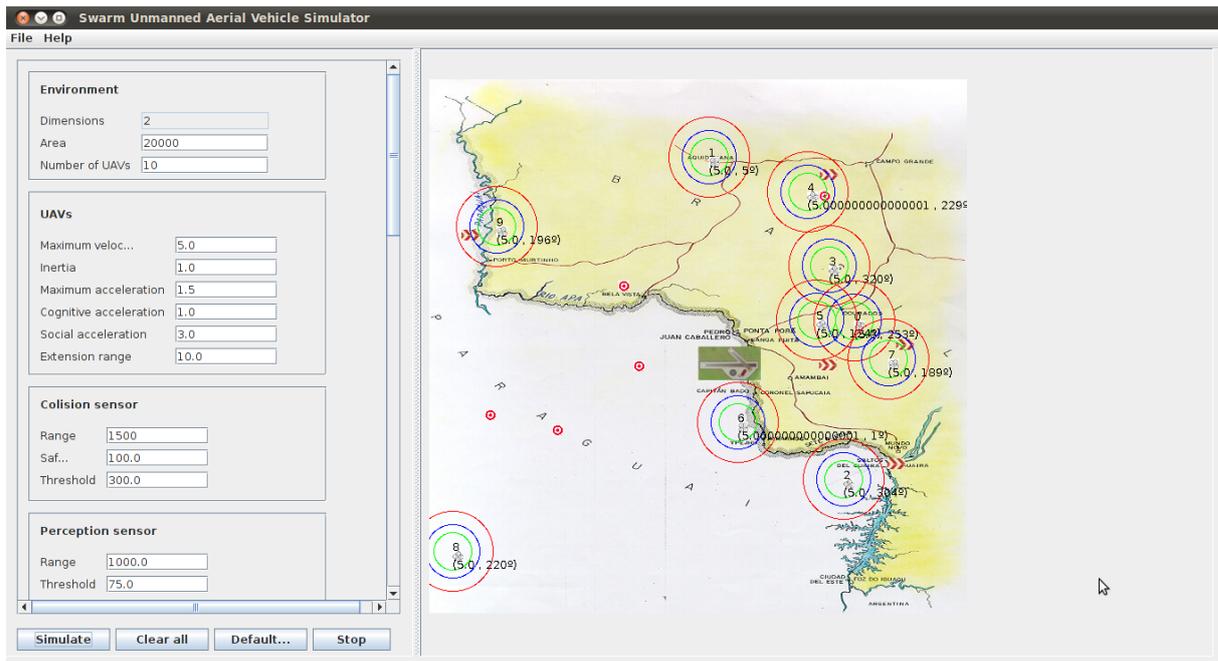


Figura 5.1: Visão geral do simulador.

ambiente;

2. Biblioteca *AWT*: essa biblioteca foi utilizada para capturar as ações dos componentes acima mencionados, possibilitando a animação da simulação;
3. Padrão de projeto *State*: esse padrão foi utilizado na implementação do UAV baseada na máquina de estados apresentada na seção 4.1 [40];
4. Padrão de projeto *Strategy*: esse padrão de projeto foi utilizado na implementação dos sensores anti-colisão, comunicação e percepção [40]. Esse padrão de projeto foi utilizado também na implementação da topologia de comunicação *ad hoc* utilizada nesse simulador;
5. Padrão de projeto *Observer*: esse padrão de projeto foi utilizado na implementação dos seguintes eventos:
 - Colisão ocorreu;
 - Comunicação estabelecida;
 - Comunicação perdida;
 - Alvo encontrado;
 - Alvo perdido;

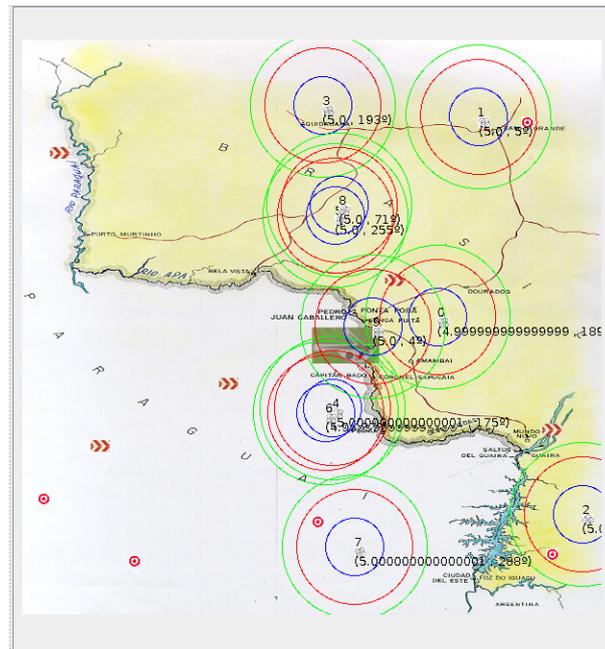


Figura 5.2: Painel de simulação.

- Alvo comunicado.

6. Java Threds;

Threds Java são utilizadas quando o número de simulações desejadas é maior que 1 (um). Dessa forma, é possível executar todas as simulações em paralelo.

Embora Java *Threds* tenham sido utilizadas para permitir mais de 1 (uma) simulação em paralelo, o simulador construído não foi implementado de forma assíncrona onde, por exemplo, cada UAV é executado dentro de uma *Thread* Java. No entanto, foi evitado o tratamento de cada UAV de forma sequencial padrão: dentro de cada iteração, o tratamento de cada UAV foi realizado de forma aleatória.

5.2 Parâmetros de simulação

Para que o usuário informe os parâmetros necessários à simulação, devido à quantidade de parâmetros envolvidos, os mesmos foram agrupados por coerência funcional.

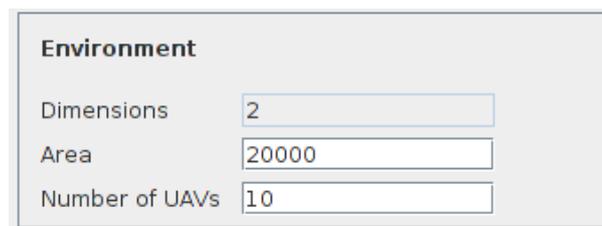
A seguir são detalhados os parâmetros em seus respectivos agrupamentos.

5.2.1 Ambiente

Os parâmetros relativos ao ambiente são:

- Dimensão: esse parâmetro não é editável, mas está aqui presente para simulações em três dimensões em futuras evoluções do simulador;
- Área: esse parâmetro determina a área do ambiente em m^2 ;
- Número de UAVs: esse parâmetro determina o número de UAVs que serão colocados inicialmente no ambiente. Os UAVs são distribuídos de maneira estatisticamente representativa.

Os parâmetros do ambiente são apresentado na figura 5.3.



Environment	
Dimensions	2
Area	20000
Number of UAVs	10

Figura 5.3: Parâmetros do ambiente.

5.2.2 UAV

Os parâmetros relativos ao UAV são:

- Velocidade máxima: esse parâmetro especifica a velocidade máxima dos UAVs em m/s ;
- Inércia: esse parâmetro especifica o fator de inércia utilizado para atualização da velocidade dos UAVs;
- Aceleração máxima: esse parâmetro especifica a aceleração máxima dos UAVs em m/s^2 ;
- Aceleração cognitiva: esse parâmetro especifica o fator aceleração cognitiva dos UAVs;
- Aceleração social: esse parâmetro especifica o fator aceleração social dos UAVs;

- Raio de extensão: esse parâmetro especifica o raio de extensão dos UAVs em metros considerando uma modelagem circular.

