

UNIVERSIDADE DE LISBOA  
FACULDADE DE CIÊNCIAS  
DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL



**Ciências**  
**ULisboa**

# **OPERAÇÃO DE VEÍCULOS AÉREOS NÃO TRIPULADOS EM AMBIENTE MARÍTIMO**

**Mestrado em Estatística e Investigação Operacional**  
Especialização em Investigação Operacional

Marina Colaço Ferreira

Dissertação orientada por:  
Professor Doutor António José Lopes Rodrigues







# Resumo

A utilização de Veículos Aéreos Não Tripulados (VANTs) em operações militares e civis tem aumentado significativamente comparada com os meios tripulados, em virtude da sua facilidade e versatilidade de emprego, baixos custos de operação e manutenção e redução do risco para a vida humana. A Marinha Portuguesa iniciou a integração destes meios como complemento dos atuais meios de vigilância, permitindo aumentar a sua capacidade de resposta em diferentes tipos de missões, nomeadamente na área de *surveillance* onde se destacam: missões de busca e salvamento; patrulha de área; monitorização e seguimento de navios ou embarcações com comportamento suspeito; monitorização e proteção de corredores de tráfego marítimo; segurança de portos; controlo da poluição; controlo da imigração e pesca ilegais; entre outras.

Pretendendo dar um contributo na otimização da utilização destes meios, o presente trabalho baseou-se fundamentalmente nas missões de patrulha de área, onde um VANT opera com limites espaciais bem definidos. Ao longo do tempo, surgem vários alvos com movimento próprio, que acabarão por ultrapassar os limites da área de operações, impedindo a continuação ou mesmo o início da sua monitorização.

Sendo importante que a monitorização dos alvos seja feita dentro da área de operações, é necessário determinar os que são considerados de maior importância e ordená-los segundo uma regra de prioridade.

Por simulação, foram testadas cinco regras de prioridade em oito instâncias diferentes do problema, num total de mil alvos cada, sendo parâmetros diferenciadores de cada instância: a taxa de chegadas de novos alvos à área, a sua velocidade e a duração de processamento necessária.

Verificou-se que a distância ao ponto de interseção com o alvo e a duração de processamento, associados ao grau de importância atribuído, são atributos determinantes na ordenação. Foram obtidos bons resultados com regras de prioridade que têm estes atributos em consideração, quando comparadas com outras regras de prioridade mais simples e comuns na literatura, em particular quando a taxa de chegadas de novos alvos à área de operações é elevada.

**Palavras-chave:** Veículos aéreos não tripulados, Problemas de roteamento dinâmico de veículos, Problemas dinâmicos de *Job Shop*, Regras de sequenciamento, Simulação



# **Abstract**

*The use of Unmanned Aerial Vehicles (UAVs) in military and civilian operations has significantly increased when compared with manned vehicles, because of their simplicity and versatility of use, low operational and life-cycle costs and low risk to human life. The Portuguese Navy began integrating these vehicles to complement its current capabilities, improving the readiness to perform different types of missions, especially in surveillance tasks, including: search and rescue missions; area patrolling; monitoring and tracking of suspicious vessels; monitoring and protection of maritime traffic corridors; harbor protection; pollution detection and control; illegal immigration and fishing control.*

*Wishing to contribute to optimize the usage of these vehicles, the present work was primarily based on studying area patrolling missions, considering an UAV operating within well-defined spatial limits. Along time, several targets appear, with movement, eventually reaching the area limits, thus precluding the continuation or even the start of their monitoring.*

*Taking into account that the processing of targets must be made within the area of operations, it is necessary to determine which targets should be considered more important and to order them according to a priority rule.*

*Through simulation, five priority rules were tested in eight different problem instances, with a total of one thousand targets each. The parameters that characterized each instance were: the rate of arrivals of new targets to the area, their speed and the required processing time.*

*It was found that the distance to the target intersection point and the processing time, as well as the degree of importance assigned to the target are critical attributes. Good results were achieved with priority rules that take these into consideration, when compared to other simple priority rules, especially when the target arrivals rate is high.*

**Keywords:** *Unmanned Aerial Vehicles, Dynamic Vehicle Routing Problems, Dynamic Job Shop Problems, Sequencing and Dispatching Rules, Simulation*



# Agradecimentos

Em primeiro lugar, começo por agradecer ao meu orientador, professor doutor António Rodrigues, sem o qual esta tese não teria sido possível. Ao longo de todo o processo, demonstrou total disponibilidade, partilha de conhecimento, apoio e principalmente muita paciência num período em que nem sempre as coisas pareceram fáceis.

Ao mesmo tempo, pude contar com a ajuda preciosa do João Mendonça, sem a qual certos pormenores teriam sido muito mais difíceis de resolver e para o qual vai o meu mais profundo agradecimento.

Em segundo lugar, agradeço o tempo despendido ao Contra-almirante Gouveia e Melo, Capitão-de-mar-e-guerra Alves Francisco, Capitão-tenente Madaleno Galocha e Primeiro-tenente TSN-EIO Gonçalves de Deus que me ajudaram a estabelecer os objetivos da tese e de que forma a mesma poderia dar um pequeno contributo à Marinha.

Em terceiro lugar, agradeço aos meus colegas por acreditarem em mim, sem os quais este caminho teria sido muito mais penoso.

Por último, mas não menos importante, agradeço ao meu namorado por todo o apoio e paciência nas horas difíceis e por fazer-me acreditar que realmente iria conseguir atingir uma meta que parecia tão longínqua.

A todos um grande bem haja!



# Índice

Resumo .....	i
<i>Abstract</i> .....	iii
Agradecimentos .....	v
Índice .....	vii
Lista de Figuras .....	ix
Lista de Tabelas .....	xi
<b>Capítulo 1</b> .....	<b>1</b>
<b>1. Introdução</b> .....	<b>1</b>
1.1. Enquadramento .....	1
1.2. Definição do problema .....	2
1.3. Objetivo da investigação .....	3
1.4. Estrutura do documento .....	4
<b>Capítulo 2</b> .....	<b>5</b>
<b>2. Revisão da literatura</b> .....	<b>5</b>
<b>Capítulo 3</b> .....	<b>14</b>
<b>3. Modelação do problema</b> .....	<b>14</b>
3.1. Tipos de problemas .....	15
3.2. <i>Sequencing vs Scheduling</i> .....	16
3.3. <i>Single-Machine Sequencing Model</i> .....	17
3.3.1. Conceitos genéricos .....	17
3.3.2. Problemas estocásticos .....	20
3.3.3. Problemas dinâmicos .....	20
3.4. <i>Dynamic Vehicle Routing Problems</i> .....	20
3.5. <i>Discrete Event vs Time Step Simulation</i> .....	23
3.6. Adaptação ao problema com UAVs .....	23
<b>Capítulo 4</b> .....	<b>26</b>
<b>4. Implementação</b> .....	<b>26</b>
4.1. Considerações iniciais .....	26
4.2. Principais parâmetros .....	27

4.3.	Processo de chegadas de novos alvos.....	28
4.4.	Determinação de <i>deadlines</i> .....	29
4.5.	Determinação do momento da interseção .....	30
4.6.	Atributos .....	35
4.7.	Regras de prioridade .....	36
4.8.	Medidas de desempenho .....	37
4.9.	Algoritmo .....	38
<b>Capítulo 5</b>	.....	<b>39</b>
<b>5. Experimentação</b>	.....	<b>39</b>
5.1.	Conceção de experiências .....	39
5.2.	Análise de resultados .....	40
5.2.1.	Comparação das duas posições de <i>loitering</i> .....	40
5.2.2.	Comparação das regras de prioridade .....	42
5.2.3.	Comparação com a regra FCFS .....	47
5.2.4.	Comparação de resultados com diferentes $\Delta t$ .....	47
<b>Capítulo 6</b>	.....	<b>50</b>
<b>6. Conclusão</b>	.....	<b>50</b>
6.1.	Considerações finais .....	50
6.2.	Trabalho futuro .....	53
<b>7. Bibliografia</b>	.....	<b>56</b>
<b>Apêndice A</b>	.....	<b>59</b>

# Lista de Figuras

<b>Figura 1:</b> Diagrama de <i>Gantt</i> (adaptado de [5]).	16
<b>Figura 2:</b> Esquema com algumas das principais definições nos problemas de <i>scheduling</i> .	18
<b>Figura 3:</b> Atualização da rota com chegada de novos clientes em DVRP [24].	21
<b>Figura 4:</b> Sequência de eventos em <i>Time-Step Simulation</i> (a) e <i>Discrete Event Simulation</i> (b) (adaptado de [2]).	22
<b>Figura 5:</b> Esquema da operação de um UAV dentro da área de operações.	24
<b>Figura 6:</b> Esquematização do “ <i>due date</i> ” e do “ <i>deadline</i> ” de um alvo.	25
<b>Figura 7:</b> Limites da área de operação e determinação do setor da direção de $\mathbf{v}_i$ (neste caso, S3).	28
<b>Figura 8:</b> Determinação do ponto de interseção	30
<b>Figura 9:</b> Componentes em x e em y do vector velocidade do alvo	31
<b>Figura 10:</b> Determinação do ângulo que define o rumo do UAV.	34
<b>Figura 11:</b> Esquematização da “folga” de um alvo.	37
<b>Figura 12:</b> Gráficos com percentagem de tempo de inatividade do UAV, em função da regra utilizada por instância.	41
<b>Figura 13:</b> Gráficos com percentagem de alvos investigados, em função da regra utilizada por instância.	41
<b>Figura 14:</b> Gráficos com utilidade por limite de autonomia, em função da regra utilizada por instância.	43
<b>Figura 15:</b> Gráficos com tempo médio de espera dos alvos investigados, em função da regra utilizada por instância.	44
<b>Figura 16:</b> Gráficos com o número médio de tarefas pendentes, em função da regra utilizada por instância.	44
<b>Figura 17:</b> Gráficos com a folga média <i>a posteriori</i> , em função da regra utilizada por instância.	45
<b>Figura 18:</b> Gráficos de comparação entre os melhores resultados obtidos e os resultados da regra FCFS.	48
<b>Figura A1:</b> Gráficos de comparação entre as duas posições de <i>loitering</i> para $\lambda=0.005$	62
<b>Figura A2:</b> Gráficos de comparação entre as duas posições de <i>loitering</i> para $\lambda=0.015$	63



# Lista de Tabelas

<b>Tabela 1:</b> Taxonomia dos problemas por evolução e qualidade da informação (adaptado de [24]).	15
<b>Tabela 2:</b> Regras de prioridade adequadas à medida de desempenho que se pretende otimizar.	19
<b>Tabela 3:</b> Pseudo-código do algoritmo	38
<b>Tabela 4:</b> Instâncias consideradas na experimentação.	39
<b>Tabela 5:</b> Resultados obtidos por aplicação da regra: $f(t,i)=Dur\_task_i$	46
<b>Tabela 6:</b> Resultados obtidos por aplicação da regra: $f(t,i)=setup\_time_i + Dur\_task_i$	46
<b>Tabela 7:</b> Comparação das medidas de desempenho para os diferentes $\Delta t$ s, com $\lambda=0.005$	49
<b>Tabela 8:</b> Comparação das medidas de desempenho para os diferentes $\Delta t$ s, com $\lambda=0.015$	49
<b>Tabela A1:</b> Resultados para $\Delta t=1$ e posição de <i>loitering</i> no centro da área	60
<b>Tabela A2:</b> Resultados para $\Delta t=1$ e posição de <i>loitering</i> no último alvo investigado	61
<b>Tabela A3:</b> Resultados para a regra FCFS com $\Delta t=1$	65
<b>Tabela A4:</b> Resultados para posição de <i>loitering</i> no centro área e $\Delta t=5$	66
<b>Tabela A5:</b> Resultados para posição de <i>loitering</i> no centro área e $\Delta t=15$	66
<b>Tabela A6:</b> Resultados para $\Delta t=1$ e posição de <i>loitering</i> no centro da área e com geração de posições iniciais dos alvos a uma distância máxima de 25 nm do centro da área	67



# Capítulo 1

## 1. Introdução

### 1.1. Enquadramento

Num período em que Portugal se propõe estender a sua Plataforma Continental para 3,6 milhões de km<sup>2</sup> (cerca de 40 vezes o território do Continente) e dispõe de uma área de Busca e Salvamento sob a sua responsabilidade de 5,8 milhões de km<sup>2</sup> (63 vezes a área terrestre nacional) [11], e por forma a permitir a adequada execução das funções de jurisdição e autoridade no mar, salvaguarda da vida humana, segurança marítima, proteção do ambiente e da biodiversidade, é extremamente importante que seja mantido um adequado Conhecimento Situacional Marítimo (CSM) conseguido através de um panorama correto e atualizado. Para o efeito, é indispensável a existência de meios adequados à vigilância e ao controlo dos espaços marítimos sob jurisdição e soberania nacionais [3].

Neste contexto, a Marinha Portuguesa assume um papel preponderante, pela sua capacidade e versatilidade de atuação num vasto leque de tarefas, nomeadamente no âmbito da defesa militar e apoio à política externa, segurança e autoridade do estado e desenvolvimento económico, científico e cultural [3].

Neste âmbito, a integração de Veículos Aéreos não Tripulados (VANTs), doravante denominados por UAVs (*Unmanned Aerial Vehicles*), como complemento dos atuais meios de vigilância, permite aumentar a sua capacidade de resposta. Contribuindo significativamente para o sucesso das operações, podem vir a substituir outros meios mais dispendiosos, como os helicópteros, ou menos eficientes, como embarcações pequenas, em determinados tipos de missões.

A utilização de UAVs em operações militares e civis tem aumentado significativamente, comparativamente com os meios tripulados, em virtude da sua facilidade e versatilidade de emprego, baixos custos de operação e manutenção e redução do risco para a vida humana.

Estes veículos podem ser empenhados em diferentes tipos de missões, dos quais se destacam as missões de *Intelligence, Surveillance, Target Acquisition and Reconnaissance* (ISTAR) [20].

Na área de *surveillance* em ambiente marítimo, destacam-se as missões de busca e salvamento, patrulha de área, monitorização e seguimento de navios ou embarcações com comportamento suspeito, proteção e monitorização de corredores de tráfego marítimo, segurança de portos, controlo da poluição, controlo da imigração e pesca ilegais, entre outras [28].

## **1.2. Definição do problema**

A operação de UAVs em meio marítimo encontra-se sujeita a fatores extremos. As condições ambientais e meteorológicas inerentes ao mar influenciarão os possíveis alvos situados na área de operações, afetando conseqüentemente o seu desempenho final. Para além disso, a posição dos alvos não será, em geral, estática, pelo que a dinâmica do meio envolvente será um importante fator a considerar.

Quando os alvos são estáticos (situação frequente em operações terrestres), é possível modelar o problema de roteamento de UAVs como uma adaptação do *Traveling Salesman Problem* (TSP) ou do *Vehicle Routing Problem* (VRP).

Com este tipo de problemas pretende-se minimizar o custo/distância total entre uma lista de locais a visitar, onde cada um será visitado apenas uma vez. Estes problemas são considerados *NP-Hard* (de difícil resolução em tempo polinomial), sendo necessário conceber heurísticas que consigam fazer boas aproximações à solução ótima.

No entanto, em operações marítimas, as plataformas que são alvo de investigação, navios e embarcações de menores dimensões, têm sempre associado um rumo e uma velocidade. Mesmo os que estão parados são afetados pelas condições atmosféricas, conferindo-lhes algum movimento designado por deriva, como acontece com boias de sinalização à navegação ou de pesca, manchas de poluição, embarcações de pesca em recolha de redes, entre outros.

Em caso de velocidades muito baixas ou quase nulas, o movimento poderá ser considerado desprezável, pelo que apenas serão considerados, para efeitos de investigação, alvos com velocidades superiores a 5 nós.

Para além disso, e ao longo do tempo, o número de alvos com interesse é alterável. Poderão surgir ou desaparecer alvos em diversas situações, obrigando a uma constante atualização da sequência de investigação otimizada. Essas situações podem passar por: terem saído da área limite de operação do UAV; serem classificados como não tendo importância para a investigação por outros meios de informação; ou, apenas tenham sido considerados “perdidos” pela monitorização.

Considere-se o problema, algo semelhante, do atendimento de pacientes (utentes) num hospital. Em cada momento existe uma fila de espera (conceptual) indicando a ordem de atendimento futuro dos utentes presentes nesse momento. No entanto, poderão surgir outros casos com maior prioridade ou poderá aumentar o grau de urgência ao longo do tempo dos já presentes, obrigando a que a sequência de atendimento seja recalculada.

Este tipo de problemas tem semelhanças com os problemas de *Job Shop Scheduling* (JSP), em que existem  $n$  “tarefas” (*jobs*) e  $m$  “máquinas”, e onde cada tarefa deve ser processada por uma máquina de cada vez [16]. Adaptando ao exemplo do hospital, as “tarefas” seriam os utentes e as “máquinas”, os médicos/enfermeiros.

Para o problema de roteamento de UAVs, a adaptação ao JSP poderá ser semelhante, onde as tarefas serão os alvos a investigar e as máquinas os UAVs.

### **1.3. Objetivo da investigação**

Pretende-se com esta dissertação contribuir para uma utilização otimizada destes meios, principalmente na área de *surveillance*, e identificar quais os alvos, de entre uma lista de vários, que merecem uma observação cuidada e prioritária, obtendo, quando necessário informação que mais tarde possa servir de testemunho a eventuais necessidades (ex. provas judiciais em atos ilícitos).

Cada alvo terá os seus próprios atributos, que serão os principais indicadores para a definição da prioridade de investigação de cada um.

Para alvos em ambiente marítimo, os atributos relevantes, de acordo com [26], serão os seguintes:

- Dimensão do contacto;
- Variações de velocidade;
- Proximidade entre dois contactos;
- Proximidade a terra;

- Distância à posição do UAV;
- Informação que indique que seja considerado “suspeito”.

Poderão ainda ser acrescentados aos anteriores atributos, os seguintes:

- Velocidade (mercantes a baixa velocidade ou lanchas a grandes velocidades);
- Rumo ou grandes variações de rumo;
- Distância aos limites da área de operação.

Estes atributos, isolados ou em conjunto, atribuem ao alvo um determinado grau de importância ou de criticidade, que, em conjunto com outros aspetos, como o tempo necessário à investigação, irão determinar as regras de prioridade para a ordenação.

No final, serão tidas em consideração diversas medidas de desempenho que determinarão a eficácia das regras implementadas.

#### **1.4. Estrutura do documento**

Para além do capítulo presente, de introdução, a dissertação inclui ainda os seguintes capítulos:

— no segundo capítulo é efetuada uma revisão da literatura, referenciando alguns estudos anteriores mais ou menos relacionados, por forma a reconhecer as semelhanças existentes em problemas similares e soluções possíveis;

— no terceiro capítulo é efetuada a modelação do problema com UAVs, descrevendo quais os ajustes necessários relativamente a problemas semelhantes, e identificando quais os principais parâmetros, variáveis, restrições e objetivos;

— no quarto capítulo é apresentada a proposta para resolução do problema, os algoritmos implementados e as regras de prioridade estudadas;

— no quinto capítulo é realizada a análise dos resultados experimentais obtidos por simulação, os quais são reportados com maior detalhe em Apêndice;

— por último, no sexto capítulo, apresentam-se as conclusões e perspetivas de trabalho futuro.

## Capítulo 2

### 2. Revisão da literatura

Os primeiros registos de utilização de UAVs remontam aos anos 50. Desde então diversos estudos têm sido efetuados, com o objetivo de otimizar a sua utilização. Cada vez mais os UAVs assumem um papel importante em diversas áreas de operação, seja apenas um UAV ou vários a operar em simultâneo, em ambiente terrestre ou marítimo, com alvos estáticos ou em movimento, em aplicações civis ou militares.

Os problemas com maior incidência de estudo são os que contemplam o roteamento dos UAVs. O UAV inicia o percurso a partir de um ponto, considerado como a sua plataforma de lançamento, efetua as tarefas atribuídas em diferentes localizações e regressa à plataforma, podendo ser a mesma do lançamento ou outra considerada conveniente.

Quando toda a informação necessária está disponível no início do problema (número e localização das tarefas, tempo necessário para executar cada tarefa, consumo de combustível, etc.) e não se altera durante a execução, o problema de roteamento de UAVs pode ser resolvido com modelos matemáticos já existentes, tais como os que são aplicados aos bem conhecidos *Travelling Salesman Problem* (TSP) e *Vehicle Routing Problem* (VRP).

São vários os fatores que podem ser tidos em consideração no roteamento de UAVs, quando são feitas aproximações a outros problemas já existentes, aproximando-os da realidade. A autonomia, o raio de operação máximo à estação que o controla, a altitude, as condições meteorológicas, a curvatura de passagem entre nodos, diferentes tipos de sensores, diferentes locais de lançamento e aterragem, avarias ou tempos de manutenção, são alguns dos aspetos considerados nos muitos estudos já efetuados.