Os parâmetros do UAV são apresentados na figura 5.4.

UAVs	
Maximum veloc...	<input type="text" value="5.0"/>
Inertia	<input type="text" value="1.0"/>
Maximum acceleration	<input type="text" value="1.5"/>
Cognitive acceleration	<input type="text" value="1.0"/>
Social acceleration	<input type="text" value="3.0"/>
Extension range	<input type="text" value="10.0"/>

Figura 5.4: Parâmetros dos UAVs.

5.2.3 Sensor anti-colisão

Os parâmetros relativos ao sensor anti-colisão são:

- Raio de alcance: esse parâmetro determina o raio de alcance do sensor de acordo com as especificações da seção 3.2.3;
- Limiar: esse parâmetro determina o limiar do sensor de acordo com as especificações da seção 3.2.3;
- Segurança: esse parâmetro determina a segurança do sensor de acordo com as especificações da seção 3.2.3.

Os parâmetros do sensor anti-colisão são apresentados na figura 5.5.

Colision sensor	
Range	<input type="text" value="1500"/>
Saf...	<input type="text" value="100.0"/>
Threshold	<input type="text" value="300.0"/>

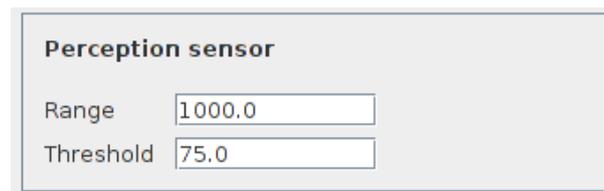
Figura 5.5: Parâmetros do sensor de colisão.

5.2.4 Sensor de percepção do ambiente

Os parâmetros relativos ao sensor de percepção do ambiente são:

- Raio de alcance: esse parâmetro determina o raio de alcance do sensor de acordo com as especificações da seção 3.2.5;
- Limiar: esse parâmetro determina o limiar do sensor de acordo com as especificações da seção 3.2.5;
- Segurança: esse parâmetro determina a segurança do sensor de acordo com as especificações da seção 3.2.5.

Os parâmetros do sensor de percepção do ambiente são apresentados na figura 5.6.



The image shows a configuration window titled "Perception sensor". It contains two input fields: "Range" with the value "1000.0" and "Threshold" with the value "75.0".

Figura 5.6: Parâmetros do sensor de percepção do ambiente.

5.2.5 Sensor de percepção dos limites do ambiente

Os parâmetros relativos ao sensor de percepção dos limites do ambiente são:

- Raio de alcance: esse parâmetro determina o raio de alcance do sensor de acordo com as especificações da seção 3.2.6;
- Limiar: esse parâmetro determina o limiar do sensor de acordo com as especificações da seção 3.2.6;
- Segurança: esse parâmetro determina a segurança do sensor de acordo com as especificações da seção 3.2.6.

Os parâmetros do sensor de percepção dos limites do ambiente são apresentados na figura 5.7.

Limits perception sensor	
Range	500.0
Saf...	75.0
Threshold	500.0

Figura 5.7: Parâmetros do sensor de percepção dos limites do ambiente.

5.2.6 Sensor de comunicação

Os parâmetros relativos ao sensor de comunicação são:

- Raio de alcance: esse parâmetro determina o raio de alcance do sensor de acordo com as especificações da seção 3.2.4;
- Limiar: esse parâmetro determina o limiar do sensor de acordo com as especificações da seção 3.2.4;
- Segurança: esse parâmetro determina a segurança do sensor de acordo com as especificações da seção 3.2.4;
- Número máximo de comunicações: esse parâmetro determina o número máximo de comunicações de acordo com as especificações da seção 3.2.4.

Os parâmetros do sensor de comunicação são apresentados na figura 5.8.

Communication sensor	
Range	1000.0
Saf...	900.0
Threshold	800.0
Maximum connections	2

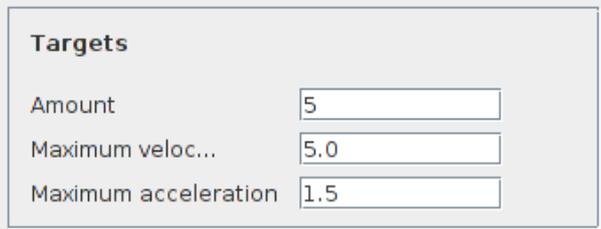
Figura 5.8: Parâmetros do sensor de comunicação.

5.2.7 Alvos

Os parâmetros relativos aos alvos são:

- Quantidade: esse parâmetro determina o número de alvos que serão colocados inicialmente no ambiente. Os alvos são distribuídos de maneira estatisticamente representativa;
- Velocidade máxima: esse parâmetro especifica a velocidade máxima dos alvos em m/s ;
- Aceleração máxima: esse parâmetro especifica a aceleração máxima dos alvos em m/s^2 .

Os parâmetros dos alvos são apresentados na figura 5.9.



Targets	
Amount	5
Maximum veloc...	5.0
Maximum acceleration	1.5

Figura 5.9: Parâmetros dos alvos.

5.2.8 Obstáculos

Os parâmetros relativos aos obstáculos são:

- Quantidade: esse parâmetro determina o número de obstáculos que serão colocados inicialmente no ambiente. Os obstáculos são distribuídos de maneira estatisticamente representativa;
- Raio de extensão: esse parâmetro especifica o raio de extensão dos obstáculos em metros considerando uma modelagem circular.

Os parâmetros dos obstáculos são apresentados na figura 5.10.



Obstacles	
Amount	5
Range	10.0

Figura 5.10: Parâmetros dos obstáculos.

5.2.9 Simulação

Os parâmetros relativos à simulação são:

- Número de simulações: esse parâmetro determina o número de simulações paralelas para os parâmetros informados;
- Número de iterações para recalculer a área de cobertura: esse parâmetro determina o número de iterações para recalculer a área de cobertura;
- Número de iterações: esse parâmetro determina o número de iterações, caracterizando o tempo de missão.

5.3 Ações do simulador

O simulador possui as seguintes ações:

- Simular: essa ação dispara o processo de simulação considerando os parâmetros acima mencionados;
- Limpar todos: essa ação apaga os valores de todos os parâmetros acima mencionados;
- Padrão: essa ação carrega os valores padrões para os parâmetros acima mencionados. Esses valores são configurados pelo usuário;
- Parar simulação: essa ação aborta a simulação atual;
- Salvar resultados: essa ação salva os resultados das simulações em formato CSV (*Comma-Separated Values*);
- Exibir painel de simulação: essa ação, quando o número de simulações for igual a um, exibe ou não o painel de simulação;
- Exibir sensores: essa ação exibe ou oculta os raios dos sensores anti-colisão, comunicação e percepção do ambiente;
- Exibir velocidades: essa ação exibe ou oculta as informações relativas às velocidades em coordenadas polares de cada UAV no painel de simulação.