Em [31] é feita uma adaptação ao TSP para o roteamento de um UAV, considerando vários depósitos que permitem o reabastecimento, podendo completar um maior número de tarefas, sem que tenha de voltar ao ponto inicial. O consumo do combustível é determinado consoante a distância entre dois nodos. Considera ainda a

questão do *Dubins' Vehicle*, onde a saída de um nodo com destino a outro é feita com uma curvatura, aumentando ligeiramente a distância entre dois pontos, o que torna o problema assimétrico (o custo do ponto A para o ponto B não é o mesmo que de B para A). O objetivo final é minimizar o consumo total de combustível.

Caso não seja possível o reabastecimento durante o percurso, é necessário determinar, de entre um conjunto de tarefas, quais as que têm maior relevância atribuindo-lhes uma bonificação, já que nem todas poderão ser realizadas. Em [12] este problema é modelado como um TSP, onde as localizações correspondem aos nodos, e a cada nodo é atribuída uma bonificação, de modo a caracterizar a importância dos alvos. A bonificação é atribuída se o nodo for visitado. Aos arcos entre os nodos são atribuídos pesos para modelar o consumo de combustível ou o tempo de voo entre dois alvos. Neste caso, o objetivo é encontrar a melhor rota que maximize a soma das bonificações, mas respeitando o consumo do combustível.

Na operação com múltiplos UAVs deverão ser consideradas outras situações. Uma delas é o facto de que nem todos os UAVs poderão ser lançados da mesma localização e, apesar de serem utilizados na mesma missão, terão de ser considerados locais de partida diferentes. Em [33] é abordada essa questão como um problema de *Multiple Depot VRP* (MDVRP). O objetivo é minimizar o número total de UAVs utilizados e a distância total de todos os percursos.

Para além de diferentes locais de lançamento, os UAVs podem ser equipados com sensores diferentes, destinados a tarefas concretas, pelo que, numa mesma missão, nem todas as tarefas podem ser atribuídas a todos os UAVs. Em [30] é adaptada esta situação a um *Heterogeneous fleet VRP* (HVRP) com o objetivo de que todos os nodos sejam visitados pelo menos por um UAV com o sensor adequado.

Ainda de acordo com [33], outra possível abordagem com adaptação ao VRP, seja utilizando apenas um UAV ou múltiplos UAVs, é o *VRP with Time Windows* (VRPTW). Existem determinadas tarefas que têm um tempo inicial a partir do qual podem ser iniciadas e/ou um limite de tempo até quando deverão estar terminadas. São exemplo as missões que requerem observação de alvos em diferentes dias, mas de preferência nas mesmas condições de visibilidade, pelo que é conveniente que sejam realizadas à mesma hora. O tempo que um alvo se mantém dentro do alcance do UAV também pode ser considerado como tendo uma janela de tempo para que possa ser investigado.

Estes são alguns exemplos na abordagem de roteamento de UAVs com modelos matemáticos já conhecidos. No entanto, na realidade, e em contraste com a

definição clássica do VRP, nem toda a informação necessária é disponibilizada desde o início, podendo variar ao longo da execução.

As operações de UAVs em certos ambientes, como em ambiente militar, estão sujeitas a constantes alterações e desvios do pré-planeado, requerendo rapidez no processo de decisão e adaptação às informações que vão surgindo. Os modelos apresentados não levam em consideração a incerteza e o dinamismo do problema, podendo levar a resultados sub-optimais ou impossíveis, nomeadamente quando a restrição da autonomia não é levada em consideração.

O consumo do combustível está sujeito a diversos fatores que farão com que o valor de consumo esperado seja diferente do real. A pensar nessas possíveis oscilações no consumo do combustível, [12] aumenta a robustez da formulação através do TSP, conforme descrito anteriormente, considerando a incerteza do consumo de combustível. Para além disso, o valor da bonificação em cada nodo também poderá variar se a prioridade do alvo também se alterar no decorrer de eventos não esperados. A robustez das soluções é obtida considerando limites inferior e superior de consumo do combustível nos arcos. O objetivo é determinar o percurso que, maximizando a soma das bonificações acumuladas, cumpra com os intervalos de consumo de combustível.

Sendo o consumo de combustível um fator de incerteza, se demorar mais tempo que o previsto num ou mais alvos, poderá ser obrigado a voltar à base antes de completar a missão. Em caso contrário, se demorar menos tempo em cada alvo, causando menor consumo de combustível, poderá ser empenhado noutros alvos que não estavam previstos. Assim, o valor máximo total de bonificação está sujeito às incertezas do consumo de combustível. Neste âmbito, outra possível abordagem para lidar com a incerteza do consumo de combustível consiste em considerar um problema de otimização estocástica com dois estágios, tal como é feito em [13]. O primeiro estágio determina o percurso ótimo e o segundo determina, baseado na incerteza dos parâmetros, quantos nodos do percurso inicial podem realmente ser visitados. Aqui o consumo de combustível segue uma distribuição de probabilidade, assumida como sendo Normal.

Em [15] é feita uma comparação entre estes dois últimos métodos e o método básico do TSP. O método de planeamento com incerteza nos arcos é o mais restritivo, considerando que segue o pior cenário, obtendo um percurso mais curto. O problema com dois estágios apresenta o percurso mais longo, contendo alvos com bonificações mais pequenas no final do percurso.

Para além da incerteza no consumo de combustível e no tempo de realização de uma tarefa, os mesmos autores [14] ainda consideram que cada alvo tem uma janela de tempo associada, contando com a incerteza do tempo necessário nos arcos e nos nodos, e o surgimento de novos alvos, que dada a sua especificidade, devem ser investigados assim que possível. Nesta situação é necessário que exista comunicação com o UAV em tempo real. Foi desenvolvido o *Maximum Coverage Stochastic Orienteering Problem with Time Windows* (MCS-OPTW), com o objetivo de determinar um percurso com o máximo de bonificação acumulado, mas direcionado para áreas onde existe grande probabilidade de surgirem novos alvos. O percurso é replaneado sempre que uma tarefa é concluída.

Um exemplo mais complexo é abordado por [1]. São consideradas as diferentes abordagens dos exemplos anteriores num só problema, considerando ainda variações ao longo do tempo em alguns dos parâmetros iniciais. Os pontos de lançamento e recolha têm movimento, variando de localização ao longo do tempo, bem como as estações de controlo de UAVs que, para além de móveis, podem controlar um número limitado e específico de UAVs ao mesmo tempo. A comunicação é perdida se os UAVs se afastarem da estação de controlo a partir de determinado raio de alcance. Os alvos têm posições fixas e fornecem bonificações se um UAV com um determinado sensor é atribuído a um alvo com necessidade desse sensor.

Foram usadas técnicas de aproximação à programação dinâmica, utilizando combinações de simulação, para modelar o movimento dos pontos de lançamento/recolha e estações de controlo, e programação matemática simples para modelar as decisões de roteamento, à semelhança dos exemplos atrás descritos. As decisões são tomadas em cada estágio, não tendo uma visão do futuro. São analisadas as posições atuais dos UAVs e as tarefas ainda por realizar, de modo a determinar as próximas atribuições.

Em [23] é feita uma abordagem ao problema onde não se sabe à partida qual a localização dos alvos, mas onde existe uma probabilidade associada a cada nodo onde é possível ou não a deteção do alvo. A própria probabilidade de deteção em cada nodo é determinada apenas com dois *time-steps* de avanço, baseada na informação de deteção ou não deteção em posições anteriores. Aqui já é considerado que os alvos não mantêm sempre a mesma posição, sendo a deteção efetivada quando o UAV e o alvo se encontram na mesma quadrícula. A velocidade dos próprios UAVs é variável entre cada nodo, dependendo da probabilidade de deteção ainda durante o trânsito. O objetivo é maximizar o número de alvos detetados num espaço de tempo pré-determinado.

Para além das missões com localizações pré-definidas ou com localizações onde a probabilidade de se detetar alvos é superior, os UAVs são muito utilizados na patrulha de áreas. Nestas situações, o objetivo principal não será otimizar a distância percorrida, mas sim otimizar a patrulha dentro de uma área, de modo a detetar qualquer alvo que surja dentro dos seus limites. Em [29] são considerados cenários com um ou dois UAVs em patrulha numa área, com alvos estáticos ou com movimento constante e define possíveis trajetórias de patrulha, de modo a que consiga cobrir o máximo de área possível, aumentando a probabilidade de deteção. São considerados os casos em que o sensor do UAV tem uma distância de deteção finita e, caso o alvo se aproxime a menos dessa distância, este é detetado. Por outro lado, considera-se que o sensor não é perfeito, seguindo uma distribuição de probabilidade para a deteção.

Também para a patrulha de áreas, [19] utiliza uma rede neuronal (do tipo *self-organizing map*, SOM) que, baseada na densidade de alvos numa determinada área, define uma rede de rotas que permite a um conjunto de UAVs tentar maximizar a deteção de alvos em movimento e posterior seguimento. A probabilidade de deteção varia consoante a densidade de alvos ou o número de UAVs empregues.

Associado ao problema de roteamento de UAVs, para busca e deteção de alvos, quando é detetado um alvo em movimento a operação a realizar pode requerer ainda algum tempo de seguimento. Considerando que a velocidade do UAV é muito superior à do alvo, é difícil fazer o seguimento apenas ajustando o rumo e velocidade à do alvo. Assim sendo, [21] aborda a problemática do seguimento de um contacto após a sua deteção, considerando que a sua posição não é estática. São considerados três possíveis padrões de seguimento: quadrado, circular integrado e *standoff* (padrão em oito, entre o alvo e o sol), assumindo que o movimento do alvo em rumo e velocidade é conhecido, e portanto determinístico. Cada um dos padrões apresenta vantagens e desvantagens, nomeadamente no que se refere à dificuldade em manter a posição sobre alvos com baixas velocidades. É no entanto possível fazer uma combinação dos padrões e adaptá-los a cada situação, considerando que um mesmo alvo pode variar várias vezes de velocidade e direção.

A maior parte dos problemas que envolvem o roteamento de veículos, em particular de UAVs, tem como objetivo determinar um conjunto de rotas que minimizem o custo de realizar um número determinado de tarefas. No entanto, na realidade, os percursos pré-determinados e otimizados podem ser sujeitos a diversos eventos que podem alterar o normal decorrer das operações. Estes eventos podem passar por avarias nos UAVs, missões abortadas por condições atmosféricas adversas ou

problemas no lançamento ou aterragem, influenciando o normal decorrer da missão e obrigando a novos planeamentos.

Em [4] é feita uma abordagem das possíveis condições adversas que podem surgir durante a execução de uma rota pré-planeada. O problema é iniciado com uma lista de tarefas a executar, sendo a atribuição aos UAVs feita de acordo com a sua exequibilidade, considerando distância, duração da tarefa e tempo de regressar à base. Das tarefas executáveis é escolhida a que tiver maior prioridade por atribuição de pesos. Cada vez que é atribuída uma tarefa, todas as restantes são novamente avaliadas e novamente escolhida a próxima tarefa a atribuir, até que todas as tarefas da lista estejam atribuídas aos UAVs. Depois de todas as tarefas atribuídas, surgem naturalmente as rotas pré-planeadas que serão testadas quanto à sua robustez, considerando diversas fases do planeamento até que todas as tarefas sejam finalizadas e os UAVs possam regressar à base. A cada fase e respetivo evento foi atribuída uma probabilidade de ocorrência, comprovando no final se a rota pré-planeada é ou não robusta. Caso não seja, todas as tarefas atribuídas a essa rota terão de ser novamente reordenadas e atribuídas.

Grande parte dos estudos efetuados no roteamento de UAVs utiliza o problema clássico de *Vehicle Routing* como base para determinar qual o melhor percurso, de modo a minimizar a distância/consumo do combustível. No entanto, mais do que noutro ambiente, as operações de UAVs em ambiente marítimo estão sujeitas a grandes alterações, como o surgimento ou desaparecimento de alvos, ou alterações de posicionamento, obrigando a constante replaneamento, que pode requerer grande esforço computacional.

Em [10] é proposta uma nova abordagem, considerando uma aproximação baseada em “filas de espera”, onde o roteamento é baseado no objetivo de minimizar, por exemplo, o tempo de espera que uma tarefa está na fila até ser finalizada. Os novos eventos são gerados com localizações e tempos de chegada aleatórios, mantendo-se ativos até chegada de um UAV e fim da missão atribuída.

Na teoria de filas de espera [32], as tarefas vão surgindo ao longo do tempo com uma determinada distribuição e respetivo tempo de necessidade de intervenção, estando disponíveis por um ou mais servidores. Os tempos de chegada, bem como os tempos de intervenção, podem ser determinísticos (com intervalos de tempo iguais ou iguais tempos de intervenção), ou de acordo com uma determinada distribuição de probabilidade, sendo necessário ordenar as tarefas de acordo com um critério que satisfaça as necessidades.

Em [27] é tida em consideração uma lista de critérios que dão uma informação qualitativa da urgência de investigação de um alvo. Esses critérios passam por condições meteorológicas (efeito do vento, visibilidade), tempo de serviço aleatório, surgimento de alvos com maior prioridade, restrições de rota, alvos com diferentes prioridades (velocidade, pessoas em perigo, distância a terra, distância ao UAV, distância aos limites da área), aspetos de segurança, missões abortadas.

Atribuindo um peso a cada parâmetro, é determinada a prioridade de cada alvo. Alvos com maior prioridade serão investigados primeiro. Considerando que muitos dos parâmetros variam ao longo do tempo, aquando da conclusão da investigação de um alvo todos os pesos são novamente recalculados e reavaliadas as prioridades, de modo a determinar qual o alvo seguinte a ser investigado.

Na mesma linha de pensamento do estudo anterior, [26] faz uma abordagem idêntica, no entanto utiliza uma lista de critérios diferente, de modo a determinar um grau de suspeita, incluindo: dimensão do barco, alteração na velocidade, posições de *rendez-vous* entre duas embarcações, perto de ilhas, perto de costa, distância ao UAV, ou informações externas que indiquem que o alvo é suspeito. Estes critérios poderão ter mais ou menos peso consoante o cenário onde são aplicados. São considerados quatro cenários diferentes: tráfico de droga em mar alto, tráfico de droga perto de costa, imigração ilegal durante o dia e imigração ilegal durante a noite. Para determinar os pesos, consoante o cenário, é utilizado o método *Analytic Hierarchy Process* (AHP). Para avaliar cada alvo, e estabelecer a sua prioridade, é utilizado o *Preference Ranking Organisation MeTHod for Enrichment Evaluation* (PROMETHEE). Esta análise tem em consideração que as prioridades dos alvos podem alterar à medida que são reavaliados.

Em [8] é considerada a sequenciação de tarefas de busca, para um único operador do UAV com restrições de tempo. São assumidas  $N$  tarefas heterogéneas (com diferentes bonificações), definidas pelo tempo total para completar a tarefa e pelo instante em que a tarefa fica disponível para ser iniciada. As tarefas são sequenciadas de modo a maximizar o total acumulado de bonificação num tempo finito. Os UAVs que não estejam empenhados podem orbitar junto da localização da próxima tarefa até que esta fique disponível.

Num caso especial onde todas as tarefas e UAVs estejam disponíveis, o problema poderá ser formulado como um problema “saco-mochila”. No caso mais geral em que os UAVs têm de chegar às tarefas em tempos diferentes (quando as tarefas ficam disponíveis), e assumindo que só temos um operador, o problema pode

ser formulado como um problema do tipo *Job-Shop Scheduling* (JSP). Enquanto no JSP os objetivos mais comuns são minimizar o tempo total para a conclusão das tarefas, o atraso total, ou o número total de tarefas atrasadas, aqui o objetivo será maximizar o total de bonificação.

As regras de ordenação podem ser várias, desde as mais simples às mais complexas, e são, em geral, baseadas em atributos como a ordem de chegada, a duração da execução da tarefa, ou a janela de tempo disponível. Para avaliação da regra aplicada, são tidas em consideração determinadas medidas de desempenho, como a média de tarefas realizadas num dado espaço de tempo, a média dos tempos de espera das tarefas ou a média dos atrasos na execução.

Em [7] é analisado um sistema com vários UAVs prontos a serem empregues em alvos com posições fixas e prioridades múltiplas. São aplicadas regras de *Dispatching* e *Loitering*, de modo a assegurar um rápido e eficaz empenhamento em alvos com alta prioridade e gestão efetiva de alvos com prioridade média. Com algum conhecimento *a priori* das zonas com maior probabilidade de surgirem novos eventos, é feito um adequado posicionamento dos UAVs, de modo a darem melhor resposta a potenciais alvos. As regras de *dispatching* testadas foram as seguintes: *First Available UAV*, *Closest Available UAV* e *Closest to be Available UAV*. Caso todos os UAVs estejam ocupados, os alvos vão sendo reordenados de acordo com outras regras. Assim que um UAV esteja livre, segue para o primeiro alvo da lista. As regras aplicadas aos alvos foram a *First Come-First Served* e a *Shortest Travel Time or Distance*. Outra opção será ordená-los numa fila de espera de acordo com a sua prioridade (ex.: 1, 2 ou 3).

Existem atualmente vários estudos em *Job Shop Scheduling*, que exploram a utilização de diferentes regras de prioridade. As regras mais simples têm em consideração um único atributo dos alvos, como o instante de chegada, a duração do processamento ou o prazo para conclusão desse processamento. Regras menos simples baseiam-se em combinações de vários atributos, procurando atender melhor à medida de desempenho que se quer otimizar [6] e [25].

Em [9] e [22] são abordados problemas dinâmicos (*dynamic job shop scheduling*), onde as tarefas surgem ao longo do tempo.

A maior parte dos estudos encontrados na literatura adaptam os problemas com UAVs a problemas já conhecidos como o VRP ou o JSP. Apesar das semelhanças, há que considerar que neste caso poderá ter que existir uma adaptação

conjunta dos dois problemas, nomeadamente das suas vertentes de *Dynamic VRP* e *Dynamic JSP*, de modo a aproximar à realidade dos UAVs.

Nesta tese, abordámos o problema considerando o surgimento de novos alvos ao longo do tempo em posições aleatórias. Cada alvo terá uma posição, rumo e velocidade próprios, conferindo-lhes movimento, e um peso associado que determina a sua criticidade. No entanto, para além deste, outros atributos serão definidos ou avaliados, possibilitando a aplicação de diferentes regras de prioridade.

Foi desenvolvido um programa para experimentação, por simulação, de diferentes regras de prioridade face a diversas instâncias do problema. Cada instância foi estudada através da geração de um número relativamente grande de alvos, e consequente simulação das tarefas de processamento de grande parte desses alvos, com o objectivo de avaliar diversas medidas de desempenho.

No processo de simulação, o estado do sistema é reavaliado a intervalos de tempo fixos, e muito pequenos, de modo a permitir continuamente equacionar qual a opção a tomar no momento presente, incluindo a possível decisão de interromper uma operação em curso.

## Capítulo 3

### 3. Modelação do problema

O dinamismo das operações com UAVs, em ambiente marítimo, requer que sejam aplicados procedimentos expeditos que consigam dar uma resposta eficiente às alterações que vão registando ao longo do tempo — por exemplo, chegadas aleatórias de novos alvos e alterações nos atributos desses alvos, nomeadamente a sua posição ou o tempo disponível para os processar. É, por isso, necessário considerar heurísticas simples, capazes de adaptar as decisões às circunstâncias, em tempo real.

Os UAVs estão sujeitos a restrições de autonomia, pelo que, para além da distância ou custo total, que se pretende minimizar, outras medidas poderão ser levadas em consideração.

Por motivos de segurança, sempre que é planeado executar uma operação com UAVs, é necessário dar conhecimento às autoridades competentes [17] e restringir a sua operação a uma área de operações bem definida, de modo a que não haja conflito com outros meios aéreos. Assim, estando a operação restringida a limites geográficos bem definidos, a investigação de alvos só poderá ser executada dentro desses limites.

Não sendo possível investigar todos os alvos dentro da área, visto terem movimento e existir um momento em que acabarão por sair, é necessário ordená-los por ordem de prioridade, de acordo com o objetivo que se pretende otimizar.

Neste capítulo, far-se-á uma descrição das principais características presentes nos problemas de sequenciamento e *dispatching*, e das formas possíveis de resolução.

Para melhor reconhecimento das designações mais comuns na literatura, optou-se por deixar alguns termos sem tradução, tentando evitar assim ambiguidades.

### 3.1. Tipos de problemas

De acordo com [24], dependendo da evolução e qualidade da informação, os problemas de otimização podem ser de natureza “estática” ou “dinâmica”, e “determinísticos” ou “estocásticos”. A Tabela 1 identifica as quatro categorias em que os problemas se podem inserir.