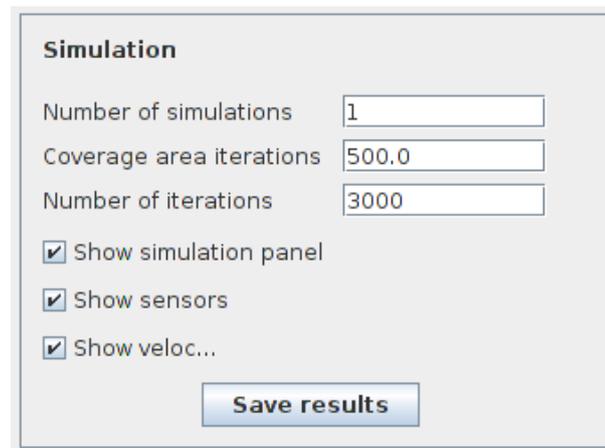
As ações são apresentadas nas figuras 5.11 e 5.12.



Figura 5.11: Ações do simulador.

5.4 Estatísticas da simulação

Quando o número de simulações é igual a um, são apresentadas as estatísticas da simulação. Um exemplo das estatísticas de simulação é apresentado na figura 5.12.



Simulation

Number of simulations

Coverage area iterations

Number of iterations

Show simulation panel

Show sensors

Show veloc...

Figura 5.12: Exemplo das estatísticas de simulação.

As informações da simulação são:

- Número de iterações;
- Percentual do tempo acumulado de saídas do espaço aéreo em relação ao tempo total de simulação;
- Percentual de UAVs que colidiram em relação ao total de UAVs inicialmente no ambiente;
- Comunicação;
 - Percentual do número de comunicações estabelecidas no momento em relação ao total de possíveis conexões;
 - Percentual da média de comunicações estabelecidas em relação ao total de possíveis conexões;

- Percentual do número máximo de comunicações estabelecidas em relação ao total de possíveis conexões;
- Percentual do número de comunicações estabelecidas em relação às comunicações perdidas.
- Área de cobertura;
 - Percentual da área coberta no momento em relação à área total do ambiente;
 - Percentual da média área coberta no momento em relação à área total do ambiente;
 - Percentual da máxima área coberta no momento em relação à área total do ambiente;
 - Percentual da mínimo área coberta no momento em relação à área total do ambiente.
- Rastreamento de alvos;
 - Percentual do número de alvos rastreados no momento em relação ao número total de alvos;
 - Percentual da média de alvos rastreados no momento em relação ao número total de alvos;
 - Percentual do número máximo de alvos rastreados no momento em relação ao número total de alvos;
 - Percentual do número de de alvos rastreados em relação aos alvos perdidos.

6 *Experimentos e resultados das simulações*

“Deus não joga dados.”

– Albert Einstein

Esse capítulo apresenta os experimentos elaborados, os resultados visuais e numéricos das simulações realizadas bem como realiza uma discussão em cima dos mesmos.

6.1 Resultados visuais

Antes de abordar a simulação numérica, é interessante a apresentação de resultados visuais plausíveis.

6.1.1 Formação de figuras geométricas simétricas

O primeiro comportamento que emerge da interação entre os UAVs é, principalmente, quando em estado de rastreamento, a formação de figuras geométricas simétricas a partir da configuração de suas posições. Tais figuras variam com o número de UAVs envolvidos no rastreamento.

As seguir, são apresentadas as figuras geométricas simétricas formadas para o número de UAVs variando de três à oito durante qualquer simulação visual. A formação dessas figuras é um comportamento que emerge das interações entre os UAVs.

- Três UAVs;

As posições de três UAVs rastreando um alvo configuram um triângulo equilátero

6.1 Resultados visuais

com o alvo em seu centro. Tal resultado pode ser verificado na figura 6.1.



Figura 6.1: Triângulo equilátero, contendo o alvo em seu centro, formado por três UAVs em estado de rastreamento.

- Quatro UAVs;
 As posições de quatro UAVs rastreando um alvo configuram um quadrado com o alvo em seu centro. Tal resultado pode ser verificado na figura 6.2.



Figura 6.2: Quadrado, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.

- Cinco UAVs;
 As posições de cinco UAVs rastreando um alvo configuram um pentágono regular com o alvo em seu centro. Tal resultado pode ser verificado na figura 6.3.
- Seis UAVs;
 As posições de seis UAVs rastreando um alvo configuram um hexágono simétrico



Figura 6.3: Pentágono regular, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.

não regular com o alvo em seu centro. Tal resultado pode ser verificado na figura 6.4.



Figura 6.4: Hexágono simétrico não regular, contendo o alvo em seu centro, formado por quatro UAVs em estado de rastreamento.

- Sete UAVs;
 As posições de sete UAVs rastreando um alvo configuram um pentágono regular com dois UAVs e o alvo em seu centro. Tal resultado pode ser verificado na figura 6.5.
- Oito UAVs;
 As posições de oito UAVs rastreando um alvo configuram um hexágono regular com dois UAVs e o alvo em seu centro. Tal resultado pode ser verificado na figura 6.6.

Outros dois comportamentos foram observados:

6.1 Resultados visuais



Figura 6.5: Pentágono regular, contendo dois UAVs e o alvo em seu centro, formado por sete UAVs em estado de rastreamento.



Figura 6.6: Hexágono regular, contendo dois UAVs e o alvo em seu centro, formado por oito UAVs em estado de rastreamento.

- Rotação das figuras geométricas formadas;
 Durante a locomoção do alvo, é possível observar a rotação de tais figuras geométricas tanto em torno de um UAV, consequência da diminuição da velocidade do UAV, quanto em torno do alvo rastreado, visando manter a formação de rastreamento.
- Adaptabilidade à velocidade do alvo.
 Durante o rastreamento, ocorre o comportamento cibernético descrito na equação (6.1),

$$\vec{v}_{uav} \rightarrow \vec{v}_{alvo} \tag{6.1}$$

ou seja, mesmo a velocidade do alvo sendo desconhecida, os UAVs calibram sua velocidade de forma que ela atinja a direção, sentido e magnitude da velocidade do alvo rastreado.



Figura 6.8: Formação de sub-enxames: conectividade global baixa.

melho, já se encontram rastreando os dois alvos em paralelo, destacados em azul, no entanto, estão apenas conectados devido à existência de UAVs, destacados em verde, equidistantes dos dois sub-enxames em formação. O objetivo de tais UAVs é ser ponte de roteamento entre os dois sub-enxames.

4. Quarto momento;

Os UAVs responsáveis por manter a conectividade entre os dois sub-enxames formados, para não ficarem sozinhos e desconectados, tomam a decisão de seguir um dos dois enxames e, assim, a conectividade global é perdida. Conforme visualizado na figura 6.10, os dois sub-enxames, destacados em vermelho, se encontram desconectados um do outro e rastreando alvos diferentes em paralelo, destacados em azul.

6.2 Resultados numéricos

Embora os resultados visuais apontem características interessantes, ainda se faz necessária a investigação da efetividade do sistema.