A evolução da informação indica que a informação disponível pode variar durante a execução do problema, como, por exemplo, os atributos das entidades presentes num sistema. A qualidade da informação indica a possível incerteza nos dados disponíveis, como os instantes de chegada das entidades a um sistema.

Quando todas as tarefas e seus atributos são conhecidos no início do problema, não sofrendo qualquer alteração ao longo do tempo, o problema é classificado como sendo estático e determinístico. Se algum dos atributos não for conhecido com exatidão, o problema será considerado estocástico. Geralmente, estes atributos seguem distribuições de probabilidade, sendo possível estimar os seus valores esperados.

Nos problemas dinâmicos, o número total e tipo de tarefas e seus atributos são desconhecidos no início do problema, sendo revelados dinamicamente ao longo do tempo. À medida que os dados vão sendo disponibilizados, são necessários novos planeamentos, de modo a contemplá-los.

**Tabela 1:** Taxonomia dos problemas por evolução e qualidade da informação (adaptado de [24]).

		Qualidade da informação	
		Dados determinísticos	Dados estocásticos
Evolução da Informação	Dados conhecidos antecipadamente	Estático e determinístico	Estático e estocástico
	Dados variáveis ao longo do tempo	Dinâmico e determinístico	Dinâmico e estocástico

### 3.2. Sequencing vs Scheduling

O sequenciamento (*sequencing*) consiste em determinar a ordem de processamento de operações pendentes. O *scheduling* (agendamento, escalonamento ou calendarização) consiste em definir a hora de início e a hora de término de cada operação.

Para além da capacidade/disponibilidade dos agentes executores, estes problemas podem estar, também, sujeitos a outras restrições como a necessidade de precedência que alguns eventos possam ter relativamente a outros ou o limite de tempo até quando deverão estar concluídos: “*deadline*” (prazo taxativo).

Estes problemas têm geralmente como objetivo minimizar o tempo total de execução das tarefas (“*makespan*”), minimizar o atraso total ou maximizar o número de tarefas realizadas num determinado período de tempo.

De forma a facilitar a visualização da ordenação e hora de início das operações para um determinado agente executor, existem modelos gráficos especialmente desenhados para o efeito, com uma visão global de todo o planeamento, permitindo maior facilidade para efetuar possíveis alterações. Um dos modelos gráficos mais comuns e simples é o diagrama de *Gantt* (Figura 1). Tarefas que necessitem de ser processadas por mais que um agente são distribuídas entre eles, só podendo ser iniciada a operação por um quando terminada a operação pelo anterior. O facto de as operações não terem iguais tempos de processamento pode conduzir naturalmente a que existam tempos de inatividade nos agentes (zonas a cinzento).

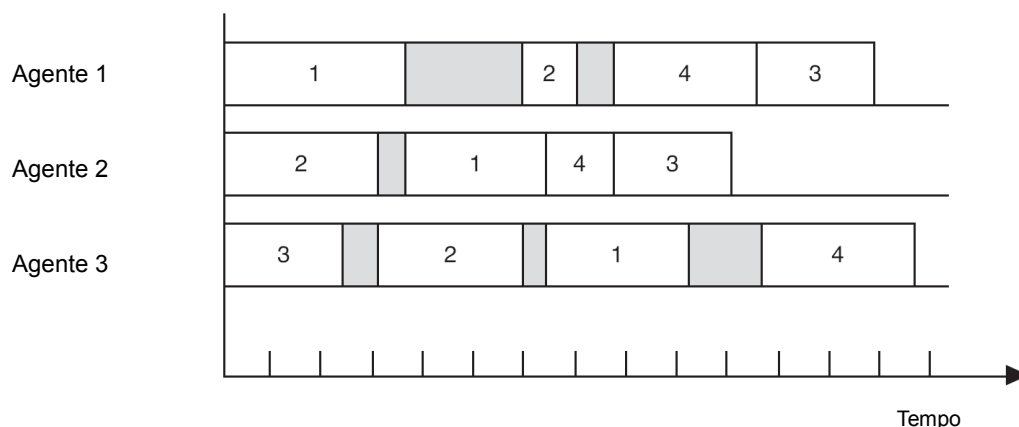


Figura 1: Diagrama de *Gantt* (adaptado de [5]).

Os principais desenvolvimentos nos problemas de *scheduling* surgiram com a necessidade de otimização de recursos na área da manufaturação. É, por isso, comum encontrar diversas designações dessa área quando se abordam esses problemas. Os agentes executores são denominados por “máquinas” que processam as “tarefas” (“*jobs*” em inglês). Para concluir uma tarefa, poderão ter de ser efetuadas várias operações, na mesma ou em máquinas diferentes, daí os problemas serem comumente designados por “*job shop problems*”.

### **3.3. Single-Machine Sequencing Model**

#### **3.3.1. Conceitos genéricos**

O problema mais simples de *job-shop scheduling* é o que considera apenas uma máquina, uma lista de tarefas completa (caso estático), e os respetivos atributos com valores perfeitamente conhecidos (caso determinístico).

Estes problemas cumprem as seguintes condições [5]:

- C1: Existem “ $n$ ” tarefas simultaneamente disponíveis para processamento no instante inicial (zero);
- C2: A máquina só pode processar uma tarefa de cada vez;
- C3: O tempo que uma tarefa demora a chegar à máquina é independente da ordem de sequenciamento e está incluído no tempo de processamento;
- C4: Todos os dados das tarefas são determinísticos e disponibilizados desde o início;
- C5: A máquina está sempre presente no sistema e nunca avaria;
- C6: A máquina nunca está parada, desde que existam tarefas à espera;
- C7: A execução de uma operação nunca é interrompida.

Os atributos das tarefas, considerados como o *input* do problema, incluem os seguintes (os dois primeiros são obrigatórios, os outros são opcionais):

- *Processing time* ( $p_i$ ): Tempo necessário à execução da tarefa  $i$
- *Release date* ( $r_i$ ): Instante a partir do qual a tarefa  $i$  está disponível para iniciar processamento
- *Due date* ( $d_i$ ): Instante até ao qual deve desejavelmente a tarefa  $i$  estar concluída (*soft constraint*)
- *Weight* ( $w_i$ ): Peso ou grau de importância atribuído à tarefa  $i$

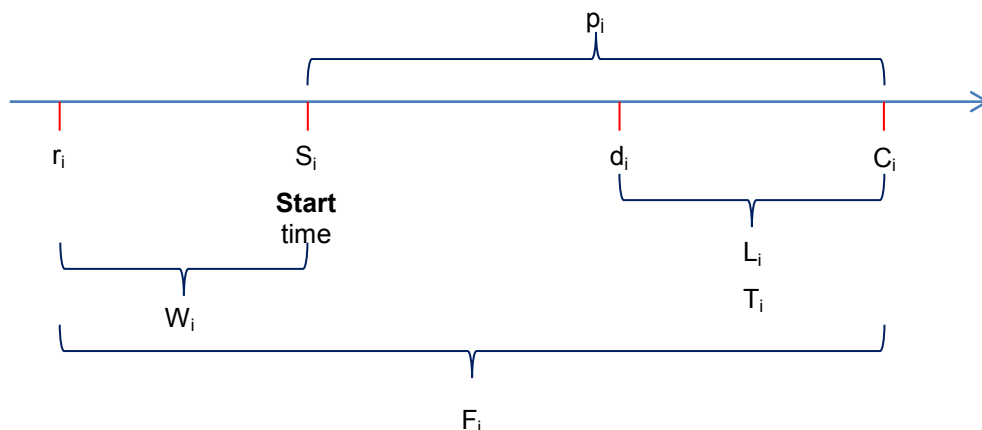
A partir de uma solução de calendarização (*schedule*), é comum avaliar as seguintes estatísticas, com base nas quais podem ser definidas diversas medidas de desempenho:

- *Completion time* ( $C_i$ ): Instante em que a tarefa  $i$  é terminada
- *Flowtime* ( $F_i$ ): Tempo de permanência da tarefa  $i$  no sistema:  $F_i = C_i - r_i$
- *Waiting time* ( $W_i$ ) Tempo de espera da tarefa  $i$ :  $W_i = F_i - p_i$
- *Lateness* ( $L_i$ ): Diferença de tempo entre o instante em que a tarefa  $i$  está terminada e o seu prazo de referência:  
 $L_i = C_i - d_i \geq 0$
- *Tardiness* ( $T_i$ ): A tarefa  $i$  é considerada “tardia” se a conclusão do seu processamento ultrapassa o prazo de referência:  
 $T_i = \max \{0, L_i\}$
- Inatividade ( $I_j$ ) Tempo total que a máquina  $j$  está inativa

Na Figura 2 é apresentado um esquema elucidativo.

Com base nestas estatísticas podem ser definidos, por agregação, diversas medidas de desempenho, incluindo:

- Tempo médio de espera das tarefas:  $\bar{W} = (\sum_{i=1}^n W_i)/n$
- Atraso médio:  $\bar{L} = (\sum_{i=1}^n L_i)/n$



**Figura 2:** Esquema com algumas das principais definições nos problemas de *scheduling*.

**Tabela 2:** Regras de prioridade adequadas à medida de desempenho que se pretende otimizar.

Medida de desempenho	Regra de prioridade
$(\min)\bar{F}, \bar{L}, \bar{W}$ ou $\bar{C}$	SPT
$(\min)\bar{F}_w, \bar{L}_w, \bar{W}_w$ ou $\bar{C}_w$	SWPT
$(\min)L_{max}$	EDD
$(\max)L_{min}$	MST

Caso tenham sido definidos pesos,  $w_i$ , será preferível calcular médias ponderadas. Também é comum considerar máximos — em especial, é habitual procurar minimizar  $C_{max} = \max C_i$ , isto é, o instante de conclusão da última operação.

Dependendo da medida de desempenho que se pretende otimizar, podem ser aplicadas várias regras de prioridade.

Entre as regras de prioridade relativamente simples mais comuns estão:

- FCFS (*First Come First Served*): Ordena as tarefas por ordem de chegada ao sistema. ( $r_{[1]} \leq r_{[2]} \leq \dots \leq r_{[n]}$ )
- SPT (*shortest processing time*): Ordena as tarefas por ordem não crescente das durações. ( $p_{[1]} \leq p_{[2]} \leq \dots \leq p_{[n]}$ )
- SWPT (*shortest weighted processing time*): ( $p_{[1]}/w_{[1]} \leq p_{[2]}/w_{[2]} \leq \dots \leq p_{[n]}/w_{[n]}$ )
- EDD (*earliest due date*): Ordena as tarefas por ordem não decrescente dos “due dates”. ( $d_{[1]} \leq d_{[2]} \leq \dots \leq d_{[n]}$ )  
 Considera as folgas\*\* das tarefas no instante 0 e dá prioridade às tarefas com menor folga.  
 ( $d_{[1]} - p_{[1]} \leq d_{[2]} - p_{[2]} \leq \dots \leq d_{[n]} - p_{[n]}$ )
- MST (*minimum slack time*): Obs.: Em geral, a folga (*slack*) de uma tarefa é avaliada num qualquer instante  $t$  por  $d_i - t - p_i$ .

A Tabela 2 indica quais as regras de prioridade a aplicar, dependendo da medida de desempenho que se pretende otimizar (*vide* [16]).

### **3.3.2. Problemas estocásticos**

Na secção anterior foi descrito o problema de sequenciamento mais básico, onde se considera apenas uma máquina e todos os atributos são determinísticos. No entanto, se for permitida relaxação na condição C4, os atributos  $p_j$ ,  $d_j$  e  $w_j$  poderão ser incertos, dando origem a problemas de natureza estocástica.

As medidas de desempenho para os problemas estocásticos serão as mesmas, no entanto, considerando a incerteza em alguns dos parâmetros, o objetivo será minimizar os valores esperados das medidas de desempenho, por exemplo,  $E[\bar{W}]$ . Em contrapartida, pode fazer-se uma aproximação dos problemas estocásticos aos determinísticos, utilizando os valores médios dos atributos.

### **3.3.3. Problemas dinâmicos**

Se a relaxação incidir nas condições C1 e C3, as tarefas têm diferentes tempos de chegada ao sistema e o tempo que demoram a chegar à máquina depende da tarefa que foi realizada imediatamente antes.

Quando os tempos de chegada ao sistema são diferentes e aleatórios, o problema passa a ser dinâmico. À medida que novas tarefas vão surgindo, é necessário avaliá-las e ordená-las conjuntamente com as restantes tarefas pendentes. Esta avaliação pode ser feita apenas quando é terminada a tarefa em curso, ou no momento em que a nova tarefa chega ao sistema. Caso tenha prioridade superior, a tarefa em curso poderá, eventualmente, ser interrompida, considerando a relaxação das condições C6 e C7.

Se uma tarefa for interrompida, o seu  $p_i$  é atualizado de acordo com o tempo de processamento que já decorreu. No processo de reordenação, será o novo  $p_i$  que será tido em consideração.

## **3.4. Dynamic Vehicle Routing Problems**

Os problemas de *Vehicle Routing* também podem necessitar de ajustes em tempo real, não só com informação do próximo cliente a servir, sujeito a chegadas de novos clientes durante o decorrer da operação, mas também da atualização da procura. Para fazer face a estas questões surgiram os *Dynamic Vehicle Routing Problems* (DVRP). Nestes problemas, ao contrário dos JSP, o tempo despendido em cada cliente é desprezável ou incluído no tempo de viagem até ao local.

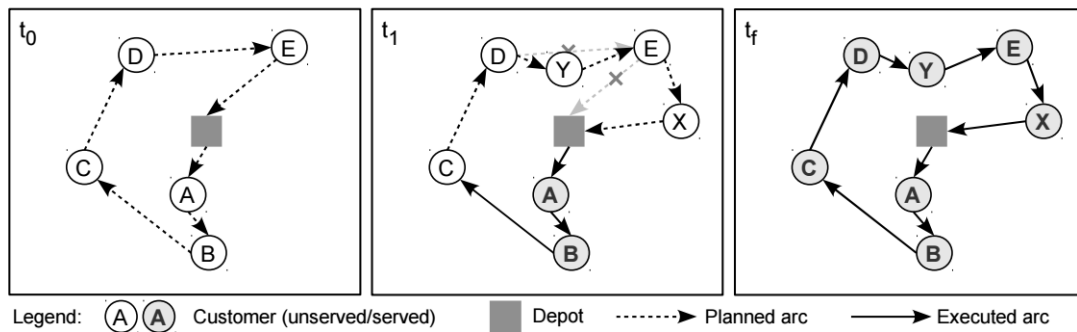
Na Figura 3 é ilustrado o dinamismo que se pode encontrar nestes problemas. Antes do veículo sair do depósito (instante  $t_0$ ), é planeada a rota que deve seguir para

os clientes conhecidos (A, B, C, D e E). No entanto, durante a operação (instante  $t_1$ ), surgem dois novos clientes (X e Y) e a rota inicial deve ser ajustada, de modo a contemplá-los. No final (instante  $t_f$ ), todos os clientes foram incluídos na rota do veículo. Este exemplo demonstra os ajustes necessários nas rotas em problemas dinâmicos, sendo para isso necessário comunicações em tempo real com o veículo.

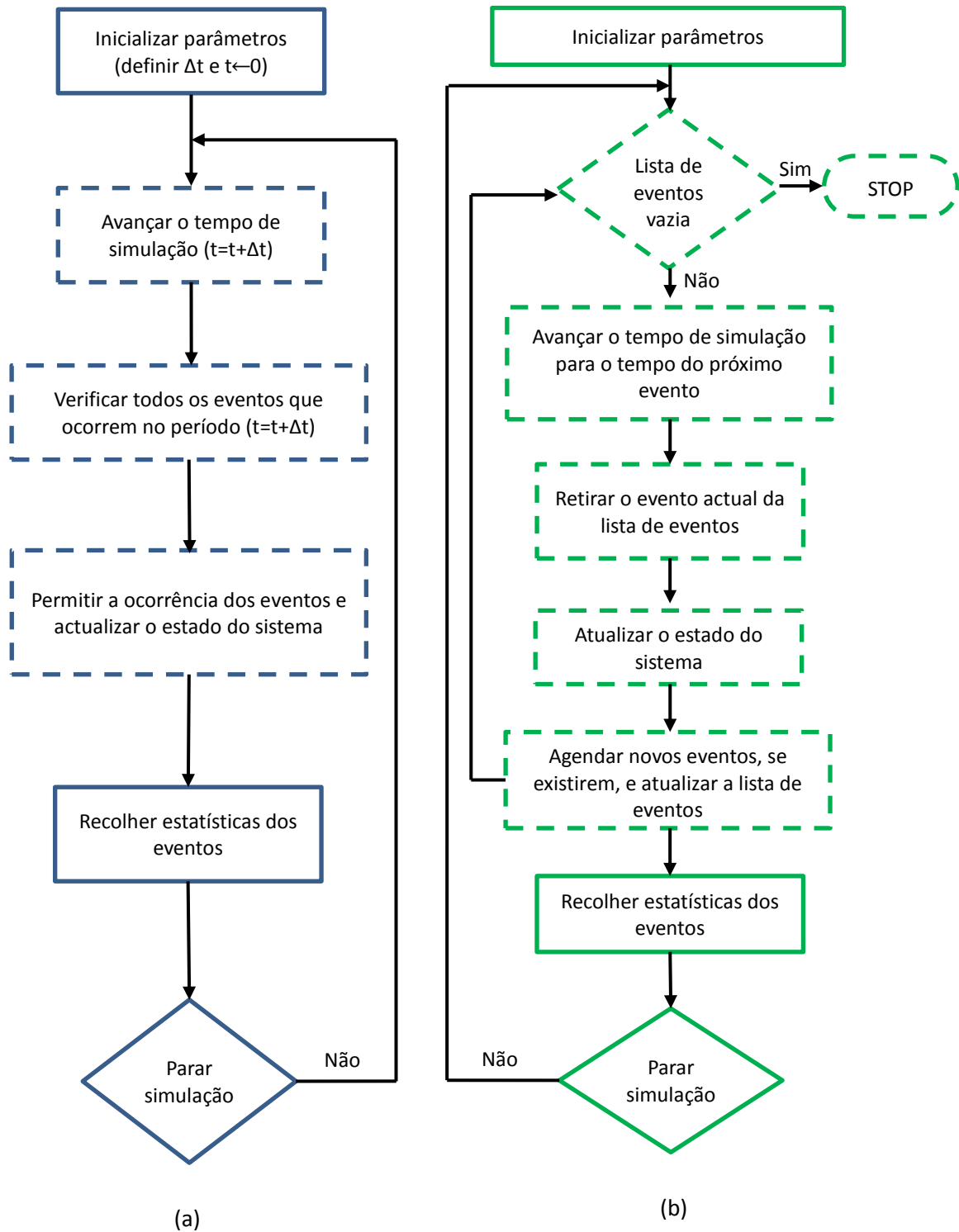
Enquanto que nos problemas estáticos o objetivo é usualmente minimizar o custo ou comprimento da rota, nos problemas dinâmicos são introduzidos novos objetivos, como a maximização do número de clientes satisfeitos num dado intervalo de tempo ou a minimização do tempo médio de espera dos clientes.

À semelhança dos JSP, também nos DVRP podem ser aplicadas regras de prioridade, de modo a determinar qual o próximo cliente a servir. Estas regras podem passar pela escolha do cliente mais próximo (*nearest neighbor*), pela ordem de chegada dos pedidos dos clientes (FCFS) ou pelo maior lucro que se obtém [18].

Se for considerada a relaxação da condição C3 da secção 3.3.1 e se a posição das tarefas não for estática, o problema poderá ser assimétrico: ir de i para j poderá não demorar o mesmo se, mais tarde, se quiser voltar de j para i.



**Figura 3:** Atualização da rota com chegada de novos clientes em DVRP [24].



**Figura 4:** Sequência de eventos em *Time-Step Simulation* (a) e *Discrete Event Simulation* (b) (adaptado de [2]).

### **3.5. Discrete Event vs Time Step Simulation**

Como já referido anteriormente, quando chega uma nova tarefa ao sistema com os seus atributos próprios, é necessário fazer uma nova reordenação de todas as tarefas, de modo a contemplá-la. O novo reordenamento poderá ser feito em dois momentos distintos: imediatamente após a chegada da tarefa ao sistema, permitindo avaliar se a nova tarefa tem maior urgência ou apenas após conclusão da tarefa em curso.

Mesmo que não surjam novas tarefas, as que já estão no sistema podem alterar os seus atributos, existindo escalada na sua prioridade. Assim sendo, um sistema poderá fazer avaliações das tarefas em intervalos bem definidos com um  $\Delta t$  fixo (*time-step simulation*) ou só após ocorrência de um evento (*discrete event*).

Na Figura 4 são apresentados esquemas da sequência dos dois tipos de simulação.