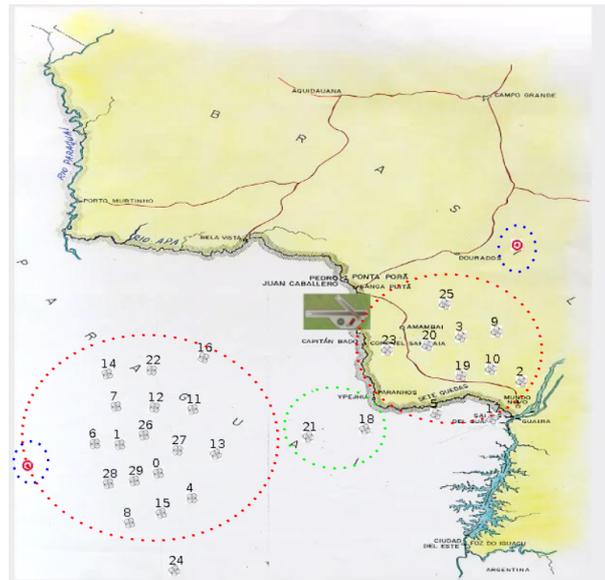


Figura 6.9: Formação de sub-enxames: conectividade global crítica.

As métricas utilizadas são:

- Colisões;
- Alvos rastreados;
- Cobertura territorial.

Os parâmetros utilizados, descritos na seção 5.2, por padrão, estão descritos na tabela 6.1.

6.2.1 Conjunto experimental 1

O experimento a seguir tem como objetivo confirmar a seguinte hipótese: a presença da comunicação entre os UAVs não só cumpre os objetivos estabelecidos, mas também torna o sistema, como um todo, mais efetivo.

Dessa forma foram utilizados os seguintes parâmetros:

1. Número de UAVs: 40;
2. Número máximo de comunicações estabelecidas com UAVs vizinhos: 0 e 2.

Foram realizadas simulações da combinação dos parâmetros acima. A tabela 6.2 mostra os resultados obtidos para as métricas definidas anteriormente.

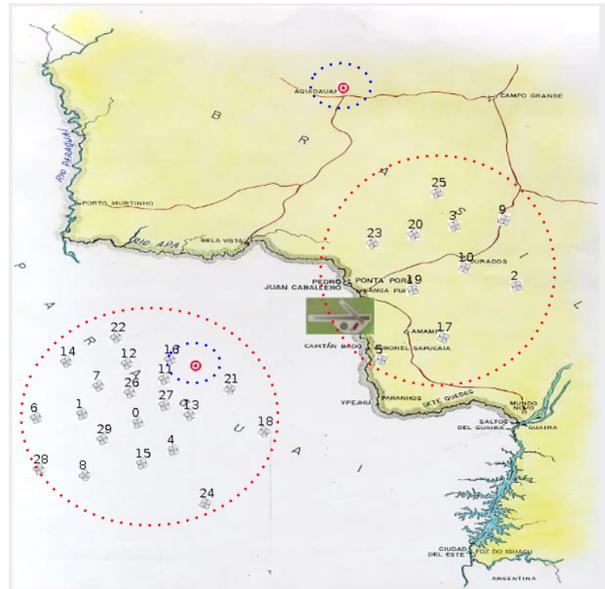


Figura 6.10: Formação de sub-ênxames: conectividade global inexistente.

O gráfico na figura 6.11 melhor apresenta os grupos de simulação.

O ganho obtido com a presença da comunicação em relação às métricas colisão e cobertura territorial é significativo. No entanto, os resultados são similares em relação à métrica alvos rastreados. Sendo assim, serão alterados os seguintes parâmetros:

1. Velocidade máxima dos alvos: 7,0 m/s

A tabela 6.3 mostra os resultados obtidos para as métricas definidas anteriormente.

O gráfico na figura 6.12 melhor apresenta os grupos de simulação.

Analisando os dados, o ganho obtido com a presença da comunicação em relação às métricas colisão e cobertura territorial é significativo.

A presença da comunicação terá impacto ainda mais significativo nos resultados quando o mecanismo de *health monitoring* for acrescentado.

A princípio, duas consequências podem decorrer desta ação:

1. Perda de alvos rastreados;
 A métrica de alvos rastreados será impactada de forma substancial, pois, quando a energia ou qualquer avaria impactar o UAV, este terá como objetivo abandonar o alvo atualmente rastreado e retornar à base.
2. Períodos de tempo não patrulhados.

Tabela 6.1: Parâmetros mantidos constantes em todos os experimentos.

Parâmetro	Valor
Área do ambiente	10.000 m^2
Velocidade máxima dos UAVs	5 m/s
Inércia dos UAVs	1
Aceleração máxima dos UAVs	1,5 m/s^2
Aceleração cognitiva dos UAVs	1,0
Aceleração social dos UAVs	3,0
Raio de extensão dos UAVs	10,0 m
Raio de alcance do sensor anti-colisão	1.000 m
Limiar do sensor anti-colisão	100 m
Segurança do sensor anti-colisão	300 m
Raio de alcance do sensor de percepção do ambiente	1.000 m
Limiar do sensor de percepção do ambiente	75,0 m
Raio de alcance do sensor de percepção dos limites do ambiente	1.000 m
Limiar do sensor de percepção dos limites do ambiente	75,0 m
Segurança do sensor de percepção dos limites do ambiente	500 m
Raio de alcance do sensor de comunicação	1.000 m
Limiar do sensor de comunicação	900,0 m
Segurança do sensor de comunicação	800,0 m
Número máximo de comunicações do sensor de comunicação	2 m
Quantidade de alvos	5
Velocidade máxima dos alvos	5,0 m/s
Aceleração máxima dos alvos	1,5 m/s
Quantidade de obstáculos	2
Raio de extensão dos obstáculos	10,0 m
Número de simulações	30,0
Número de iterações para recalculer a área de cobertura	500
Número de iterações	100.000

Na ausência de coordenação entre os UAVs, pode não ocorrer o revezamento para o restabelecimento das condições normais de operação. Tal revezamento, provavelmente, ocorrerá de maneira natural devido à utilização da comunicação e o paradigma de inteligência de enxames. A consequência será o não revezamento na patrulha, podendo gerar períodos de patrulha ociosa ou pouco significativa, pois, a maioria dos UAVs se encontrarão na base.

6.2.2 Conjunto experimental 2

O experimento a seguir tem como objetivo validar a seguinte hipótese: o aumento do número de UAVs utilizados aumenta a efetividade do sistema.

Tabela 6.2: Resultado do experimento 1 quando $v_{alvmax} = v_{uavmax}$.