### **3.6. Adaptação ao problema com UAVs**

Considerando agora o problema onde a máquina/veículo é o UAV e as tarefas/clientes os alvos a investigar, reconhece-se a existência de semelhanças tanto com os *Dynamic Job Shop Problems (DJSP)* como com os *Dynamic Vehicle Routing Problems (DVRP)*.

Na literatura, a maioria dos estudos com JSP despreza o tempo de afetação da tarefa à máquina, *setup time*, ou inclui-o no tempo de processamento,  $p_i$ . Por outro lado, nos estudos com o DVRP, o tempo de processamento ou tempo nos nodos é desprezável e o *setup time* assume o papel predominante.

Nos problemas com UAVs, tanto o *setup time* como o tempo de processamento são igualmente expressivos, sendo a relaxação das condições C3 e C6, particularmente relevante. Existe, por isso, necessidade de se adaptar e procurar novas regras de prioridade para estes problemas específicos.

O UAV é caracterizado pelo tipo (de acordo com o sensor com que é equipado), o ponto de lançamento e recolha, autonomia, velocidade e raio máximo de operação da estação de controlo (podendo coincidir com o ponto de lançamento ou não).

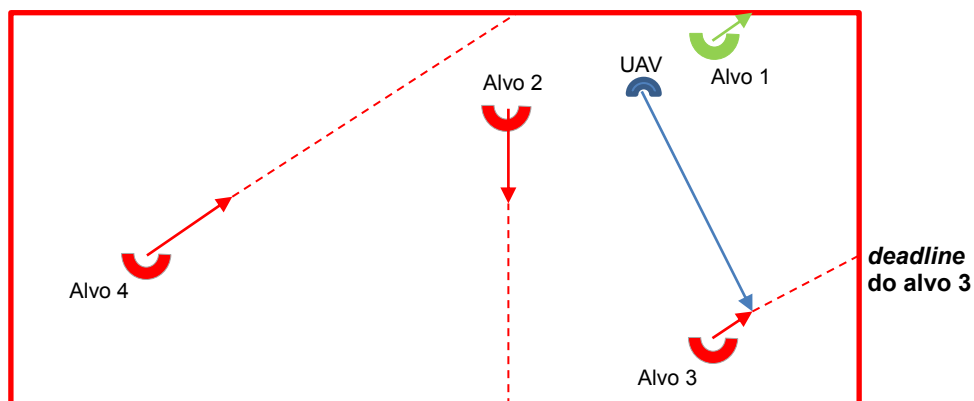
Os alvos são caracterizados pela posição, rumo, velocidade, grau de criticidade, distância ao limite da área, tempo de processamento necessário e hora de chegada ao sistema.

Na Figura 5, é esquematizada a operação de um UAV dentro da área de operações. No *time step* imediatamente após terminada a investigação do “Alvo 1”, que se prepara para sair da área, é efetuada escolha do alvo seguinte, dependendo da regra de prioridade. É crucial que a investigação seja efetuada (iniciada e terminada) dentro dos limites da área.

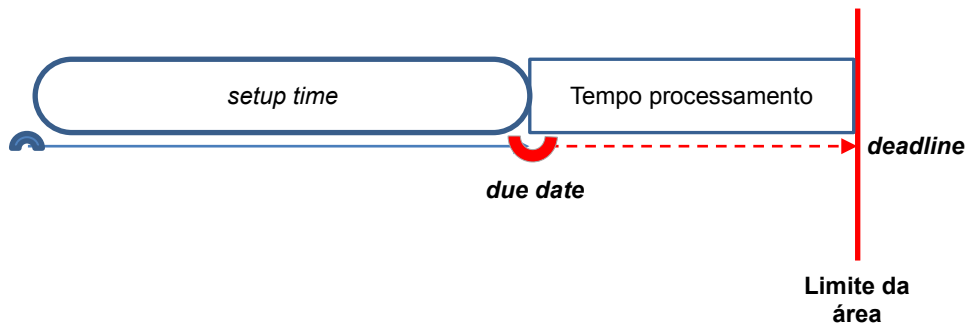
Cada alvo tem um prazo (*deadline*) que coincide com o instante em que passa para fora dos limites da área. Neste problema, o *due date*,  $d_i$ , de cada alvo é o instante limite em que o UAV tem de alcançar o alvo, de modo a que consiga, em princípio, completar a sua investigação, isto é, face ao que se prevê para o movimento futuro do alvo e para a duração da operação. Na Figura 6 é exemplificada a definição do *due date* e do *deadline*.

Basearemos o estudo no problema mais simples, considerando apenas um UAV numa área de operações, tentando, tanto quanto possível, investigar o maior número de alvos que surgem na área ao longo do tempo.

Os atributos dos alvos serão na sua maioria determinísticos; no entanto, o instante de entrada no sistema e as posições iniciais serão estocásticas e, ao contrário do que é considerado na maior parte dos estudos no âmbito do DJSP e do DVRP, os alvos terão movimento, alterando a sua posição ao longo do tempo, o que terá consequências diretas no *setup time* (tempo de voo até intersecção) estimado em cada instante.



**Figura 5:** Esquema da operação de um UAV dentro da área de operações.



**Figura 6:** Esquematização do “*due date*” e do “*deadline*” de um alvo.

## Capítulo 4

### 4. Implementação

#### 4.1. Considerações iniciais

A implementação do simulador das heurísticas de sequenciação foi feita em MATLAB R2013a.

Considerou-se um UAV em missão de *surveillance* numa área de operações pré-definida, com a missão de investigar vários alvos em movimento e com diferentes atributos. As características do UAV foram baseadas no modelo AR3 da *Tekever*, em testes na Marinha Portuguesa, e os alvos têm uma posição inicial gerada aleatoriamente dentro dos limites da área de operação.

Apenas um alvo estará no sistema no instante  $t=0$ , surgindo novos alvos ao longo do tempo, de acordo com um processo de chegadas *Poisson*.

Os alvos têm rumo e velocidade constantes ao longo de toda a simulação e só podem ser investigados dentro da área de operação. Caso contrário, após ultrapassarem os limites da área são considerados como “não investigados”. O tempo que cada alvo demora a atingir um dos limites da área determinará o seu *deadline*, instante a partir do qual já não poderão ser processados. A cada alvo gerado é também atribuído um peso, que determinará a sua importância ou grau de criticidade, e a duração da investigação necessária.

Durante a simulação, poderão existir momentos em que não existam alvos em espera para processamento. Assim sendo, o UAV ficará a pairar numa posição pré-definida, à espera que surjam novos alvos (posição de *loitering*). Foram consideradas duas alternativas para esse posicionamento, ou reposicionamento:

- o ponto central da área de operações, ou
- o local em que terminou a investigação do último alvo.

Toda a informação será reavaliada em intervalos de tempo definidos por um  $\Delta t$  fixo, e em cada instante é escolhido o alvo a ser investigado ou confirmada/modificada a escolha do alvo antes feita, dependendo do objetivo que se pretende atingir.

Apesar do maior esforço computacional, o facto de se reavaliar o sistema a intervalos de tempo fixos (*time steps*) pequenos — para que a perda de precisão por arredondamentos seja negligenciável —, permite uma maior agilidade na recondução do UAV. Através de pequenos ajustes, a interceção com um alvo pode então ser mais exata. Esta questão seria ainda mais relevante se fossem considerados como estocásticos o rumo e velocidade dos alvos.

Para além disso, a análise do sistema a cada *time step* permitiria, em geral, fazer previsões baseadas num histórico recente. As decisões são tipicamente melhores quando apoiadas em previsões que tirem partido de observações tão recentes quanto possível.

## **4.2. Principais parâmetros**

A área de operações foi definida como uma área retangular, com limite nas coordenadas cartesianas  $[x_{\min} \ x_{\max} \ y_{\min} \ y_{\max}]$  (Figura 7), considerando  $x_{\min}=-60$ ,  $x_{\max}=60$ ,  $y_{\min}=-40$ ,  $y_{\max}=40$ .

Os dados iniciais considerados foram os seguintes:

### **UAV:**

- Velocidade máxima: 65 kt;
- Autonomia: até 10 h = 600 min;
- Posição inicial no ponto (0,0);

### **Alvos:**

*Atributos estáticos:*

- Número de identificação:  $\{1, \dots, n\}$ ;
- Instante de entrada: instante de tempo em que o alvo entra na área de operações.
- Posição inicial:  $(x, y) \in [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$ ;
- Rumo ( $\text{rumo}_i$ ):  $\{000, \dots, 360\}^\circ$ ;
- Velocidade ( $\|v_i\|$ ):  $\{5, \dots, 30\}$  nós;
- Duração da tarefa ( $\text{Dur\_task}_i$ ):  $\{10, \dots, 50\}$  min;
- Peso ( $w_i$ ):  $\{1, \dots, 20\}$ ;

- *Deadline* (deadline): instante em que se presume que o alvo sairá da área de operações;

*Atributos dinâmicos:*

- Posição em cada instante  $t$ .

### 4.3. Processo de chegadas de novos alvos

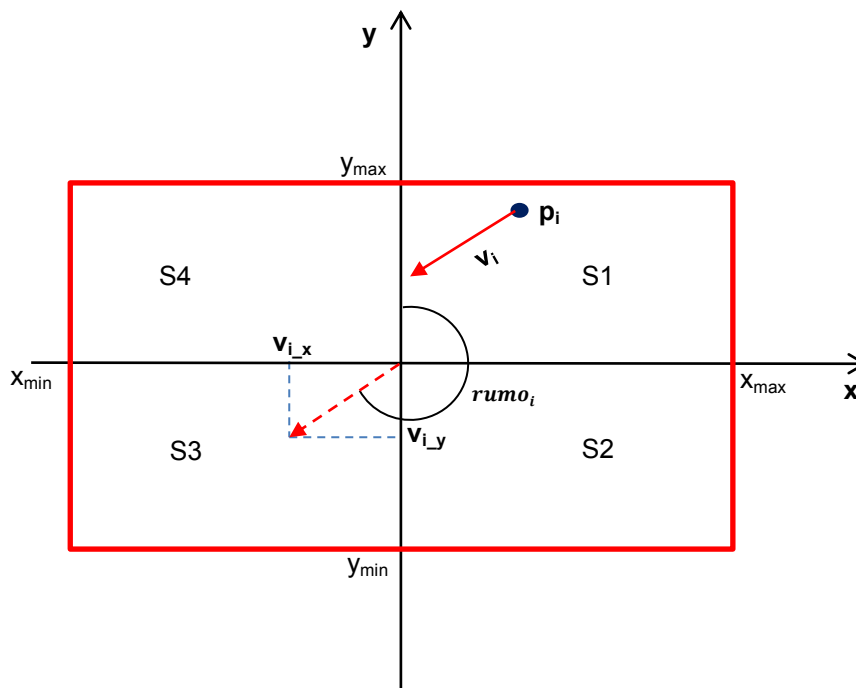
Considera-se que o surgimento de novos alvos segue um processo *Poisson*, ou seja, a probabilidade de que existam  $k$  surgimentos num determinado intervalo de tempo é dada pela expressão:

$$f(k; \lambda) = \frac{e^{-\lambda} \lambda^k}{k!}$$

onde  $\lambda$  indica a frequência com que surgem novos alvos num determinado período de tempo. Se num período de 10 min apenas surgir um novo alvo,

$$\lambda = \frac{1}{10}$$

A posição inicial do alvo é gerada de acordo com uma distribuição Uniforme para cada uma das coordenadas geográficas.



**Figura 7:** Limites da área de operação e determinação do setor da direção de  $v_i$  (neste caso, S3).

#### 4.4. Determinação de deadlines

A distância percorrida por um alvo num dado intervalo de tempo de comprimento  $T$  é definida por  $v_i * T$ . Designando por  $p_i(t)$  a posição do alvo num instante  $t$ , então, a partir da posição no instante atual,  $t_0$ :

$$\begin{cases} p_{i_x}(t) = p_{i_x}(t_0) + v_i * (t - t_0) * \sin(\text{rumo}_{\text{alvo}}) \\ p_{i_y}(t) = p_{i_y}(t_0) + v_i * (t - t_0) * \cos(\text{rumo}_{\text{alvo}}) \end{cases}$$

Então, o instante  $t$  em que o alvo atingirá um dos limites da área de operações, dependendo da direção do seu vetor  $\mathbf{v}_i$ , definida pelo rumo $_i$ , pode ser determinado substituindo  $p_{i_x}(t)$  e  $p_{i_y}(t)$  pelos limites  $x_{\min}$  ou  $x_{\max}$  e  $y_{\min}$  ou  $y_{\max}$ , respetivamente.

Assim, dependendo do setor da direção do vetor  $\mathbf{v}_i$  (Figura 7) o instante  $t$  em que o alvo atinge o limite da área é:

No setor S1:

$$t_{x_{\max}} = \frac{(x_{\max} - p_{i_x}(t_0))}{(v_{e_i} * \sin(\text{rumo}_i))} \text{ e } t_{y_{\max}} = \frac{(y_{\max} - p_{i_y}(t_0))}{(v_{e_i} * \cos(\text{rumo}_i))}$$

No setor S2:

$$t_{x_{\max}} = \frac{(x_{\max} - p_{i_x}(t_0))}{(v_{e_i} * \sin(\text{rumo}_i))} \text{ e } t_{y_{\min}} = \frac{(y_{\min} - p_{i_y}(t_0))}{(v_{e_i} * \cos(\text{rumo}_i))}$$

No setor S3:

$$t_{x_{\min}} = \frac{(x_{\min} - p_{i_x}(t_0))}{(v_{e_i} * \sin(\text{rumo}_i))} \text{ e } t_{y_{\min}} = \frac{(y_{\min} - p_{i_y}(t_0))}{(v_{e_i} * \cos(\text{rumo}_i))}$$

No setor S4:

$$t_{x_{\min}} = \frac{(x_{\min} - p_{i_x}(t_0))}{(v_{e_i} * \sin(\text{rumo}_i))} \text{ e } t_{y_{\max}} = \frac{(y_{\max} - p_{i_y}(t_0))}{(v_{e_i} * \cos(\text{rumo}_i))}$$

O limite, em  $x$  ou  $y$ , que for atingido em menor tempo, dependendo do setor, determinará o *deadline* do alvo:

$$\text{deadline}_i = \min \{ t_{x_{\min}}, t_{x_{\max}}, t_{y_{\min}}, t_{y_{\max}} \}$$

Todos os alvos que ultrapassem os limites da área de operações não poderão ser investigados, então deve-se ainda subtrair o tempo de duração da investigação, de modo a que esta ocorra ainda dentro da área. Assim, o instante limite em que o UAV deve chegar ao alvo, de modo a que consiga investigá-lo dentro dos limites da área, determinará o seu *due date* e será definido por:

$$due\_date_i = deadline_i - Dur\_task_i$$

#### 4.5. Determinação do momento da interseção

Considerando que os alvos não são estáticos, a partir do momento em que é decidido investigar um alvo, há que ter em atenção que a posição do alvo diferente da atual quando o UAV conseguir intersectá-lo.

Conforme ilustrado na Figura 8, o alvo parte de uma posição  $p_i$  com uma velocidade  $v_i$  e direção  $rumo_i$ . O UAV parte da posição  $p_{UAV}$  com uma velocidade  $v_{UAV}$ . Importa determinar o ponto  $p_{int}$  onde o UAV consegue intersestar o alvo, o tempo  $T$  que demora para o fazer e a direção,  $rumo_{UAV}$ , que o UAV tem de tomar para atingir essa posição no menor tempo possível.

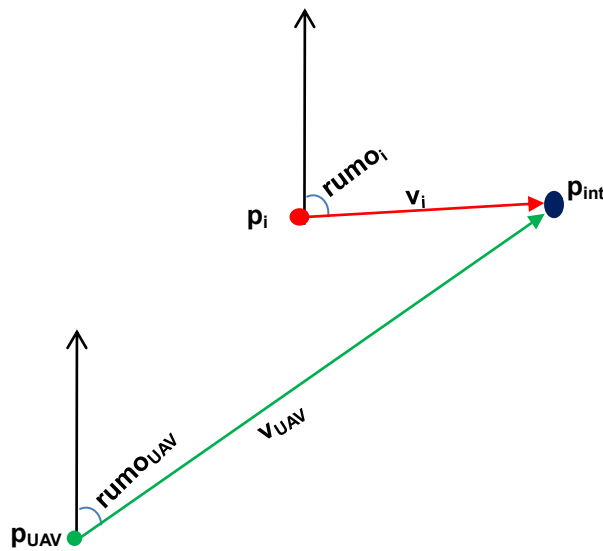


Figura 8: Determinação do ponto de interseção

No plano cartesiano os vetores velocidade dos alvos e do UAV são caracterizados pelas suas componentes nos eixos dos xx e dos yy (Figura 9), onde  $\text{rumo}_i$  ou  $\text{rumo}_{UAV}$  é o ângulo da direção do vector e:

$$v_{i_x} = v_i * \sin(\text{rumo}_i)$$

$$v_{i_y} = v_i * \cos(\text{rumo}_i)$$

$$v_{UAV_x} = v_{UAV} * \sin(\text{rumo}_{UAV})$$

$$v_{UAV_y} = v_{UAV} * \cos(\text{rumo}_{UAV})$$

Pode-se então definir os vetores  $\mathbf{v}_{UAV}$  e  $\mathbf{v}_i$ , os pontos de partida de cada um e o ponto de interseção através das suas componentes cartesianas:

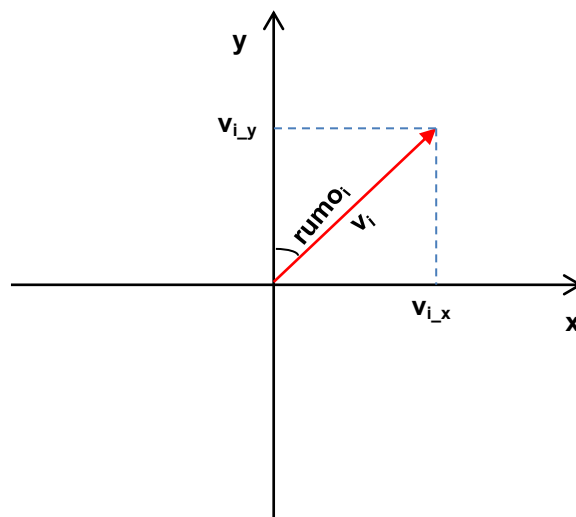
$$\mathbf{v}_{UAV} = [v_{UAV_x}, v_{UAV_y}]$$

$$\mathbf{v}_i = [v_{i_x}, v_{i_y}]$$

$$\mathbf{p}_i = (p_{i_x}, p_{i_y})$$

$$\mathbf{p}_{UAV} = (p_{UAV_x}, p_{UAV_y})$$

$$\mathbf{p}_{int} = (p_{int_x}, p_{int_y})$$



**Figura 9:** Componentes em x e em y do vector velocidade do alvo

$\mathbf{p}_{int}$  pode ser determinado através de  $\mathbf{p}_i$  e da distância percorrida pelo alvo ou através de  $\mathbf{p}_{UAV}$  e da distância percorrida pelo UAV:

$$\mathbf{p}_{int} = \mathbf{p}_i + d_{percorrida_i}$$

ou 
$$\mathbf{p}_{int} = \mathbf{p}_{UAV} + d_{percorrida_{UAV}}$$

onde:

$$d_{percorrida_i} = \mathbf{v}_i * t$$

$$d_{percorrida_{UAV}} = \mathbf{v}_{UAV} * t$$

e “t” é o tempo que decorreu.

Designando o tempo necessário até chegar a  $\mathbf{p}_{int}$  por T, tem-se:

$$\mathbf{p}_{int} = \mathbf{p}_i + \mathbf{v}_i * T$$

ou 
$$\mathbf{p}_{int} = \mathbf{p}_{UAV} + \mathbf{v}_{UAV} * T$$

donde

$$\mathbf{p}_i + \mathbf{v}_i * T = \mathbf{p}_{UAV} + \mathbf{v}_{UAV} * T$$

Expandindo, em função das componentes x e y:

$$\begin{cases} p_{i_x} + v_{i_x} * T = p_{UAV_x} + v_{UAV_x} * T \\ p_{i_y} + v_{i_y} * T = p_{UAV_y} + v_{UAV_y} * T \end{cases} \quad (1)$$

Como só temos a magnitude de  $\mathbf{v}_{UAV}$ , que designamos por  $V_{UAV}$ , a direção do UAV pode ser qualquer. Pelo teorema de Pitágoras, o quadrado da norma ou magnitude de um vector é igual à soma dos quadrados das componentes ortogonais que o definem:  $V_{UAV}^2 = v_{UAV_x}^2 + v_{UAV_y}^2$ . Assim, a partir da equação (1),

$$v_{UAV_x} = (p_{i_x} - p_{UAV_x}) * \frac{1}{T} + v_{i_x} \quad (2)$$

$$v_{UAV_y} = (p_{i_y} - p_{UAV_y}) * \frac{1}{T} + v_{i_y}$$

tem-se:

$$\left( (p_{i_x} - p_{UAV_x}) * \frac{1}{T} + v_{i_x} \right)^2 + \left( (p_{i_y} - p_{UAV_y}) * \frac{1}{T} + v_{i_y} \right)^2 = V_{UAV}^2$$

Desenvolvendo, chega-se à seguinte equação de 2º grau:

$$\left( (p_{i_x} - p_{UAV_x})^2 + (p_{i_y} - p_{UAV_y})^2 \right) * \frac{1}{T^2} + 2 * \left( (p_{i_x} - p_{UAV_x}) * v_{i_x} + (p_{i_y} - p_{UAV_y}) * v_{i_y} \right) * \frac{1}{T} + (v_{i_x}^2 + v_{i_y}^2 - V_{UAV}^2) = 0$$

Fazendo  $k = \frac{1}{T}$  pode-se aplicar a fórmula resolvente de equações de 2º grau, onde:

$$a = (p_{i_x} - p_{UAV_x})^2 + (p_{i_y} - p_{UAV_y})^2$$

$$b = 2 * \left( (p_{i_x} - p_{UAV_x}) * v_{i_x} + (p_{i_y} - p_{UAV_y}) * v_{i_y} \right)$$

$$c = (v_{i_x}^2 + v_{i_y}^2 - V_{UAV}^2)$$

e

$$k = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \Leftrightarrow T = \frac{2a}{-b \pm \sqrt{b^2 - 4ac}}$$

Desde que  $V_{UAV} > ||v_i||$  e dado que só existe interesse na solução em que  $T > 0$ , fica-se só com:

$$T = \frac{2a}{-b + \sqrt{b^2 - 4ac}}$$

obtendo-se assim o tempo que o alvo e o UAV demoram até à interseção.

De modo a que o UAV se dirija ao ponto de interseção  $p_{int}$  é necessário determinar a sua direção, através do ângulo  $\theta_{UAV}$ .

$\theta$  é o ângulo que  $\overrightarrow{v_{UAV}}$  faz com o eixo dos xx e pode ser calculado como:

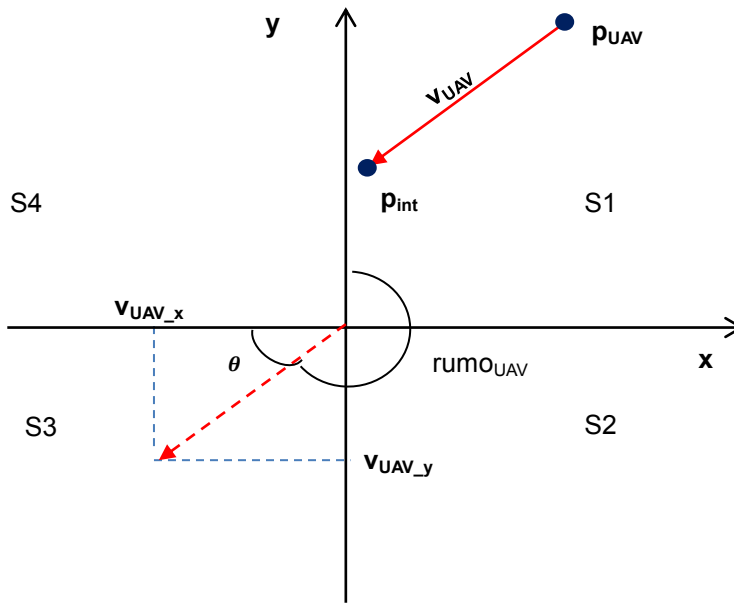
$$\theta = \left\| \tan^{-1} \left( \frac{v_{UAV_y}}{v_{UAV_x}} \right) \right\|$$

onde  $v_{UAV_x}$  e  $v_{UAV_y}$  são determinados pela equação (2).

É exceção a situação em que  $v_{UAV_x} = 0$ , onde:

$$\begin{cases} \text{se } v_{UAV_y} > 0, & \theta = 90 \\ \text{se } v_{UAV_y} < 0, & \theta = -90 \end{cases}$$

Como se considera apenas o módulo de  $\theta$ , fica  $\theta = 90$ , para as duas situações de exceção.



**Figura 10:** Determinação do ângulo que define o rumo do UAV.

Dependendo do sector em que se encontra  $\mathbf{v}_{UAV}$  (Figura 10) — sendo que o rumo se contabiliza a partir do Norte e admitindo que o eixo dos  $yy$  aponta sempre para Norte —, o ângulo que vector  $\mathbf{v}_{UAV}$  faz com o eixo dos  $yy$  é:

No setor 1:  $rumo_{UAV} = 90 - \theta$

No setor 2:  $rumo_{UAV} = 90 + \theta$

No setor 3:  $rumo_{UAV} = 360 - (90 + \theta)$

No setor 4:  $rumo_{UAV} = 360 - (90 - \theta)$

Sendo o sistema atualizado e reavaliado a cada *time step* com intervalos fixos, as posições de interseção do UAV com o alvo poderão não coincidir num instante múltiplo do *time step*. Concretamente, no instante  $t$  podem estar ainda em aproximação a  $\mathbf{p}_{int}$  e no instante  $t + \Delta t$  já terem ultrapassado esse ponto. Assim, num qualquer instante  $t$ , se o UAV e o alvo estiverem a uma distância inferior a uma distância  $D$  pré-definida, considera-se que já atingiram  $\mathbf{p}_{int}$  e pode dar-se início à fase de investigação.

Essa distância  $D$  é determinada por:

$$D = d_{percorrida_i} + d_{percorrida_{UAV}}$$

com base nas distâncias que o alvo ou o UAV percorrem em cada  $\Delta t$ :

$$d_{percorrida_i} = (vel_i * \Delta t)/60$$

$$d_{percorrida_{UAV}} = (vel_{UAV} * \Delta t)/60$$

## 4.6. Atributos

A cada *time step*,  $\Delta t$ , todos os alvos presentes no sistema são avaliados. Os que terminaram o seu processamento saem do sistema e entram no conjunto dos “investigados”; para os restantes verifica-se a exequibilidade de investigação. Os alvos em que o seu *due date* já não permita que o UAV chegue a tempo de os intercepar e concluir a sua investigação são incluídos no conjunto dos “*não\_investigados*”.

Os restantes são incluídos no conjunto *S*, de onde será escolhido o próximo alvo a investigar, de acordo com a regra de prioridade estabelecida.

Os atributos e conjuntos utilizados foram, então, os seguintes:

- *contactos* Número total de alvos no sistema em cada instante *t*
- *peso<sub>i</sub>* Valor de importância atribuído ao alvo *i*
- *Dur\_task<sub>i</sub>* Tempo de processamento do alvo *i*
- *deadline<sub>i</sub>* Instante em que o alvo *i* atinge o limite da área de operações
- *due\_date<sub>i</sub>* Instante limite para que o UAV intercepte o alvo *i*
- *S* Conjunto dos alvos exequíveis de processar
- *investigados* Conjunto com indicadores, 0 ou 1, alvos: investigados
- *nao\_investigados* Conjunto dos alvos não investigados
- *setup\_time<sub>i</sub>* Tempo de voo (mínimo) até interseção do alvo *i*
- *tempo\_entrada<sub>i</sub>* Instante de entrada do alvo *i*
- *tempo\_espera<sub>i</sub>* Intervalo de tempo desde que o alvo *i* entra no sistema até que é inicializado o seu processamento
- *C\_time<sub>i</sub>* Instante de conclusão da investigação do alvo *i*

## 4.7. Regras de prioridade

As regras de prioridade servem para escolher, num dado instante  $t$ , qual o alvo  $j$  a processar, através da avaliação de um indicador,  $f$ , baseado num ou mais atributos:

$$j = \underset{i \in S}{\operatorname{argmin}} f(t, i)$$

Listamos a seguir as 5 regras de prioridade consideradas no estudo experimental realizado. Estas regras foram escolhidas a título meramente ilustrativo, dado que muitas outras regras, mais ou menos complexas, poderiam ter sido consideradas.

- Seleção do alvo para o qual a posição de interseção é a mais próxima da posição do UAV:

$$\text{Regra 1: } f(t, i) = \text{setup\_time}_i$$

- Seleção do alvo que será o primeiro a sair da área:

$$\text{Regra 2: } f(t, i) = \text{deadline}_i$$

- Seleção do alvo com maior grau de criticidade:

$$\text{Regra 3: } f(t, i) = \frac{1}{\text{peso}_i}$$

- Seleção do alvo com maior grau de criticidade, conjugando com os tempos de processamento e distância do UAV ao seu ponto de interseção:

$$\text{Regra 4: } f(t, i) = \frac{\text{setup\_time}_i + \text{Dur\_task}_i}{\text{peso}_i}$$

- Seleção do alvo com menor “folga”:

$$\text{Regra 5: } f(t, i) = (\text{deadline}_i - t) - (\text{setup\_time}_i + \text{Dur\_task}_i)$$

Refira-se que a “folga” de um alvo é determinada pelo intervalo de tempo que o UAV ainda tem para iniciar o *setup time*, de modo a que consiga cumprir com o seu *due date* (Figura 11).

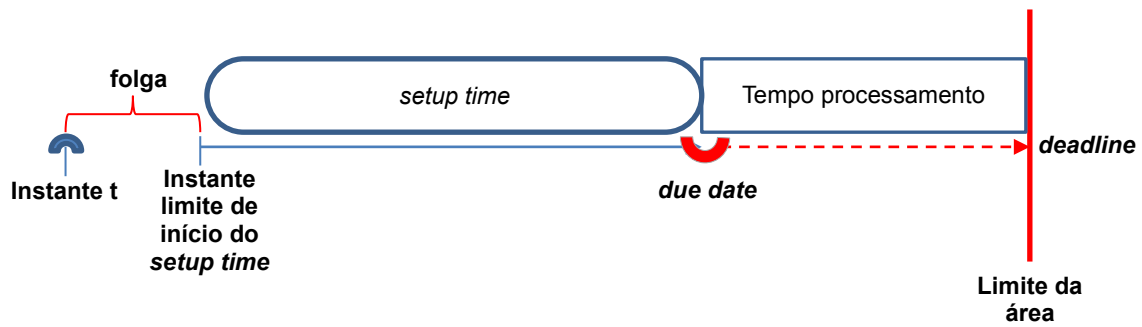


Figura 11: Esquematização da “folga” de um alvo.

#### 4.8. Medidas de desempenho

As medidas de desempenho avaliadas, relativas ao UAV ou aos alvos, foram as seguintes:

- **Percentagem de Tempo Inativo:** Percentagem de tempo (relativamente à duração da simulação) em que o UAV se encontra em viagem para a posição de *loitering* ou a pairar à espera que surjam novos alvos; inversamente, o UAV está “ativo” se estiver em rota de interseção de um alvo (*setup time*) ou se estiver a investigar um alvo;
- **Percentagem de alvos investigados;**
- **Tempo médio de espera dos alvos investigados;**
- **Folga média *a posteriori*:** Média das folgas restantes, calculadas só para os alvos investigados, aquando da conclusão das tarefas;
- **Utilidade média por unidade de tempo:** Soma dos pesos dos alvos investigados e normalizada para a unidade de tempo correspondente a um ciclo de autonomia do UAV, que neste caso serão 600 min;
- **Média de tarefas pendentes:** De modo a determinar a sobrecarga do UAV, a cada  $T=10*\Delta t$  foi registado o número de alvos ainda no sistema à espera de processamento. No final da simulação foi calculada a média desses valores.

## 4.9. Algoritmo

O algoritmo de simulação implementado é, essencialmente, o que se encontra na Tabela 3, onde o critério de paragem exigiu a geração de um número suficientemente grande de alvos (concretamente, 1000) e que todos os alvos presentes no sistema fossem processados.

---

**Tabela 3:** Pseudo-código do algoritmo

---

$t \leftarrow 0$

**WHILE** critério de paragem não for satisfeito **DO**

$t \leftarrow t + \Delta t$

- Eventualmente, gerar a entrada de novos alvos
- Atualizar as posições de todos os alvos
- Atualizar S (incluindo a possível inserção de novos alvos e a exclusão de alvos entretanto inexecutáveis)

**IF** S não vazio **THEN**

- Determinar o alvo  $i \in S$  a investigar, baseado na regra escolhida
- Iniciar percurso até ao alvo  $i$ ;  
ou, se já estiver sob investigação, continuar a segui-lo;  
ou, ainda, terminar essa investigação, caso aplicável.

**ELSE**

- Dirigir o UAV para a posição de *loitering* (ou mantê-lo nessa posição)

**END IF**

**END WHILE**

---

---

## Capítulo 5

### 5. Experimentação

#### 5.1. Conceção de experiências

Em apêndice são apresentados os principais resultados obtidos na experimentação. Foram testadas 8 instâncias, cada uma envolvendo o processamento de um total de 1000 alvos. As instâncias 1 e 5 foram geradas com taxas de chegada de, em média, um alvo a cada três horas (distribuição *Poisson* com  $\lambda=0.005$ ) e de um alvo por hora (distribuição *Poisson* com  $\lambda=0.015$ ), respetivamente. As restantes instâncias foram geradas como derivações das primeiras, por modificação dos valores da velocidade e da duração das tarefas de referência,  $v_{iRef}$  e  $Dur\_task_{iRef}$ .

Num computador com processador *Intel Core2 Solo* de 1.4GHz e com 4GB de memória RAM, a média de tempo de processamento de cada instância foi na ordem dos 200 seg para  $\lambda=0.005$  e 113 seg para  $\lambda=0.015$ .

Na Tabela 4 é apresentado um resumo das instâncias consideradas.

**Tabela 4:** Instâncias consideradas na experimentação.

Taxa de chegadas	Instâncias
$\lambda=0.005$	<u>1</u> : $v_{i1} = v_{iRef}, Dur\_task_{i1} = Dur\_task_{iRef}$ <u>2</u> : $v_{i2} = 0.75 * v_{iRef}, Dur\_task_{i2} = Dur\_task_{iRef}$ <u>3</u> : $v_{i3} = 1.25 * v_{iRef}, Dur\_task_{i3} = Dur\_task_{iRef}$ <u>4</u> : $v_{i4} = v_{iRef}, Dur\_task_{i4} = 1.25 * Dur\_task_{iRef}$
$\lambda=0.015$	<u>5</u> : $v_{i5} = v_{iRef}, Dur\_task_{i5} = Dur\_task_{iRef}$ <u>6</u> : $v_{i6} = 0.75 * v_{iRef}, Dur\_task_{i6} = Dur\_task_{iRef}$ <u>7</u> : $v_{i7} = 1.25 * v_{iRef}, Dur\_task_{i7} = Dur\_task_{iRef}$ <u>8</u> : $v_{i8} = v_{iRef}, Dur\_task_{i8} = 1.25 * Dur\_task_{iRef}$

Para cada instância foram testadas as 5 regras, considerando duas posições diferentes de *loitering*: o centro da área — neste caso, o ponto (0,0) — e a posição do último alvo investigado.

Como *benchmark* foi também considerada a regra FCFS (*First Come First Served*), comum em problemas de *Job Scheduling*, que ordena os alvos por ordem de chegada.

Foram também testados diferentes valores de  $\Delta t$ :  $\Delta t = 1$ ,  $\Delta t = 5$  e  $\Delta t = 15$  (minutos). O  $t_{\text{final}}$  médio em cada simulação rondou os 197600 min para  $\lambda=0.005$  e os 65000 min para  $\lambda=0.015$ , dependendo do tempo de entrada do último alvo no sistema.

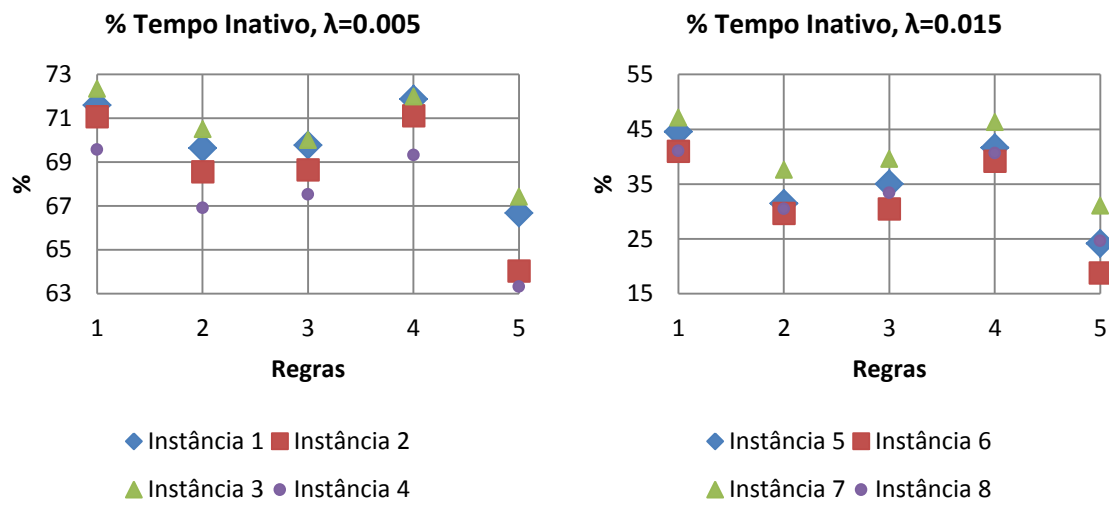
## **5.2. Análise de resultados**

### **5.2.1. Comparação das duas posições de loitering**

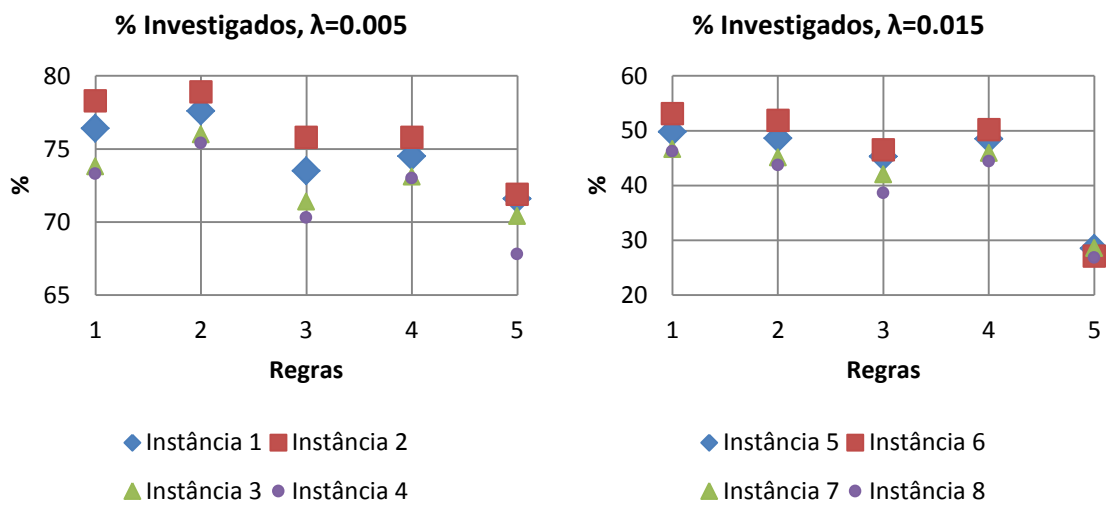
Analisando os resultados reportados nas Tabelas A1 e A2 e nas Figuras A1 e A2, em apêndice, verifica-se que entre as duas posições de *loitering* consideradas, a que apresenta melhores resultados em todas as medidas de desempenho é a do centro da área, nomeadamente nas instâncias com taxa de chegadas mais baixa, já que o tempo de permanência na posição de *loitering* é maior (%tempo inativo).

Dado que a geração de novos alvos é feita com distribuição Uniforme em toda a área de operações, o centro da área acaba por ser a posição mais adequada para fazer face a novas chegadas.

Por este motivo, nas próximas secções faz-se uma análise dos resultados considerando apenas os que foram obtidos com a posição de *loitering* no centro da área.



**Figura 12:** Gráficos com percentagem de tempo de inatividade do UAV, em função da regra utilizada por instância.



**Figura 13:** Gráficos com percentagem de alvos investigados, em função da regra utilizada por instância.

### **5.2.2. Comparação das regras de prioridade**

De acordo com as Figuras 12 e 13, o tempo de inatividade do UAV situa-se entre os 63% e os 73% para  $\lambda=0.005$  e entre os 18% e os 48% para  $\lambda=0.015$ , com percentagem de alvos investigados entre os 67% e os 79% e entre os 26% e os 54% respetivamente.