	Colisões	Cobertura territorial	Alvos rastreados
Rede de comunicação <i>ad hoc</i> 2-conectado			
Média	10,92%	40,23%	99,07%
Desvio padrão	5,67%	1,01%	0,91%
Sem comunicação			
Média	40,50%	34,94%	99,45%
Desvio padrão	7,02%	0,97%	0,51%

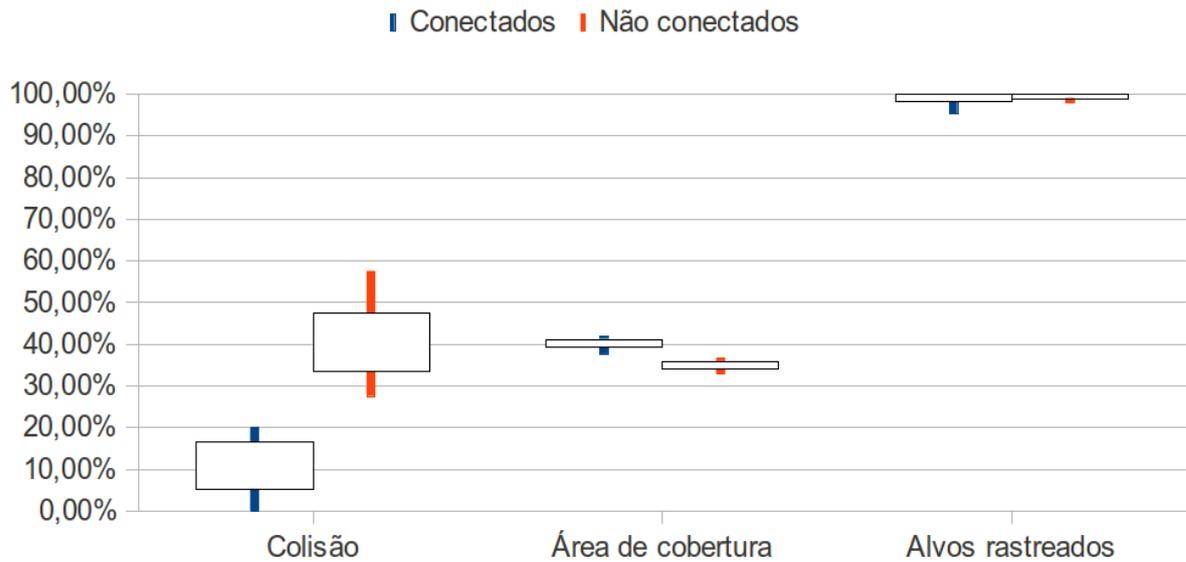


Figura 6.11: Experimento 1: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvmax} = v_{uavmax}$.

Dessa forma foram utilizados os seguintes parâmetros:

1. Número de UAVs: 1, 5, 20 e 40.

A tabela 6.4 mostra os resultados obtidos para as métricas definidas anteriormente.

O gráfico na figura 6.13 melhor apresenta os grupos de simulação.

Assim como foi feito anteriormente, serão alterados os seguintes parâmetros:

1. Velocidade máxima dos alvos: 7,0 m/s

A tabela 6.5 mostra os resultados obtidos para as métricas definidas anteriormente.

O gráfico na figura 6.14 melhor apresenta os grupos de simulação.

Tabela 6.3: Resultado do experimento 1 quando $v_{alvomax} > v_{uavmax}$.

	Colisões	Cobertura territorial	Alvos rastreados
Rede de comunicação <i>ad hoc</i> 2-conectado			
Média	6,50%	70,34%	81,77%
Desvio padrão	5,48%	1,82%	1,78%
Sem comunicação			
Média	38,25%	64,92%	72,36%
Desvio padrão	6,02%	3,32%	2,59%

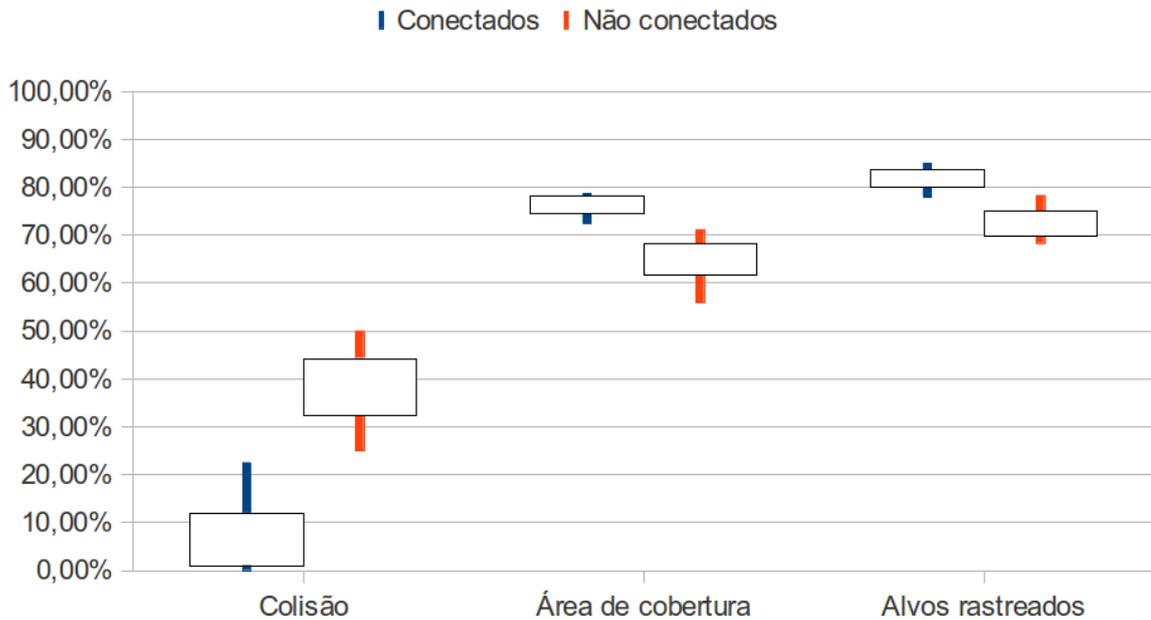


Figura 6.12: Experimento 1: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvomax} > v_{uavmax}$

De acordo com os resultados apresentados na Tabela 6.5, a efetividade do sistema aumenta quando o número de UAVs também aumenta.

Os parâmetros dos sensores anti-colisão, de comunicação e de percepção do ambiente, assim como outros parâmetros, também são fatores importantes na análise do sistema. No entanto, não serão analisados mais parâmetros neste trabalho por razões de escopo.

6.3 Discussões dos resultados obtidos

Os resultados obtidos evidenciam a potencialidade da utilização desse tipo de sistemas para as atividades em questão. É importante destacar que, durante todas as simulações realizadas, nenhum UAV invadiu o espaço aéreo vizinho.

Tabela 6.4: Resultado do experimento 2 quando $v_{alvmax} = v_{uavmax}$.

	Colisões	Cobertura territorial	Alvos rastreados
		1 UAV	
Média	0,0%	0,15%	0,0%
Desvio padrão	0,0%	0,15%	0,0%
		5 UAVs	
Média	9,33%	20,86%	59,38%
Desvio padrão	19,46%	2,89%	9,11%
		20 UAVs	
Média	8,83%	35,69%	96,67%
Desvio padrão	7,73%	1,35%	2,63%
		40 UAVs	
Média	10,92%	40,23%	99,07%
Desvio padrão	5,67%	1,01%	0,91%

Tabela 6.5: Resultado do experimento 2 quando $v_{alvmax} > v_{uavmax}$.

	Colisões	Cobertura territorial	Alvos rastreados
		1 UAV	
Média	0,0%	0,13%	0,0%
Desvio padrão	0,0%	0,12%	0,0%
		5 UAVs	
Média	3,33%	20,74%	20,26%
Desvio padrão	10,61%	3,64%	5,08%
		20 UAVs	
Média	4,00%	54,77%	61,13%
Desvio padrão	6,21%	2,40%	2,90%
		40 UAVs	
Média	6,50%	76,34%	81,77%
Desvio padrão	5,48%	1,82%	1,78%

De forma específica, foi evidenciado que a presença de comunicação é vital para a efetividade do sistema e que o acréscimo de UAVs também contribui para o aumento da mesma.

A análise dos resultados põe em evidência a escalabilidade do sistema. A maior prova disso é a análise da métrica colisões.

Fica claro que, antes da implantação de UAVs em campo para realização das atividades de patrulha e rastreamento coordenadas, um dimensionamento através de simulação, se realizado anteriormente, evita a utilização de recursos não necessários e, conseqüentemente, reduz custos de recursos, projeto e operacionais.

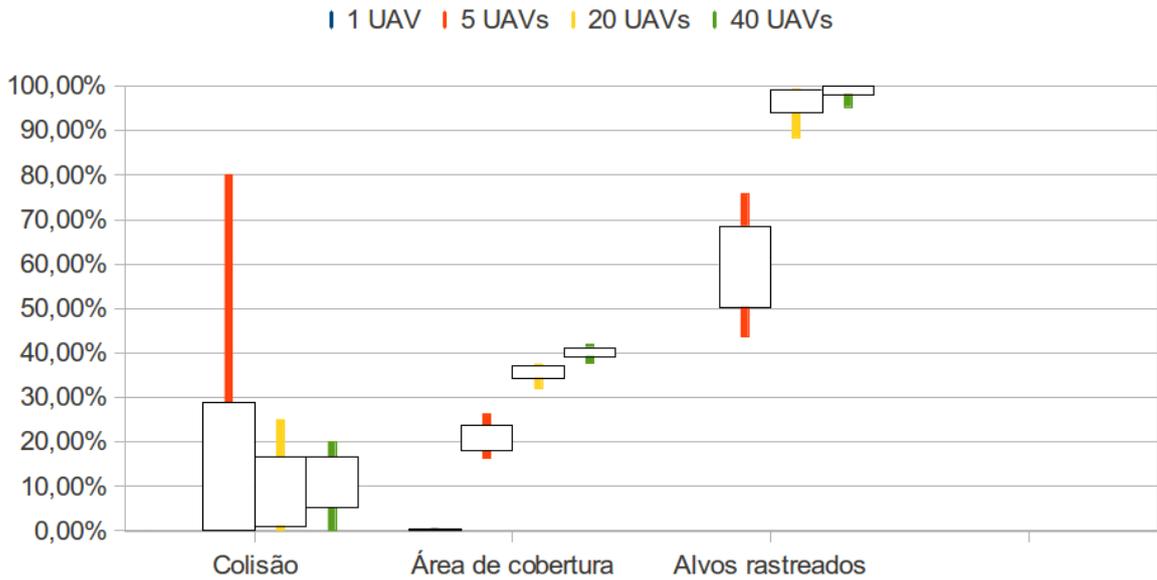


Figura 6.13: Experimento 2: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvomax} = v_{uavmax}$

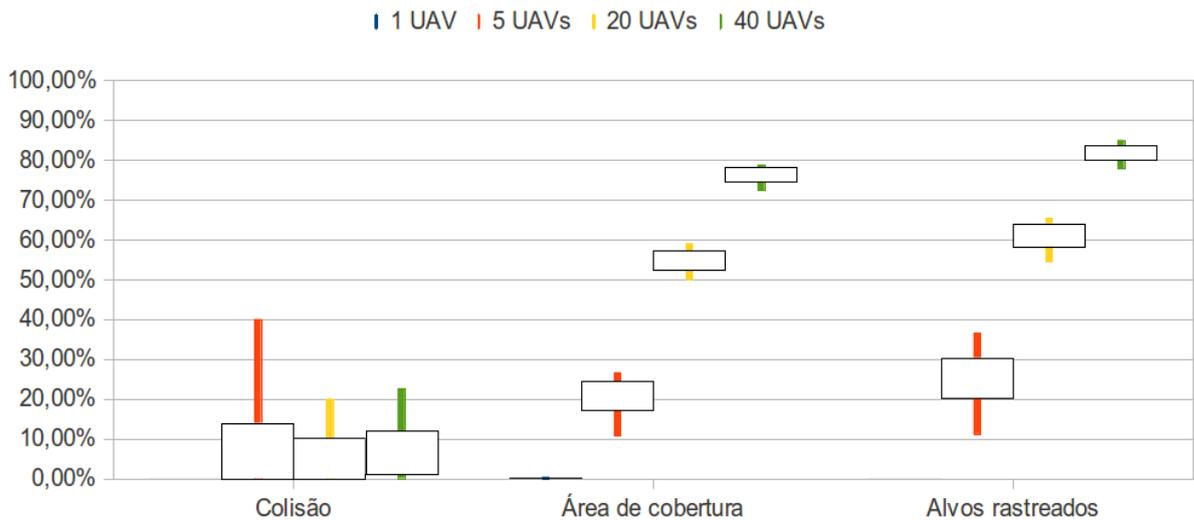


Figura 6.14: Experimento 2: Efetividade do sistema em relação à colisões, cobertura territorial e alvos rastreados quando $v_{alvomax} > v_{uavmax}$

7 *Conclusões e Trabalhos Futuros*

*“O futuro dependerá
daquilo que fazemos no presente.”*

– Mohandas Karamchand Gandhi

Neste trabalho foi construído um simulador que possibilitasse a visualização do comportamento coordenado de múltiplos UAVs utilizando inteligência de enxames. As estratégias e mecanismos adotados, juntamente com os resultados obtidos, evidenciam a possibilidade de avaliar previamente a utilização de UAVs em atividades de patrulha territorial e rastreamento de alvos móveis simultaneamente.

Este simulador tem potencial para auxiliar no dimensionamento de UAVs reais à serem implantados em campo, reduzindo custos de projeto, com recursos e operacionais.

Em trabalhos futuros, serão implementados os seguintes requisitos:

1. Acrescentar mais uma dimensão, visando possibilitar simulação em três dimensões, visto a importância da variável altitude no domínio dos veículos aéreos.
2. Acrescentar fatores aerodinâmicos não considerados neste projeto, aproximando-se da realidade aerodinâmica existente;
3. Acrescentar fatores energéticos, incluindo consumo de energia de cada UAV, relacionando também este consumo energético com os recursos utilizados pelo UAV como, por exemplo, sensores, dinâmica, etc;
4. Acrescentar o mecanismo de *Health Monitoring* para adequação com a realidade das missões de patrulha, monitoramento, etc.
5. Utilizar outras abordagens e algoritmos de inteligência de enxames existentes;
6. Permitir a utilização de UAVs, alvos e obstáculos heterogêneos.