Considerando estes valores e para os dados do problema, quando a taxa de chegadas é de 1 alvo a cada 3 horas ( $\lambda=0.005$ ) a utilização de apenas um UAV é suficiente, visto a sua percentagem de inatividade ser superior a 60%. No entanto, quando a taxa aumenta para aproximadamente 1 alvo/hora é necessário escolher criteriosamente a regra implementada, de modo a que a percentagem de alvos investigados ronde os 50%.

O facto das posições iniciais dos alvos serem geradas aleatoriamente por toda a área de operações implica que existam alvos muito próximos dos limites e, conseqüentemente, se o seu rumo for em direção à fronteira da área de operações os *deadlines* são muito baixos, não permitindo que o UAV os alcance.

Se forem criadas instâncias onde as posições iniciais dos alvos se concentram a uma distância limite do centro da área, a percentagem de alvos investigados aumenta para valores entre os 71% e os 94% para  $\lambda=0.005$  e entre os 57% e os 76% para  $\lambda=0.015$ , não considerando para já a regra 5, conforme se verifica na Tabela A6 em apêndice.

Para a taxa de chegadas mais baixa, a regra 5 apresenta resultados aceitáveis, apesar de serem piores que as restantes regras. No entanto, quando a taxa de chegadas aumenta ( $\lambda=0.015$ ), a percentagem de alvos investigados piora, não sendo superior a 21%.

Em termos de comportamento das regras para estas duas medidas de desempenho, salientam-se as regras 1 e 4 para a otimização do tempo de inatividade, qualquer que seja a taxa de chegadas de novos alvos. Para a percentagem de alvos investigados destacam-se as regras 1 e 2 para  $\lambda=0.005$  e com resultados muito próximos as regras 1, 2 e 4 para  $\lambda=0.015$ .

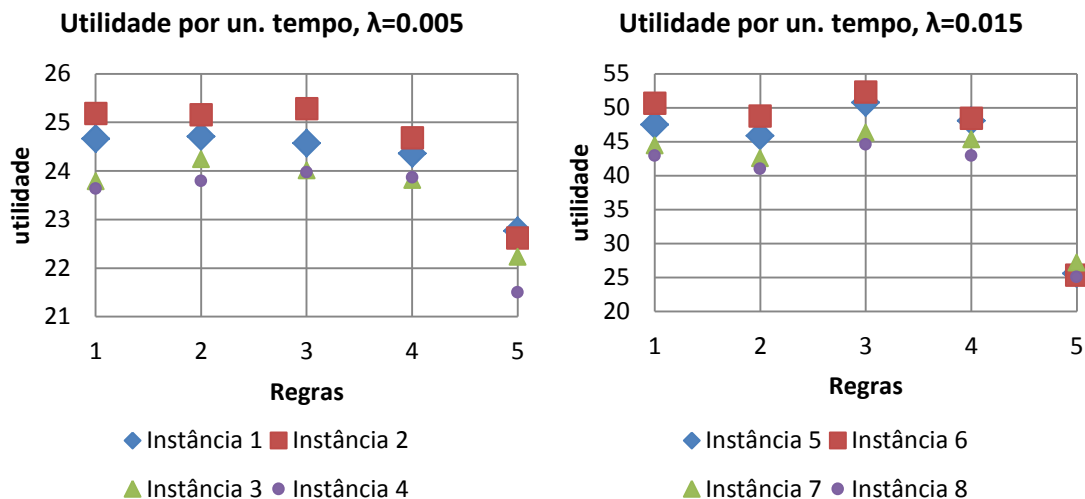
Para a utilidade total, e de acordo com a Figura 14, as regras com melhor desempenho são as regras 1, 2, 3 e 4 para qualquer que seja a taxa de chegadas. No entanto, a regra 3 destaca-se ligeiramente quando a taxa de chegadas é superior. Existe alguma relação entre a percentagem de alvos investigados com a utilidade total, no entanto não é linear e exemplo disso é a regra 3. Apesar de não ser das melhores

regras em termos de percentagem de alvos investigados, acaba por se destacar na utilidade total.

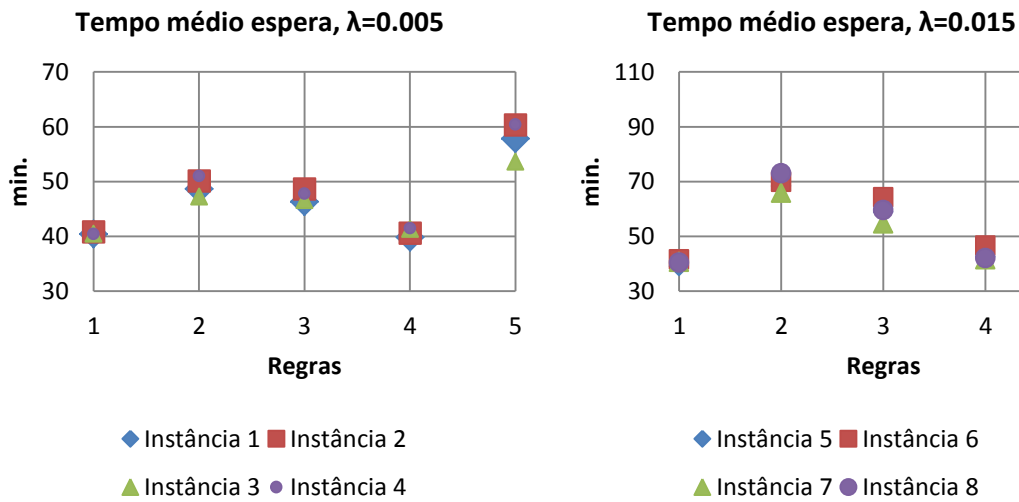
Relativamente ao tempo médio de espera de um alvo investigado, tempo que decorre desde que o alvo surge na área até que seja iniciada a sua investigação, e de acordo com a Figura 15, os valores variam entre 39 e 61 min para  $\lambda=0.005$  e entre 39 e 104 min para  $\lambda=0.015$ . Neste caso, com um intervalo de chegada de novos alvos mais pequeno, aumenta o número de alvos em fila de espera e consequentemente os seus tempos de espera.

Para esta medida de desempenho, destacam-se as regras 1 e 4, qualquer que seja a taxa de chegadas.

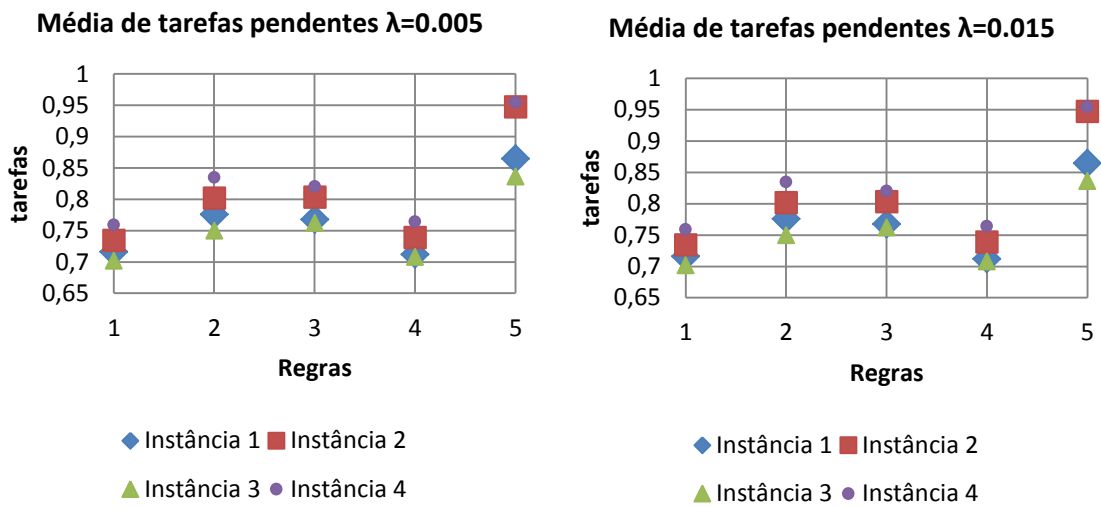
Verifica-se, conforme a Figura 16, que as regras 1 e 4 são também as que minimizam a média de tarefas pendentes por intervalo considerado ( $=10*\Delta t$ ). De facto, em modelos clássicos de filas de espera, é sabido que em médias, menores filas correspondem a menores tempos de espera.



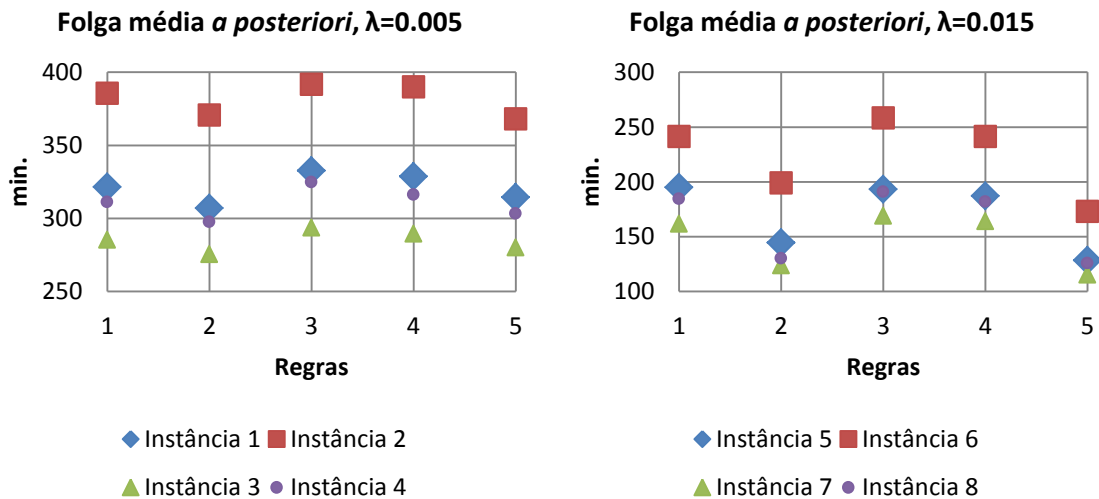
**Figura 14:** Gráficos com utilidade por limite de autonomia, em função da regra utilizada por instância.



**Figura 15:** Gráficos com tempo médio de espera dos alvos investigados, em função da regra utilizada por instância.



**Figura 16:** Gráficos com o número médio de tarefas pendentes, em função da regra utilizada por instância.



**Figura 17:** Gráficos com a folga média *a posteriori*, em função da regra utilizada por instância.

Verifica-se, para todas as medidas de desempenho analisadas, que os resultados sofrem um agravamento quando existe aumento na velocidade e na duração das tarefas dos alvos. Estes atributos têm influência no *deadline*, fazendo com que os alvos fiquem com uma janela de tempo disponível mais restrita. As instâncias 4 e 8 são as que apresentam geralmente piores resultados, considerando o aumento de 25% na duração das tarefas, provocando uma diminuição média de 7,5 min no *deadline*.

Relativamente à medida de folga média *a posteriori*, que mede o tempo que sobra entre o *deadline* de um alvo e o momento em que termina a sua investigação, destacam-se as regras 1, 3 e 4, com maior incidência nas instâncias com  $\lambda=0.015$ , conforme Figura 17.

De uma forma global, as regras 1 e 4 são as que apresentam melhores resultados, exceto na percentagem de alvos investigados, onde também se destaca a regra 2, e na utilidade total, onde se destaca a regra 3.

Por outro lado, a regra 5 é a que apresenta piores resultados. Para determinar a folga de cada alvo, entra-se, para além de outros atributos, com o *setup time*. Se o UAV estiver perto do alvo em processamento e surgir outro numa posição suficientemente afastada, o *setup time* do novo alvo pode ser tal que implique uma menor folga. Assim sendo, o UAV interrompe o percurso que estava a fazer e dirige-se para o novo alvo. Essas alterações no plano de voo podem conduzir a maus resultados face a certas medidas de desempenho.

Ao considerar instâncias onde a posição inicial dos alvos é gerada numa distância máxima ao centro da área, não só a percentagem de alvos investigados melhora, como todas as restantes medidas de desempenho também, qualquer que seja a taxa de chegadas.

Verifica-se que o *setup time* e a duração da tarefa têm grande influência na prioridade dos alvos. Se às regras em estudo juntarmos, à semelhança da regra 1, uma regra que só depende da duração da tarefa:  $f(t,i)=Dur\_task_i$  e se da regra 4 excluirmos o atributo do peso, ficando apenas:  $f(t,i)=setup\_time_i + Dur\_task_i$ , verifica-se que, para a primeira, os resultados são muito próximos da regra 1, melhorando inclusive algumas das medidas de desempenho e, para a segunda, existe uma melhoria, apesar de pequena, em quase todas as medidas de desempenho.

As Tabelas 5 e 6 sintetizam os resultados obtidos para todas as instâncias, sendo indicadas a **negrito** as melhorias obtidas.

**Tabela 5:** Resultados obtidos por aplicação da regra:  $f(t,i)=Dur\_task_i$

Instância	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº Médio de tarefas pendentes
1	70,40	76,5	45,19	328,33	24,55	0,75
2	68,83	77,3	47,55	391,02	24,70	0,79
3	70,99	74,1	44,99	289,94	23,72	0,74
4	67,86	74,2	47,36	317,81	23,79	0,80
5	37,87	49,0	55,90	210,80	46,07	1,73
6	32,02	50,4	65,11	<b>277,71</b>	47,08	1,97
7	42,54	45,2	54,68	<b>177,11</b>	42,69	1,59
8	34,93	45,9	61,90	<b>204,39</b>	43,37	1,80

**Tabela 6:** Resultados obtidos por aplicação da regra:  $f(t,i)=setup\_time_i + Dur\_task_i$

Instância	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº Médio de tarefas pendentes
1	71,81	76,6	40,12	323,66	24,66	0,71
2	<b>71,18</b>	78,1	<b>40,43</b>	387,07	25,13	<b>0,73</b>
3	72,09	74,5	<b>40,28</b>	285,44	23,92	<b>0,70</b>
4	69,33	74,4	41,05	313,26	<b>23,99</b>	0,76
5	43,98	<b>50,4</b>	40,41	<b>196,49</b>	47,51	1,47
6	40,52	<b>53,6</b>	42,39	245,68	50,67	1,58
7	46,61	<b>47,0</b>	41,77	167,82	44,46	1,40
8	40,14	<b>47,7</b>	42,45	183,28	<b>44,78</b>	1,57

### **5.2.3. Comparação com a regra FCFS**

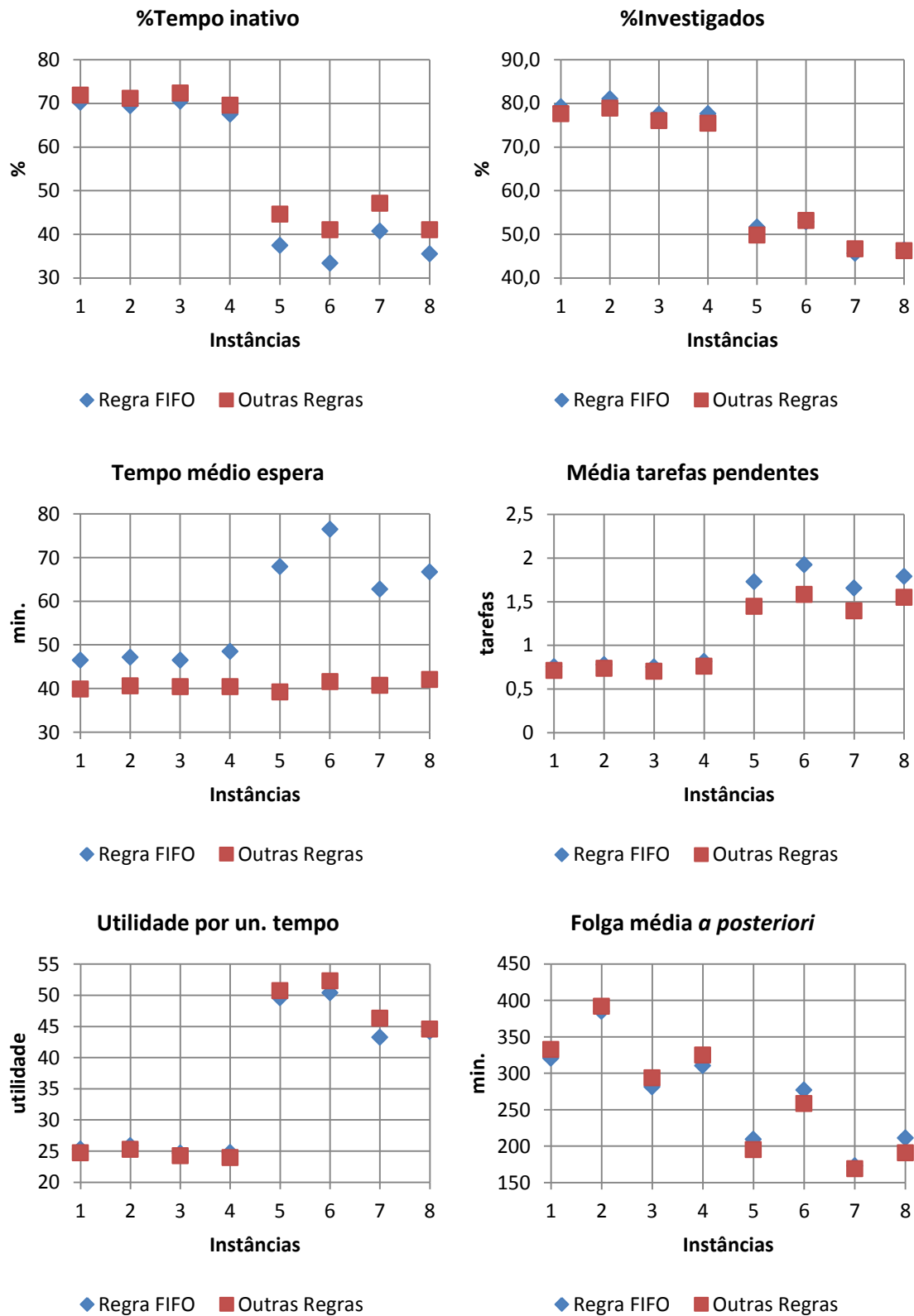
Comparando os melhores resultados obtidos pelos testes das 5 regras em estudo, com a regra simples FCFS (Tabela A3 em apêndice), e de acordo com a Figura 18, verifica-se que para as instâncias com  $\lambda=0.005$  (instancias 1, 2, 3 e 4), os valores das medidas de desempenho não apresentam grandes diferenças, melhorando em alguns casos com a regra FCFS, pelo que investigar os alvos pela sua ordem de entrada no sistema, quando a taxa de chegadas é pequena, é aparentemente suficiente.

No entanto, para as restantes instâncias, onde a taxa de ocupação do UAV é superior, existem melhorias significativas, quando implementadas as regras em estudo, nomeadamente para as medidas de percentagem do tempo inativo, tempo médio de espera e média de alvos no sistema.

### **5.2.4. Comparação de resultados com diferentes $\Delta t$**

Com variações no  $\Delta t$  verificam-se piores resultados à medida que o intervalo aumenta. O facto do intervalo de interrogações do sistema em busca de alvos mais prioritários ser maior, implica que variações relevantes nos atributos possam não ser detetadas com a antecedência mais conveniente. As medidas onde se verificam maiores diferenças, conforme se verifica nas Tabelas 7 e 8, são a percentagem de tempo inativo, percentagem de alvos investigados e tempo médio de espera.

No entanto, as variações nas medidas de desempenho não são muito elevadas, podendo definir-se um compromisso entre o tempo necessário para a computação e a qualidade das soluções.



**Figura 18:** Gráficos de comparação entre os melhores resultados obtidos e os resultados da regra FCFS.

**Tabela 7:** Comparação das medidas de desempenho para os diferentes  $\Delta t$ s, com  $\lambda=0.005$ 

	$\Delta t=1$	$\Delta t=5$	$\Delta t=15$
<b>%Tempo inativo</b>	71,87	67,82	59,08
<b>%investigados</b>	77,6	73,2	72,1
<b>Tempo médio espera (min)</b>	39,89	45,99	58,91
<b>Folga média <i>a posteriori</i> (min)</b>	332,67	328,97	326,35
<b>Utilidade por un. tempo</b>	24,71	23,17	23,00
<b>Nº Médio de tarefas pendentes</b>	0,71	0,61	0,72
<b>CPU (s)</b>	187	42,3	19

**Tabela 8:** Comparação das medidas de desempenho para os diferentes  $\Delta t$ s, com  $\lambda=0.015$ 

	$\Delta t=1$	$\Delta t=5$	$\Delta t=15$
<b>%Tempo inativo</b>	44,56	38,39	29,40
<b>%investigados</b>	49,8	46,3	41,8
<b>Tempo médio espera (min)</b>	39,25	46,40	59,98
<b>Folga média <i>a posteriori</i> (min)</b>	195,05	194,62	173,44
<b>Utilidade por un. tempo</b>	50,75	44,29	42,62
<b>Nº Médio de tarefas pendentes</b>	1,45	1,39	1,85
<b>CPU (s)</b>	99,7	24,5	11,1

## Capítulo 6

### 6. Conclusão

#### 6.1. Considerações finais

Com a crescente utilização de UAVs em diversos tipos de operações, têm surgido vários estudos que pretendem dar um contributo para a otimização do seu desempenho.