Referências Bibliográficas

- [1] BETHKE, B. *Persistent Vision-Based Search and Track Using Multiple UAVs*. Dissertação (Mestrado) — Massachusetts Institute of Technology, jun. 2007.
- [2] GOEBEL, G. *Unmanned Aerial Vehicles*. Acessado em: Outubro de 2011. Disponível em: <<http://www.vectorsite.net/twuav.html>>.
- [3] PORTELA, M. *Simulação de robôs aquáticos utilizando inteligência de enxames para inspeção de dutos submarinos*. 2009. Trabalho de conclusão de curso - Engenharia da Computação, UPE, Recife-PE.
- [4] SINSLEY, G. et al. An intelligent controller for collaborative unmanned air vehicles. In: *IEEE Symposium on Computational Intelligence in Security and Defense Applications*. [S.l.: s.n.], 2007. p. 139–144.
- [5] J. P. How, E. K.; KUWATA, Y. Flight demonstrations of cooperative control for uav teams. In: *Proceedings of 3rd Unmanned Unlimited Technical Conference, Workshop and Exhibit*. [S.l.: s.n.], 2004. p. 505–513.
- [6] BONABEAU, E.; DORIGO, M.; THERAULAZ, G. *Swarm intelligence: from natural to artificial systems*. New York, NY, USA: Oxford University Press, Inc., 1999.
- [7] KENNEDY, J.; EBERHART, R.; SHI, Y. *Swarm Intelligence*. 1. ed. San Francisco, Califórnia, USA: Morgan Kaufmann Publishers, 2001.
- [8] ENGELBRECHT, A. P. *Computational Intelligence An Introduction*. 1. ed. West Sussex, Inglaterra: Wiley, 2002.
- [9] KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks*. [S.l.: s.n.], 1995. v. 4, p. 1942–1948.
- [10] KENNEDY, J. Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance. In: *Proceedings of the Congress on Evolutionary Computation*. [S.l.: s.n.], 1999. v. 3, p. 1931–1938.
- [11] EBERHART, R.; SHI, Y. *Computational Intelligence: concepts to implementations*. Burlington, MA, EUA: Elsevier Inc, 2007.
- [12] SHARKEY, A. J. C. The application of swarm intelligence to collective robots. In: *Advances in Applied Artificial Intelligence*. [S.l.]: John Fulcher, Idea Group Publishing, 2006. p. 157–185.

- [13] CAO, Y. et al. Cooperative mobile robotics: antecedents and directions. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. 'Human Robot Interaction and Cooperative Robots'*. [S.l.: s.n.], 1995. v. 4, p. 226–234.
- [14] SAHIN, E. et al. Swarm robotics. In: BLUM, C.; MERKLE, D. (Ed.). *Swarm Intelligence*. [S.l.]: Springer Berlin Heidelberg, 2008, (Natural Computing Series). p. 87–100.
- [15] PARKER, L. E. Current research in multirobot systems. *Artificial Life Robotics*, v. 7, p. 1–5, 2003.
- [16] SHARKEY, A. J. Robots, insects and swarm intelligence. *Artif. Intell. Rev.*, v. 26, p. 255–268, dez. 2006.
- [17] PUGH, J.; MARTINOLI, A. Distributed adaptation in multi-robot search using particle swarm optimization. In: ASADA, M. et al. (Ed.). *From Animals to Animats 10*. [S.l.]: Springer Berlin / Heidelberg, 2008, (Lecture Notes in Computer Science, v. 5040). p. 393–402.
- [18] DHARIWAL, A.; SUKHATME, G.; REQUICHA, A. Bacterium-inspired robots for environmental monitoring. In: *Proceedings of IEEE International Conference on Robotics and Automation*. [S.l.: s.n.], 2004. v. 2, p. 1436–1443.
- [19] MENG, Y.; KAZEEM, O.; MULLER, J. A hybrid aco/pso control algorithm for distributed swarm robots. In: *Proceedings of IEEE Swarm Intelligence Symposium*. [S.l.: s.n.], 2007. p. 273–280.
- [20] WANG, Y.; LIANG, A.; GUAN, H. Frontier-based multi-robot map exploration using particle swarm optimization. In: *Proceedings of IEEE Symposium on Swarm Intelligence*. [S.l.: s.n.], 2011. p. 1–6.
- [21] ZHU, Q.; LIANG, A.; GUAN, H. A pso-inspired multi-robot search algorithm independent of global information. In: *Proceedings of IEEE Symposium on Swarm Intelligence*. [S.l.: s.n.], 2011. p. 1–7.
- [22] WANG, Z.; KUMAR, V. Object closure and manipulation by multiple cooperating mobile robots. In: *Proceedings of IEEE International Conference on Robotics and Automation*. [S.l.: s.n.], 2002. v. 1, p. 394–399.
- [23] GROSS, R.; MONDADA, F.; DORIGO, M. Transport of an object by six pre-attached robots interacting via physical links. In: *Proceedings of IEEE International Conference on Robotics and Automation*. [S.l.: s.n.], 2006. p. 1317–1323.
- [24] GROSS, R. et al. Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, v. 22, n. 6, p. 1115–1130, dez. 2006.
- [25] LEI, B.; LI, W. A fuzzy behaviours fusion algorithm for mobile robot real-time path planning in unknown environment. In: *Proceedings of IEEE International Conference on Integration Technology*. [S.l.: s.n.], 2007. p. 173–178.

- [26] RIGATOS, G. g. Distributed gradient and particle swarm optimization for multi-robot motion planning. *Robotica*, Cambridge University Press, New York, NY, USA, v. 26, p. 357–370, May 2008.
- [27] SAHIN, E. et al. Swarm-bot: pattern formation in a swarm of self-assembling mobile robots. In: *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*. [S.l.: s.n.], 2002. v. 4, p. 6–9.
- [28] SUN, R.; PETERSON, T. Multi-agent reinforcement learning: weighting and partitioning. *Neural Netw*, Elsevier Science Ltd., Oxford, UK, UK, v. 12, p. 727–753, jun. 1999.
- [29] ORTIZ, C. L.; VINCENT, R.; MORISSET, B. Task inference and distributed task management in the centibots robotic system. In: *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*. New York, NY, USA: ACM, 2005. (AAMAS '05), p. 860–867.
- [30] SAHIN, E. Swarm robotics: From sources of inspiration to domains of application. In: PAYTON, D.; BENI, G. (Ed.). *Swarm Robotics*. [S.l.]: Springer Berlin Heidelberg, 2005, (Lecture Notes in Computer Science, v. 3342). p. 10–20.
- [31] MCCORMICK, B. W. *Aerodynamics, Aeronautics, and Flight Mechanics*. International 2nd revised. [S.l.]: Wiley, 1995.
- [32] AERONAUTICAL, G. A. *General Atomics Predator page*. Acessado em: Novembro de 2011. Disponível em: <<http://www.gasi.com/products/aircraft/predator.php>>.
- [33] IEEE. *IEEE 802.15 WPAN Task Group 4 (TG4)*. Acessado em: Novembro de 2011. Disponível em: <<http://http://www.ieee802.org/15/pub/TG4.html>>.
- [34] EBERHART, R.; SHI, Y. Tracking and optimizing dynamic systems with particle swarms. In: *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*. [S.l.: s.n.], 2001. v. 1, p. 94–100 vol. 1.
- [35] LI, C.; YANG, S. A generalized approach to construct benchmark problems for dynamic optimization. In: *Simulated Evolution and Learning*. [S.l.: s.n.], 2008. p. 391–400.
- [36] HU, X.; EBERHART, R. Adaptive particle swarm optimization: detection and response to dynamic systems. In: *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*. [S.l.: s.n.], 2002. v. 2, p. 1666–1670.
- [37] BERGH, F. van den; ENGELBRECHT, A. A cooperative approach to particle swarm optimization. *Evolutionary Computation, IEEE Transactions on*, v. 8, n. 3, p. 225–239, jun 2004.
- [38] LIU, L.; WANG, D.; YANG, S. Compound particle swarm optimization in dynamic environments. In: *Proceedings of the 2008 conference on Applications of evolutionary computing*. Berlin, Heidelberg: Springer-Verlag, 2008. (Evo'08), p. 616–625.
- [39] WEICKER, K. Performance measures for dynamic environments. In: *Parallel Problem Solving from Nature—PPSN VII*. [S.l.]: Springer-Verlag, 2002. p. 64–73.

- [40] GAMMA, E. et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. 1. ed. [S.l.]: Addison-Wesley Professional, 1994.

APÊNDICE A – Especificação de requisitos do simulador

Esse anexo tem como objetivo especificar os requisitos dos trabalhos de conclusão dos estudantes Diego Pinheiro e David Alain do curso de Engenharia de Computação da Universidade de Pernambuco.

R01 – Perceber invasão no ambiente

Em sua patrulha, o UAV deve perceber que ocorreu invasões no ambiente de patrulha.

R02 – Perceber geolocalização

O UAV deve ter conhecimento de sua geolocalização para auxiliar a patrulha e melhor gerenciar os atuadores.

R03 – Perceber obstáculos

O UAV tem conhecimento sobre obstáculos em sua volta. Essa percepção é limitada por um raio de alcance. A implementação prática deve ser construída utilizando sensores tipo RADAR.

R04 – Comunicar-se com outros UAVs

O UAV deve possuir uma forma de comunicação com os outros UAVs e com a estação terrestre direta ou indiretamente (utilização de redes de paradigma ad hoc, padrão 802.15.4). Essa comunicação é limitada por um raio de alcance.

R05 – Monitorar saúde

Durante toda a operação, o UAV deve monitorar sua respectiva saúde, somente engajando-se em missões se sua saúde estiver em boas condições. Em caso de anormalidade em sua saúde, o UAV deve priorizar o restabelecimento de suas condições normais, não permitindo sua participação em outras operações. Os componentes de saúde são:

- Combustível;
- Taxa de consumo, em energia/unidade de tempo;
- Capacidade máxima de energia.

R06 – Impedir colisão

Utilizando sua percepção de obstáculos e geolocalização, o UAV deve identificar em sua proximidade obstáculos ou outros UAVs e alterar seu direcionamento de forma a impedir possíveis colisões.

R07 – Rastrear alvos móveis

Ao perceber invasão no ambiente, o UAV deve identificar o alvo e rastreá-lo ao longo do ambiente enquanto estiver no espaço de busca.

R08 – Comunicar invasão com outros UAVs

Ao perceber invasão no ambiente, antes de iniciar e durante o rastreamento o UAV deve comunicar a respectiva invasão com as informações do alvo.

R09 – Controlar atuadores

Nas operações de patrulha e principalmente nas operações de rastreamento, para satisfazer seus objetivos, como minimizar sua distância em relação ao alvo. o UAV deve controlar seus atuadores de forma inteligente. Os atuadores serão:

- Propulsão, para aumentar ou diminuir a aceleração através das Rotações Por Minuto (RPM) do motor;
- Ângulo de subida/descida, para ganhar ou perder altitude;
- Ângulo de direcionamento, para rotacionar para esquerda ou para direita.

R10 – Realizar cobertura territorial mínima

O UAV deve patrulhar o ambiente buscando, em conjunto com os outros UAVs, cobrir o máximo de território possível. Os UAVs devem garantir um cobertura territorial mínima.

R11 – Patrulhar ambientes expirados

Após um determinado tempo, os ambientes que forma patrulhados devem ser patrulhados novamente, garantido assim, a confiabilidade da cobertura territorial.

R12 – Patrulha não determinística

O caminho de patrulha deve ser estocástico para que o invasores não identifiquem um padrão na patrulha, ou seja uma vulnerabilidade, e utilizem a mesma para auxiliar possíveis invasões.

R13 – Não invadir o espaço aéreo vizinho

Durante toda sua operação, o UAV deve respeitar as normas internacionais a respeito do espaço aéreo de modo a não invadir territórios vizinhos.

R14 – Conectividade

Todo UAV deve estar conectado à outros 2(dois) UAVs, ou seja, manter em seu alcance de comunicação outros 2(dois) UAVs. A estrutura da rede é ad hoc, ou seja, todas as unidades funcionarão como roteadores. A partir do conhecimento da geolocalização da estação terrestre e sua localização, o roteamento será realizado considerando a menor distância euclidiana entre os UAVs conectados e a estação terrestre.

R15 – Dinâmica dos UAV's

Embora o mecanismo de locomoção dos UAVs esteja relacionado com aerodinâmica, serão abstraídos diversos fatores ligados com aerodinâmica, no entanto, podem ser encontrados maiores detalhes em [31].

Não serão considerados os seguintes itens:

- Física da força de sustentação e seus respectivos princípios físicos como a terceira lei de Newton e o princípio de Bernulli, pressões dinâmicas de escoamento de fluido;
- Número de Reynolds, utilizado em mecânica dos fluidos para cálculo regime de escoamento de determinado fluido sobre uma superfície;
- Teoria do perfil aerodinâmico, ângulo de ataque, ângulo de incidência, etc.;
- Forças aerodinâmicas e momentos em perfis, bem como a capacidade que um perfil aerodinâmico possui de gerar essas forças;
- Centro de pressão e centro aerodinâmico do perfil;
- Formas geométricas e localizações dos componentes do UAV;
- Arrasto em aeronaves;

Serão considerados os seguintes atributos para dinâmica dos UAVs:

- Velocidade horizontal, expressa em metros/unidade de tempo;
- Velocidade máxima, (V_{max}), expressa em metros/unidade de tempo;
- Ângulo de direcionamento, de 0 a 30 graus para à direita ou esquerda. Esse ângulo influenciará a velocidade angular;
- Direção, expressa em graus. Em UAVs reais pode-se utilizar uma bússola.

R16 – Coordenação dos agentes Em suas operações, os UAVs apresentam:

- Sincronismo;
 - Atuando de forma a afastar-se se sua distância ultrapassar um limite mínimo de aproximação (ver 3.2.3);
 - Atuando de forma a se aproximar se sua distância ultrapassar um limite máximo de afastamento (ver 3.2.4).
- Cognição;
 - De maneira estocástica quando em estado de Patrulha;
 - Perseguindo e comunicando as posições dos alvos quando em estado de Rastreamento.
- Social;

Considerando o direcionamento para o qual o grupo como um todo está se movendo.

A velocidade resultante pode ser entendida de acordo com a equação A.1.

$$\vec{V}(t+1) = \vec{V}(t) * \omega + \vec{V}_{antiColisao} + \vec{V}_{comunicacao} + \vec{V}_{cognitivo} + \vec{V}_{social} \quad (A.1)$$

onde:

- $\vec{V}(t+1)$ – nova velocidade;
- $\vec{V}(t)$ – velocidade anterior;

- ω – fator de inércia;
- $\vec{V}_{antiColisao}$ – repulsão obtida a partir do sensor anti-colisão;
- $\vec{V}_{comunicacao}$ – atração obtida a partir do sensor de comunicação;
- $\vec{V}_{cognitivo}$ – calculado em função do sensor de percepção de alvos do UAV;
- \vec{V}_{social} – calculado em função do direcionamento dos vizinhos para o melhor local.

R17 – Reagrupamento

Ao perder a conectividade com o enxame, o UAV deve ir para o estado de reagrupamento.