Na literatura encontram-se referências nas mais variadas vertentes da utilização destes meios, principalmente com adaptações aos problemas de *Vehicle Routing* (VRP) e não tão frequentemente aos problemas de *Job Scheduling* (JSP).

O problema de roteamento de um UAV pode ser descrito como um problema onde um veículo sai de um depósito, após determinada a sua rota — pelo menos a rota inicial —, com destino aos locais onde irá desempenhar a sua função. À chegada a cada local, inicia a execução de uma operação, que terá uma determinada duração. É, neste ponto, que este problema tem semelhanças tanto com VRP como com JSP.

Enquanto em VRP, o tempo que é despendido em cada cliente visitado é normalmente desprezável ou incluído no tempo de percurso, *setup time*, em JSP, pelo contrário, é o tempo de processamento da tarefa que tem um papel predominante, podendo incluir o tempo de preparação da tarefa ou do próprio processador. Para problemas com UAVs ambos poderão ser considerados e ter implicações relevantes na escolha da próxima operação a realizar.

Qualquer um destes problemas tem sido também objeto de muita investigação, considerando que o número de clientes/tarefas não é totalmente conhecido no início do problema, mas sim surgindo ao longo do tempo de forma estocástica, e considerando também tempos de processamento estocásticos, obrigando a um replaneamento em tempo real, conferindo uma perspetiva dinâmica aos problemas.

Apesar da aplicação de regras de prioridade ser mais comum em JSP, tem vindo a ser admitido em problemas dinâmicos de *Vehicle Routing* que o objetivo não é determinar *a priori* a rota final do veículo, mas ir adaptando o seu percurso à medida

que a operação decorre, permitindo englobar novos clientes ou atender a alterações na procura. Existe, por isso, necessidade de dispor de comunicações em tempo real e de um permanente conhecimento da posição e outros atributos dos clientes.

Neste estudo considerou-se um UAV a operar numa área com limites bem definidos, onde surgem alvos de acordo com um processo *Poisson de chegadas*, tendo sido realizadas experiências com diferentes taxas de chegada. Os alvos têm movimento, alterando a sua posição ao longo do tempo, o que tem implicações diretas nos valores dos tempos de preparação pré-operação (*setup times*). Mesmo que o movimento seja determinístico e perfeitamente conhecido, este atributo tem fortes implicações, não só na escolha do alvo a investigar como na variação do grau de prioridade dos alvos, já que a exequibilidade de investigação de um alvo depende fortemente do *setup time* necessário.

Quando não existem alvos em espera para processamento, é permitido ao UAV pairar numa posição de *loitering*, tendo sido consideradas duas alternativas: no centro da área de operações ou na posição do último alvo investigado.

Apesar de termos considerado apenas um UAV, o problema poderá ser facilmente adaptado a missões com múltiplos UAVs se a área de operações for dividida em sub-áreas mais pequenas, atribuindo um UAV a cada. Não foi, para já, admitido que possam operar vários UAVs na mesma área.

Foram testadas cinco regras de prioridade em oito instâncias diferentes do problema, sendo parâmetros diferenciadores de cada instância: a taxa de chegadas, a velocidade e duração de processamento dos alvos, tendo sido considerada como *benchmark* a regra FCFS (*First Come First Served*), que ordena os alvos por ordem de chegada.

Verificou-se que tanto o *setup time* como a duração de processamento são elementos vitais na ordenação dos alvos, tendo sido notada uma melhoria quando estes dois atributos são conjugados numa mesma regra.

Apesar de não ter sido considerada, a autonomia do UAV é um fator determinante na operação do UAV, sendo por isso importante fazer uma escolha criteriosa dos alvos cuja investigação é mais urgente. A urgência ou importância pode ser determinada com base em determinados fatores, referidos no capítulo 1, secção 1.3, sendo o operador ou alguém com responsabilidade apropriada a atribuir os diferentes pesos aos alvos. A regra 3, neste caso, demonstrou bons resultados quando o objetivo principal é maximizar a utilidade associada à investigação dos alvos.

Foi permitido que o processamento de um alvo, incluindo o seu *setup time*, fosse interrompido em detrimento de um alvo com maior prioridade. No entanto, o facto de se interromper uma tarefa em curso pode implicar que já não seja possível voltar a executá-la, por já não ser possível alcançar o alvo antes que este saia da área de operações. É, por isso, necessário que exista um balanceamento apropriado entre os diferentes atributos, de modo a que se consiga dar resposta a um maior número de solicitações.

De todas as regras em estudo, a regra 5 foi a que apresentou resultados relativamente mais fracos. O facto de a “folga” de um alvo depender do *setup time*, sendo tanto menor quanto maior este for, implica que se abandone um alvo próximo em direção a outro, mais afastado, quando provavelmente seria possível seguir para o segundo alvo em tempo útil após término do primeiro. O mesmo acontece com a interrupção da própria investigação. É, por isso, de evitar a aplicação de regras em que o fator de ordenação dependa de atributos com grandes variações como é o caso do *setup time* ou da duração de processamento.

É, ainda, de assinalar que a comparação com a regra de *benchmark* FCFS permitiu verificar que, em ambientes com maior número de alvos por unidade de tempo, a investigação por ordem de chegada não é eficaz.

Em ambientes militares, é necessário fazer uma atualização permanente do panorama, isto é, do conhecimento situacional. Para além de uma eventual escalada na criticidade dos alvos ao longo do tempo, poderão surgir outros com maior urgência de investigação. É, por isso, importante que mesmo durante o percurso e processamento de um alvo, se mantenha uma permanente observação do ambiente em redor e a disponibilidade para, de forma ágil, reconsiderar opções anteriormente tomadas.

A simulação pseudo-contínua do sistema, com recurso a *time steps*, em detrimento da *simulação de eventos discretos*, permite não só a verificação em cada instante da necessidade de pequenos ajustes como a observação de acontecimentos que podem dar indicação de ações suspeitas, como por exemplo a aproximação excessiva entre dois alvos.

Quando um UAV opera numa área com limites bem definidos e a chegada dos alvos ocorre em posições aleatórias em toda a extensão da área, podem surgir alvos em que o seu *deadline* seja tal que não permita sequer a aproximação do UAV, influenciando os resultados obtidos. Os resultados sofrem uma melhoria se, ao invés de uma distribuição aleatória por toda a área, for definido um limite em torno do centro

da área onde os alvos só serão considerados se detetados dentro desses limites. A delimitação da área de operações serve para limitar a operação do UAV, só podendo seguir alvos dentro desses limites. Não significa, no entanto, que tenha de processar todos os alvos que surjam.

## **6.2. Trabalho futuro**

Este estudo baseou-se na vertente mais simples da operação com UAVs. Considerou-se que não existem alterações ao longo do tempo de alguns dos atributos mais relevantes, a monitorização dos alvos é perfeita (não sofrendo influências de fatores externos) e só se opera um UAV. No entanto, o presente estudo poderá servir de base para a concepção de estudos futuros mais complexos, já que a literatura em geral não contempla este problema como uma conjugação simultânea de vários atributos importantes, como o *setup time*, a duração de processamento e o movimento dos alvos.

Aproximando-nos da realidade, é importante aumentar a complexidade do problema e considerar outros pressupostos na definição dos principais atributos.

Na realidade, o movimento dos alvos não é habitualmente linear ou feito a velocidade constante, mesmo que aparentemente se dirijam de um ponto ao outro sem interrupções. Principalmente em ambiente marítimo, existem fatores externos, tais como a ondulação e o vento, que introduzem pequenas variações no movimento dos alvos, ditando erros na posição que se vão agravando ao longo do tempo.

Não existindo conhecimento exato da variação do movimento de um alvo, é importante manter um histórico atualizado, de modo a que seja possível fazer previsões de movimentos futuros.

O movimento não linear e a velocidade não constante terá consequências diretas não só no *setup time* como no instante em que um alvo atinge os limites da área, sendo necessário implementar intervalos de atuação, de modo a que as operações tenham lugar ainda dentro da área de operações.

À semelhança do movimento dos alvos, também o tempo de duração de processamento pode alterar. Não dispondo do conhecimento prévio exato do tempo que o UAV necessitará para processar um alvo, poderá ter também implicações negativas na sua finalização, caso o alvo esteja demasiado perto dos limites da área.

A geometria da área de operações também não é, em geral, um quadrado ou retângulo perfeitos, pelo que há que considerar outras formas geométricas, aumentando a dificuldade na determinação dos *deadlines* dos alvos.

A posição de *loitering* é outro aspeto importante. No presente estudo não existe indicação de predominância do surgimento de novos alvos apenas num setor da área, mas será importante fazer o levantamento ou a predição de zonas com maior probabilidade da presença futura de alvos merecedores de investigação e posicionar o UAV de modo a que fique a uma distância favorável para futuras interceções.

A autonomia do UAV não foi considerada, de modo a permitir uma obtenção de estatísticas suficientes para a avaliação precisa das medidas de desempenho. No entanto, este é um fator que tem grandes implicações na operação de um UAV. Para além da distância que lhe será permitido percorrer, considerando que terá ainda de voltar à plataforma de lançamento, a autonomia (limitada) poderá ainda condicionar qual a melhor velocidade a praticar, já que maiores velocidades poderão proporcionar mais gastos na autonomia. A dimensão da área de operações também poderá depender da autonomia, já que não deverá ser demasiado grande, se não permitir o regresso em segurança do UAV. A consideração de custos de operação, nomeadamente com combustível, poderá também ser um critério importante para a otimização.

A introdução da operação de vários UAVs em simultâneo pode ser considerada de duas formas: dividindo a área de operações, de modo a que cada UAV fique com uma área, independente, da sua responsabilidade ou operando vários UAVs na mesma área, com maior ou menor coordenação entre eles. Os alvos que se movimentem entre subáreas contíguas, se não forem processados por um UAV, poderão passar a ser considerados pelo UAV afeto à subárea seguinte.

Nem sempre a posição dos alvos é conhecida, pelo que o UAV poderá ser apenas direcionado para uma zona onde é expectável a localização dos alvos. Nesta situação, só será determinado qual o alvo a processar quando o UAV se encontrar a uma distância suficientemente próxima e que consiga confirmar a existência de alvos. Se, em vez de apenas um alvo, se encontrarem vários na mesma zona, suficientemente próximos uns dos outros, o *setup time* já não será o atributo mais importante, sendo então de considerar outras regras de prioridade.

Podem ser consideradas diferentes plataformas de lançamento e recolha e posicionadas de forma estratégica, de modo a permitir um maior afastamento do UAV, bem como o seu reabastecimento. Estas plataformas podem ainda ter movimento, sendo importante saber em cada instante a sua posição, de modo a estabelecer instantes limite de regresso do UAV ou poderão mesmo acompanhar o movimento do UAV, mesmo com velocidade mais baixa, permitindo maior liberdade de movimento.

O próprio movimento do UAV poderá não ser em linha reta, principalmente quando abandona a posição de um alvo e se dirige para outro, existindo sempre uma curvatura no seu movimento, o que aumentará a distância percorrida e consequentemente o consumo de combustível.

Por fim, é de considerar que nem todos os alvos serão sujeitos ao mesmo tipo de processamento. Um UAV pode estar equipado com diferentes tipos de sensores e deve existir um balanceamento entre os alvos que necessitam de um mesmo sensor juntando-os na mesma ordem de processamento antes do UAV voltar à plataforma para mudar de sensor.

Poderão ainda ser considerados sensores com limitações, dependendo das condições atmosféricas, altitude ou velocidade do UAV.

# Bibliografia

- [1] Ahner, D. K.; Buss, A. H.; Ruck, J. (2006) **Assignment scheduling capability for unmanned aerial vehicles - A discrete event simulation with optimization in the loop approach to solving a scheduling problem.** *Proceedings of the 2006 Winter Simulation Conference.*
- [2] Al Rowaei, A. A.; Buss, A. H.; Lieberman, S. (2011). **The effects of time advance mechanism on simple agent behaviors in combat simulations.** *Proceedings of the 2011 Winter Simulation Conference.*
- [3] Almirante Chefe do Estado Maior da Armada. (2014). **Diretiva de planeamento da Marinha.** *Marinha Portuguesa.*
- [4] Alver, Y.; Özdoğan, M.; Yücesan, E. (2012). **Assessing the robustness of UAV assignments.** *Proceedings of the 2012 Winter Simulation Conference.*
- [5] Baker, K. R.; Trietsch, D. (2009). **Principles of Sequencing and Scheduling.** *John Wiley & Sons.*
- [6] Barman, S. (1998). **The impact of priority rule combinations on lateness and tardiness.** *IIE Transactions*, v.30, p. 495-504.
- [7] Bednowitz, N.; Batta, R.; Nagi, R. (2014). **Dispatching and loitering policies for unmanned aerial vehicles under dynamically arriving multiple priority targets.** *Journal of Simulation*, v. 8, p. 9-24.
- [8] Bertuccelli, L. F.; Beckers, N. W. N.; Cummings, M. L. (2010). **Developing operator model for UAV search scheduling.** *Proceedings of AIAA Guidance Navigation and Control Conference.*
- [9] Dominic, P. D. D.; Kaliyamoorthy, S.; Kumar, M. S. (2004). **Efficient dispatching rules for dynamic job shop scheduling.** *International Journal of Advanced Manufacturing Technology*, v. 24, p. 70-75.
- [10] Enright, J. J.; Frazzoli, E.; Pavone, M.; Savla, K. (2015). **UAV routing and coordination in stochastic, dynamic environments.** In: Valavanis, K. P.; Vachtsevanos, G. J. (Eds.). (2015). *Handbook of Unmanned Aerial Vehicles.* Springer. Cap. 85, p. 2079-2109.
- [11] Estado-Maior da Armada. (2011). **Marinha: Ao Serviço de Portugal.** Disponível em <http://www.marinhasplp.org/PT/asmarinhas/doutrinas/Documents/>. Acedido em outubro 2015.
- [12] Evers, L.; Dollevoet, T.; Barros, A. I.; Monsuur, H. (2012). **Robust UAV mission planning.** *Annals of Operations Research*, v. 222, p. 293-315.

- [13] Evers, L.; Dollevoet, T.; Barros, A. I.; Monsuur, H. (2014a). **A two-stage approach to the orienteering problem with stochastic weights**. *Computers & Operations Research*, v. 43, p. 248-260.
- [14] Evers, L.; Dollevoet, T.; Barros, A. I.; Monsuur, H. (2014b) **Online stochastic UAV mission planning with time windows and time-sensitive targets**. *European Journal of Operational Research*, v. 238(1), p. 348-362.
- [15] Evers, L.; Dollevoet, T.; Barros, A. I.; Monsuur, H. (2015). **UAV mission planning: from robust to agile**. In Zeimpekis et al. (eds) *Military Logistics*, p. 1-17. Springer.
- [16] French, S. (1982). **Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop**. Ellis Horwood.
- [17] Gabinete da Autoridade Aeronáutica Nacional. (2013). **Emissão de licenças especiais de aeronavegabilidade para sistemas de aeronaves não tripuladas**. *Autoridade Aeronáutica Nacional*. Circular nº01/13, 1ª ed.
- [18] Ghiani, G. et al. (2003). **Real-time vehicle routing: Solutions concepts, algorithms and parallel computing strategies**. *European Journal of Operational Research*, 151(1), p. 1-11.
- [19] Henriques, R.; Bação, F.; Lobo, V. (2009). **UAV path planning based on event density detection**. *Proceedings of the IEEE International Conference on Advanced Geographic Information Systems & Web Services, 2009, GEOWS'09*. p. 112-116.
- [20] House of Commons Defence Committee. (2008). **The Contribution of Unmanned Aerial Vehicles to ISTAR Capability**. *The Stationery Office Limited, London*.
- [21] Husby, C. R. (2005). **Path generation tactics for a UAV following a moving target**. University of Washington. Dissertação de Mestrado.
- [22] Kutanoglu, E.; Sabuncuoglu, I. (1999). **An analysis of heuristics in a dynamic job shop with weighted tardiness objectives**. *International Journal Production Research*, v. 37, p. 165-187.
- [23] Mccadden, K. K.; Nigus, C. A. (2008). **Allocation of UAV search efforts using dynamic programming and bayesian updating**. Naval Postgraduate School, Monterey. Dissertação de Mestrado.
- [24] Pillac, V. et al. (2013). **A review of dynamic vehicle routing problems**. *European Journal of Operational Research*, v. 225(1), 1-11.
- [25] Rajendran, C.; Holthaus, O. (1999). **A comparative study of dispatching rules in dynamic flowshops and jobshops**. *European Journal of Operational Research*, v. 116, p. 156-170.

- [26] Ries, J.; Ishizaka, A. (2012) **A multi-criteria support system for dynamic aerial vehicle routing problems**. In *Proceedings of the 2<sup>nd</sup> IEEE International Conference on Communications, Computing and Control Applications (CCCA)*, p. 1-4.
- [27] Ries, J.; Ishizaka, A.; Beullens, P. (2011). **Toward a decision support system for aerial vehicle routing problems**. *International Conference on Operations Research*.
- [28] Sousa, J. et al. (2015). **Unmanned aircraft systems for maritime operations**. In Valavanis, K. P.; Vachtsevanos, G. J. (Eds.). (2015) *Handbook of Unmanned Aerial Vehicles*, p. 2787-2811. Springer.
- [29] Sözen, V. (2014, June). **Optimal Deployment of Unmanned Aerial Vehicles for Border Surveillance**. Naval Postgraduate School, Monterey. Dissertação de Mestrado.
- [30] Sundar, K.; Rathinam, S. (2013). **A primal-dual heuristic for heterogeneous unmanned vehicle path planning problem**. *International Journal of Advanced Robotic Systems*, v. 10(349), p. 1-9.
- [31] Sundar, K.; Rathinam, S. (2013). **Algorithms for routing an unmanned aerial vehicle in the presence of refueling depots**. *IEEE Transactions on Automation Science and Engineering*, v.11(1), p. 287-294.
- [32] Terekhov, D.; Tran, T. T.; Down, D. G.; Beck, J. C. (2014). **Integrating queueing theory and scheduling for dynamic scheduling problems**. *Journal of Artificial Intelligence Research*, v. 50, p. 535-572.
- [33] Weinstein, A.; Schumacher, C. (2007). **UAV scheduling via the vehicle routing problem with time windows**. In *Proc. AIAA Infotech@ Aerospace 2007 Conference and Exhibit. California*.

## **Apêndice A**

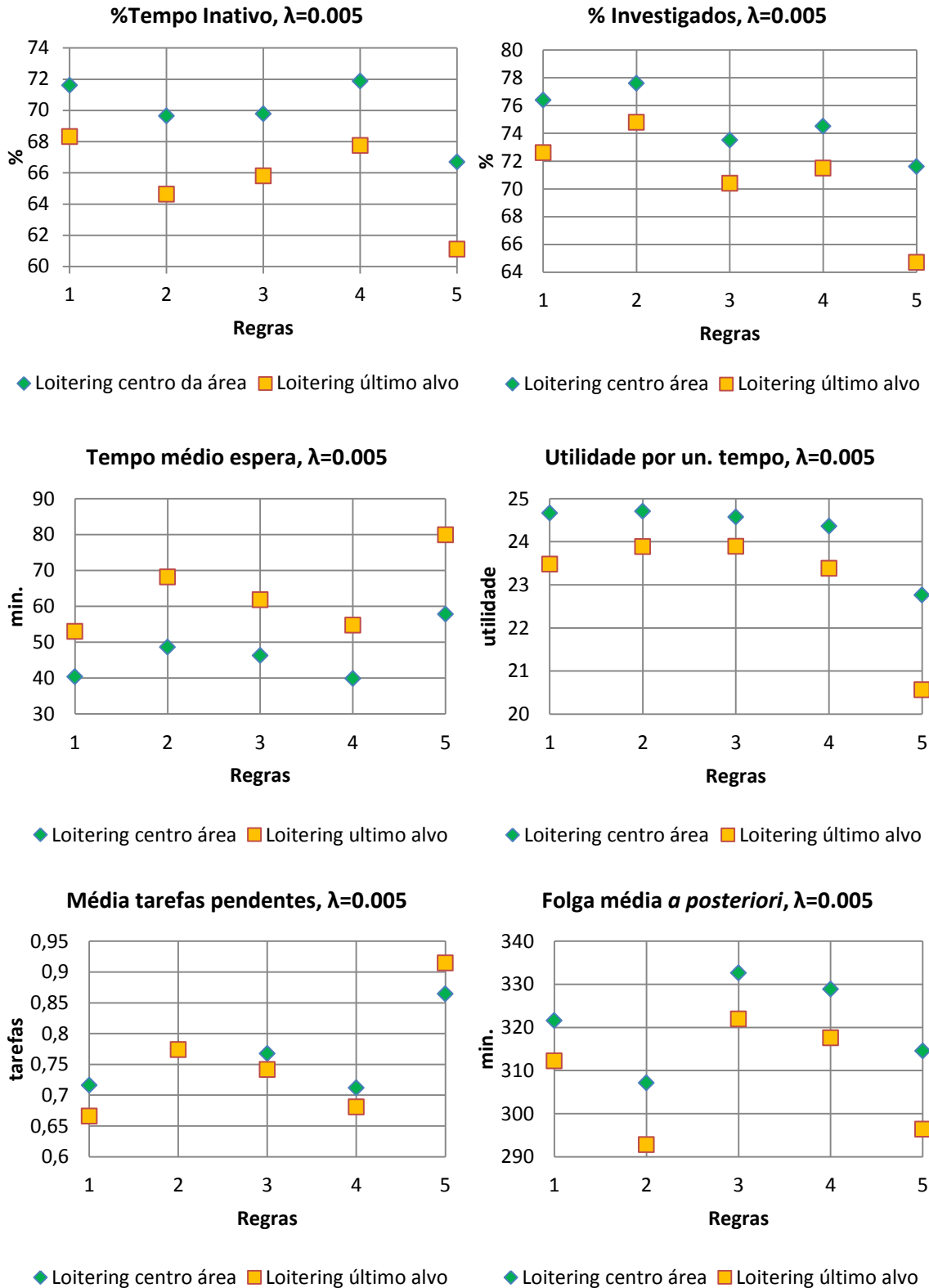
## A.1 Resultados com $\Delta t=1$ e diferentes posições de *loitering*

Tabela A1: Resultados para  $\Delta t=1$  e posição de *loitering* no centro da área

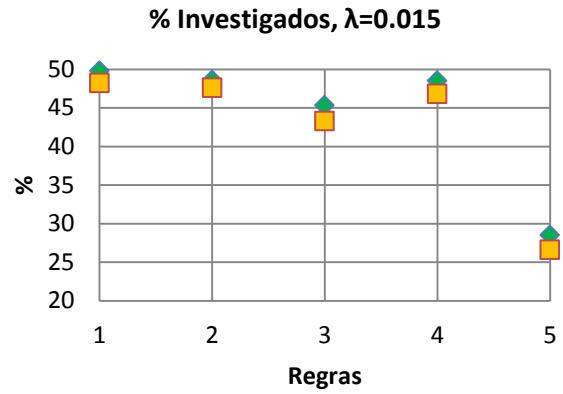
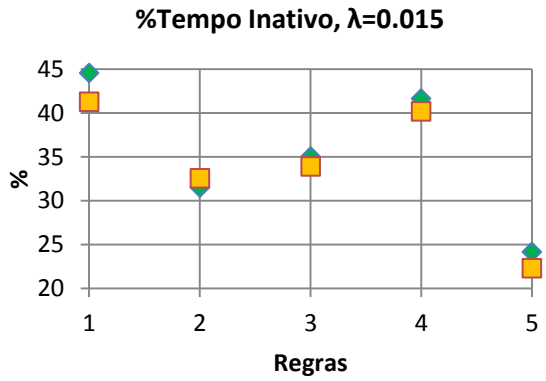
Instância	Regra	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº médio de tarefas pendentes
1	1	71,60	76,4	40,45	321,62	24,66	0,72
	2	69,64	<b>77,6</b>	48,66	307,13	<b>24,71</b>	0,78
	3	69,77	73,5	46,34	<b>332,67</b>	24,57	0,77
	4	<b>71,87</b>	74,5	<b>39,89</b>	328,87	24,36	<b>0,71</b>
	5	66,68	71,6	57,84	314,53	22,77	0,86
2	1	71,07	78,3	40,87	385,84	25,18	<b>0,73</b>
	2	68,56	<b>78,9</b>	50,17	370,96	25,16	0,80
	3	68,64	75,8	48,71	<b>391,97</b>	<b>25,28</b>	0,80
	4	<b>71,11</b>	75,8	<b>40,65</b>	390,04	24,69	0,74
	5	64,04	71,9	60,36	368,27	22,62	0,95
3	1	<b>72,35</b>	73,8	<b>40,48</b>	285,37	23,78	<b>0,70</b>
	2	70,51	<b>76,0</b>	47,27	275,66	<b>24,24</b>	0,75
	3	70,01	71,4	46,66	<b>293,67</b>	24,01	0,76
	4	72,00	73,1	41,37	289,67	23,81	0,71
	5	67,42	70,4	53,64	280,16	22,23	0,84
4	1	<b>69,56</b>	73,3	<b>40,47</b>	311,22	23,63	<b>0,76</b>
	2	66,91	<b>75,4</b>	51,03	297,81	23,79	0,83
	3	67,52	70,3	47,79	<b>324,91</b>	<b>23,97</b>	0,82
	4	69,32	73,0	41,50	316,37	23,87	<b>0,76</b>
	5	63,32	67,8	60,39	303,39	21,50	0,95
5	1	<b>44,56</b>	<b>49,8</b>	<b>39,25</b>	<b>195,05</b>	47,49	<b>1,45</b>
	2	31,49	48,6	68,37	144,69	45,87	1,97
	3	35,02	45,3	58,93	193,43	<b>50,75</b>	1,87
	4	41,63	48,5	42,62	187,40	48,05	1,56
	5	24,15	28,5	89,83	128,71	25,57	2,87
6	1	<b>41,01</b>	<b>53,1</b>	<b>41,63</b>	241,53	50,69	<b>1,58</b>
	2	29,63	51,9	69,93	199,10	48,77	2,06
	3	30,44	46,5	64,31	<b>258,62</b>	<b>52,34</b>	2,06
	4	39,18	50,2	46,74	241,57	48,46	1,67
	5	18,78	27,1	103,33	173,31	25,35	3,44
7	1	<b>47,10</b>	<b>46,6</b>	<b>40,74</b>	161,98	44,44	<b>1,39</b>
	2	37,59	45,1	65,91	123,92	42,55	1,79
	3	39,58	42,0	54,76	<b>169,08</b>	<b>46,31</b>	1,67
	4	46,29	45,9	41,66	164,28	45,33	1,43
	5	31,06	28,6	70,06	115,36	27,19	2,37
8	1	<b>41,02</b>	<b>46,2</b>	<b>40,55</b>	184,52	42,92	<b>1,55</b>
	2	30,45	43,7	73,09	130,28	40,98	2,02
	3	33,40	38,6	59,69	<b>190,95</b>	<b>44,59</b>	2,02
	4	40,62	44,4	42,12	182,09	42,93	1,58
	5	24,65	26,8	87,89	125,93	25,07	2,56

**Tabela A2:** Resultados para  $\Delta t=1$  e posição de *loitering* no último alvo investigado

Instância	Regra	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº médio de tarefas pendentes
1	1	<b>68,31</b>	72,6	<b>53,06</b>	312,25	23,49	<b>0,67</b>
	2	64,63	<b>74,8</b>	68,19	292,78	23,89	0,77
	3	65,80	70,4	61,84	<b>321,96</b>	<b>23,90</b>	0,74
	4	67,74	71,5	54,76	317,57	23,38	0,68
	5	61,10	64,7	79,91	296,40	20,57	0,91
2	1	<b>68,18</b>	74,9	<b>51,96</b>	378,53	24,14	<b>0,67</b>
	2	65,65	<b>77,2</b>	62,50	363,10	24,45	0,75
	3	65,65	72,9	60,20	<b>390,59</b>	<b>24,78</b>	0,75
	4	67,88	73,8	52,89	379,16	23,94	0,68
	5	59,44	67,4	83,36	360,04	21,35	0,97
3	1	<b>67,91</b>	67,6	56,22	281,83	21,94	<b>0,67</b>
	2	64,03	<b>70,3</b>	72,00	265,27	<b>22,81</b>	0,78
	3	65,70	65,2	64,63	<b>291,05</b>	22,20	0,74
	4	67,81	66,0	<b>56,11</b>	285,05	21,59	0,68
	5	59,88	60,4	87,04	271,38	19,77	0,92
4	1	<b>65,76</b>	68,8	<b>55,67</b>	306,89	22,28	<b>0,72</b>
	2	62,32	<b>70,6</b>	70,44	286,44	<b>22,42</b>	0,83
	3	63,40	65,4	64,43	<b>321,81</b>	22,29	0,80
	4	65,11	68,4	56,86	310,61	22,29	0,74
	5	58,14	60,1	86,39	296,92	19,16	0,98
5	1	<b>41,27</b>	<b>48,2</b>	<b>43,81</b>	191,02	46,09	<b>1,39</b>
	2	32,55	47,6	68,88	148,96	44,33	1,81
	3	33,92	43,3	62,23	<b>195,09</b>	<b>48,89</b>	1,77
	4	40,18	46,8	49,16	184,98	46,58	1,46
	5	22,31	26,6	100,85	129,14	24,62	2,82
6	1	<b>39,00</b>	<b>51,5</b>	<b>46,70</b>	238,85	49,09	<b>1,48</b>
	2	28,26	51,4	77,10	189,24	47,92	2,06
	3	29,21	43,6	70,28	<b>262,04</b>	49,18	2,01
	4	38,07	50,1	51,21	234,34	<b>49,84</b>	1,55
	5	18,19	25,6	124,25	167,76	24,16	3,37
7	1	<b>45,64</b>	<b>44,6</b>	<b>44,63</b>	158,59	41,99	<b>1,27</b>
	2	38,68	43,2	67,32	128,43	40,01	1,62
	3	38,94	38,7	61,05	<b>170,59</b>	<b>43,11</b>	1,55
	4	44,73	42,3	46,39	159,18	41,94	1,32
	5	31,40	27,6	75,62	119,14	24,94	2,15
8	1	<b>40,07</b>	<b>44,5</b>	<b>44,44</b>	173,37	41,70	<b>1,44</b>
	2	30,08	42,0	78,88	128,69	39,78	1,94
	3	33,22	37,5	62,56	<b>186,15</b>	<b>43,43</b>	1,86
	4	38,54	44,0	49,47	168,23	42,66	1,51
	5	22,96	22,8	103,31	95,66	20,81	2,74

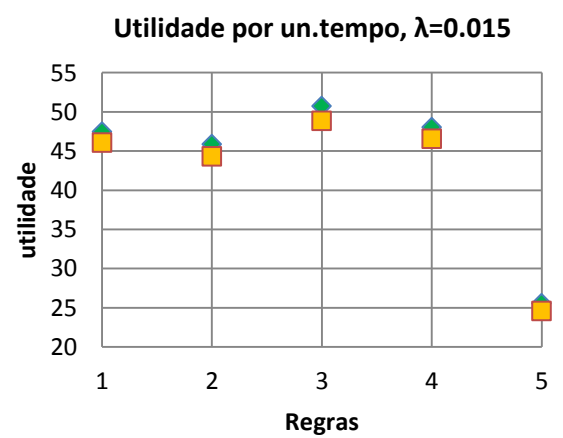
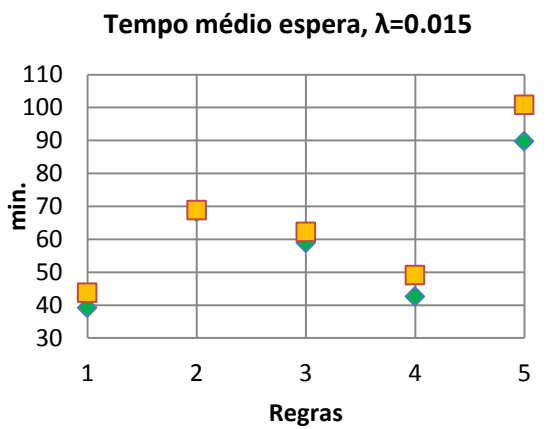


**Figura A1:** Gráficos de comparação entre as duas posições de *loitering* para  $\lambda=0.005$



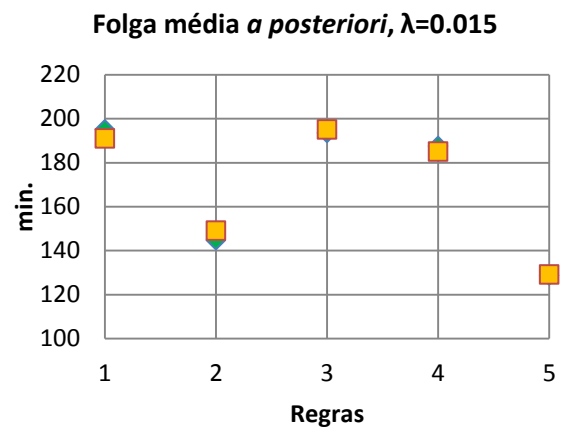
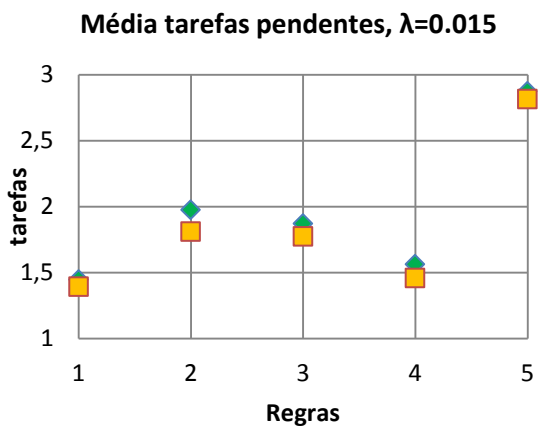
◆ Loitering centro da área ■ Loitering último alvo

◆ Loitering centro área ■ Loitering último alvo



◆ Loitering centro área ■ Loitering último alvo

◆ Loitering centro área ■ Loitering último alvo



◆ Loitering centro área ■ Loitering último alvo

◆ Loitering centro área ■ Loitering último alvo

**Figura A2:** Gráficos de comparação entre as duas posições de *loitering* para  $\lambda=0.015$



## A.2 Resultados para a regra FCFS

**Tabela A3:** Resultados para a regra FCFS com  $\Delta t=1$

Instância	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº médio de tarefas pendentes
1	70,19	79,2	46,53	320,70	25,31	0,75
2	69,45	81,0	47,19	384,93	25,85	0,78
3	70,45	77,5	46,53	281,24	24,70	0,75
4	67,48	77,6	48,50	310,67	24,76	0,82
5	37,39	51,6	67,96	209,56	49,59	1,73
6	33,33	52,8	76,51	277,02	50,42	1,92
7	40,68	45,6	62,82	173,23	43,26	1,65
8	35,50	46,4	66,78	211,20	44,21	1,79

### A.3 Resultados com $\Delta t=5$ e $\Delta t=15$

**Tabela A4:** Resultados para posição de *loitering* no centro área e  $\Delta t=5$

Instância	Regra	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº médio de tarefas pendentes
1	1	59,01	71,5	<b>45,99</b>	307,44	<b>23,00</b>	0,74
	2	56,85	<b>72,1</b>	55,55	289,78	22,85	0,75
	3	58,05	66,9	53,40	<b>326,35</b>	22,65	<b>0,72</b>
	4	<b>59,08</b>	69,7	47,37	308,68	22,57	0,73
	5	53,66	66,4	68,64	287,16	21,15	0,74
5	1	<b>38,39</b>	<b>46,3</b>	<b>46,40</b>	182,10	43,48	<b>1,39</b>
	2	28,32	44,3	72,25	141,89	41,94	1,69
	3	31,35	39,1	65,82	<b>194,62</b>	<b>44,29</b>	1,63
	4	36,81	45,0	47,98	185,39	43,46	1,43
	5	23,73	28,2	92,00	125,71	25,94	2,01

**Tabela A5:** Resultados para posição de *loitering* no centro área e  $\Delta t=15$

Instância	Regra	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. tempo	Nº médio de tarefas pendentes
1	1	<b>67,69</b>	70,7	<b>58,91</b>	314,00	22,91	0,63
	2	65,29	<b>73,2</b>	68,88	300,49	<b>23,17</b>	0,63
	3	66,37	67,9	64,40	<b>328,97</b>	22,81	0,63
	4	67,82	70,2	57,56	321,98	22,89	<b>0,61</b>
	5	62,60	66,8	85,33	302,59	21,12	0,67
5	1	<b>29,40</b>	41,5	63,11	160,93	39,46	1,87
	2	21,19	37,6	99,19	106,89	36,64	2,18
	3	25,14	35,7	72,38	<b>173,44</b>	<b>42,62</b>	2,00
	4	27,96	<b>41,8</b>	<b>59,98</b>	159,33	40,42	<b>1,85</b>
	5	18,68	23,4	113,80	105,44	21,56	2,59

## A.4 Resultados para instâncias com as posições iniciais dos alvos a uma distância limite do centro da área

**Tabela A6:** Resultados para  $\Delta t=1$  e posição de *loitering* no centro da área e com geração de posições iniciais dos alvos a uma distância máxima de 25 nm do centro da área

Instância	Regra	%Tempo Inativo	% investigados	Tempo médio de espera	Folga média a posteriori	Utilidade por un. Tempo	Nº médio de tarefas pendentes
1	1	<b>77,58</b>	89,2	<b>17,28</b>	450,39	29,00	<b>0,54</b>
	2	75,42	<b>91,4</b>	25,54	441,87	<b>29,86</b>	0,60
	3	75,22	86,3	24,59	<b>457,15</b>	29,05	0,61
	4	77,51	88,1	20,12	447,38	28,75	<b>0,54</b>
	5	66,04	76,8	42,63	386,71	24,95	1,04
2	1	<b>77,44</b>	90,9	<b>19,82</b>	521,09	29,70	<b>0,54</b>
	2	75,89	<b>93,3</b>	23,25	512,13	<b>30,28</b>	0,58
	3	74,64	88,6	26,53	<b>523,83</b>	29,74	0,61
	4	77,41	90,3	20,01	516,22	29,44	<b>0,54</b>
	5	68,61	83,8	42,34	479,68	27,19	0,89
3	1	<b>78,08</b>	84,7	24,04	422,04	27,51	<b>0,54</b>
	2	75,32	<b>87,9</b>	30,28	413,18	<b>28,52</b>	0,61
	3	75,18	83,2	30,31	<b>426,08</b>	28,02	0,61
	4	77,84	84,3	<b>23,47</b>	417,55	27,45	<b>0,54</b>
	5	64,57	71,2	51,57	329,73	23,34	1,09
4	1	<b>74,43</b>	84,4	25,86	449,36	27,51	0,62
	2	71,82	86,6	32,28	440,16	<b>28,32</b>	0,69
	3	71,98	81,6	32,26	<b>458,22</b>	27,83	0,68
	4	74,40	83,4	<b>25,19</b>	448,40	27,27	<b>0,61</b>
	5	62,54	71,8	47,25	375,49	23,50	1,18
5	1	<b>44,83</b>	72,9	<b>36,32</b>	326,58	<b>69,99</b>	<b>1,39</b>
	2	28,00	<b>73,8</b>	72,34	278,61	69,77	2,18
	3	28,52	62,3	81,33	<b>326,67</b>	68,49	2,22
	4	41,28	72,2	48,57	325,44	68,65	1,51
	5	8,37	20,0	272,80	191,90	18,80	5,09
6	1	<b>45,40</b>	75,1	<b>46,98</b>	<b>396,59</b>	<b>72,58</b>	<b>1,37</b>
	2	30,73	<b>75,4</b>	75,52	350,91	71,56	2,12
	3	27,15	64,3	91,17	389,56	70,29	2,41
	4	42,17	75,1	51,04	393,80	71,37	1,47
	5	6,14	17,1	367,41	189,43	16,31	5,94
7	1	<b>43,11</b>	<b>70,6</b>	<b>61,32</b>	285,85	67,25	<b>1,47</b>
	2	23,96	64,4	110,72	222,69	61,04	2,43
	3	28,43	61,5	90,03	<b>292,59</b>	<b>67,57</b>	2,16
	4	40,97	69,7	62,40	282,51	66,85	1,55
	5	9,06	17,5	360,05	139,05	16,32	4,13
8	1	<b>36,52</b>	<b>67,3</b>	<b>68,86</b>	308,50	63,67	<b>1,66</b>
	2	18,10	64,5	117,88	234,49	60,63	2,71
	3	22,51	57,7	111,66	<b>321,25</b>	<b>64,44</b>	2,42
	4	35,59	66,4	72,65	306,61	63,20	1,70
	5	8,13	20,8	331,87	184,27	19,15	4,76