

**Faculdade de Engenharia da Universidade do Porto**



**FEUP**

**Sistema de controlo de manobras para um  
veículo aéreo não tripulado**

Ricardo Joel da Mota Gomes

Dissertação de Projecto realizada no âmbito do  
Mestrado Integrado em Engenharia Electrotécnica e de Computadores  
Major Automação

Orientador: Prof. João Sousa

Junho de 2009



# Resumo

Actualmente, a área de investigação no controlo de veículos aéreos não tripulados (UAV) tem crescido imenso. As suas aplicações vão desde a monitorização do tráfego, detecção de fogos, procura e salvamento, transporte, investigação (atmosférica, ambiental), seguimento de estruturas, entre muitas outras áreas.

Este trabalho foca-se na missão de seguimento de estruturas naturais, com destaque no seguimento de rios. A localização da estrutura é obtida pelo processamento de imagem de vídeo, onde são extraídos pontos que localizam a estrutura.

Através dos pontos são desenvolvidos algoritmos de geração de caminhos e métodos para obter caminhos viáveis. Neste caso, as restrições da velocidade e da velocidade de volta são consideradas, pois influenciam o raio de curvatura do UAV.

Numa fase seguinte, são analisadas e projectadas soluções para problemas como o seguimento do caminho, quer a velocidade constante, quer variável; e o seguimento de um caminho não viável, onde é necessário executar um conjunto de manobras para manter a estrutura dentro do campo de visão da câmara. A projecção do campo de visão da câmara é analisada tendo em conta os ângulos *pitch*, *bank* e *yaw* do UAV, e os ângulos *pan* e *tilt* da câmara.

As várias soluções e controladores desenvolvidos, baseados na teoria de *sliding mode control*, são simulados através do Matlab/Simulink<sup>TM</sup> e *Stateflow* de forma a demonstrar a sua performance.



# Abstract

Currently, the research area in the control of unmanned aerial vehicles (UAVs) has grown enormously. Its applications ranging from traffic monitoring, detection of fires, search and rescue, transport, research (atmospheric, environmental), monitoring and follow of natural structures, among many other areas.

This work focuses on the task of follow of natural structures, with emphasis on tracking and evolving rivers. The structure location is given by points extracted from a video image.

Through the points are developed algorithms to generate paths and methods to obtain feasible paths. In this case, velocity and turn rate restrictions are considered, because the influence in radius of curvature of UAVs.

In the next step, solutions to problems such as tracking and the evolution of the path with constant velocity and velocity adjustment are analyzed and projected. When the path is not feasible is necessary to run a set of maneuvers, so that structure is kept within the field of view of the camera. The projection of camera field of view is analyzed taking into account the pitch, bank and yaw angles of UAV, and pan, tilt angles of camera.

The various controllers and solutions developed, based on the theory of sliding mode control, are simulated using Matlab/Simulink<sup>TM</sup> and Stateflow in order to demonstrate its performance.



*A todos aqueles que tornaram possível este momento*



# Índice

Resumo .....	i
Abstract .....	iii
Índice .....	vii
Lista de Figuras .....	ix
Lista de Tabelas .....	xiii
Abreviaturas e Símbolos .....	xv
<b>Capítulo 1 .....</b>	<b>1</b>
Introdução .....	1
1.1 - Objectivos do trabalho.....	3
1.2 - Âmbito do trabalho.....	3
1.3 - Estrutura do documento .....	4
<b>Capítulo 2 .....</b>	<b>5</b>
Revisão bibliográfica .....	5
2.1 - Modelo geral não linear.....	5
2.2 - Teoria de Lyapunov.....	5
2.2.1 - Estabilidade em sistemas contínuos .....	5
2.2.2 - Funções de Lyapunov .....	7
2.2.3 - Função de controlo de Lyapunov.....	8
2.3 - Controlo por Sliding Mode .....	9
2.3.1 - Exemplos.....	12
2.3.2 - Condições “ <i>matched</i> e <i>unmatched</i> ” .....	13
2.3.3 - Múltiplas <i>sliding surfaces</i> .....	13
2.3.4 - <i>Dynamic Surface Control</i> e relação com o <i>Integrator Backstepping</i> .....	13
2.4 - Modelos de sistemas híbridos e o seu controlo.....	14
2.4.1 - Sistemas híbridos .....	14
2.4.2 - Estabilidade em sistemas híbridos .....	14
2.5 - Modelo de uma câmara .....	16
2.6 - Aproximação de pontos .....	17
2.6.1 - Aproximação por um polinómio .....	17
2.6.2 - Interpolação por uma <i>spline</i> .....	18
2.6.3 - Cálculo da curvatura.....	18
<b>Capítulo 3 .....</b>	<b>21</b>
Trabalho relacionado .....	21

3.1 - Planeamento do caminho .....	21
3.2 - Seguimento de estruturas .....	22
<b>Capítulo 4 .....</b>	<b>29</b>
Formulação dos problemas.....	29
4.1 - Modelo 6-dof .....	29
4.2 - Modelo simplificado do UAV .....	32
4.3 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo .....	34
4.4 - Seguimento de uma curva viável com velocidade constante .....	35
4.5 - Seguimento de uma curva viável com ajustamento da velocidade.....	35
4.6 - Seguimento de uma curva não viável .....	36
<b>Capítulo 5 .....</b>	<b>37</b>
Abordagem .....	37
5.1 - Projecção do campo de visão da câmara .....	37
5.2 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo .....	39
5.2.1 - Aproximação por um polinómio.....	39
5.2.2 - Interpolação por uma <i>spline</i> cúbica.....	41
5.3 - Seguimento de uma curva viável com velocidade constante .....	44
5.4 - Seguimento de uma curva viável com ajustamento da velocidade.....	48
5.5 - Seguimento de uma curva não viável .....	48
<b>Capítulo 6 .....</b>	<b>59</b>
Simulação e resultados .....	59
6.1 - Ambiente de simulação.....	59
6.2 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo .....	60
6.3 - Seguimento de uma curva viável com velocidade constante .....	61
6.4 - Seguimento de uma curva viável com ajustamento da velocidade.....	63
6.5 - Seguimento de uma curva não viável .....	64
<b>Capítulo 7 .....</b>	<b>71</b>
Conclusões e trabalho futuro .....	71
7.1 - Conclusões.....	71
7.2 - Trabalho futuro .....	72
Anexo A Diagrama de blocos geral do Simulink.....	73
Anexo B Diagrama de estados do controlador híbrido em Stateflow.....	74
Anexo C Modelo do UAV .....	75
Anexo D Modelo do vento .....	76
<b>Referências.....</b>	<b>77</b>

# Lista de Figuras

Figura 1-1- MQ-1 Predador e RQ-4 Global Hawk. ....	1
Figura 1-2 - Aerosonde. ....	2
Figura 2-1- Ilustração da definição de estabilidade [4]. ....	6
Figura 2-2- Ilustração da definição de estabilidade assintótica [4]. ....	6
Figura 2-3- Ilustração da definição de instabilidade [4]. ....	7
Figura 2-4 - Comportamento do método <i>sliding mode control</i> . ....	9
Figura 2-5 - Função sigmóide (sign). ....	11
Figura 2-6 - Região limitadora. ....	11
Figura 2-7 - Função de saturação. ....	11
Figura 2-8- Sistema híbrido $H$ [3]. ....	14
Figura 2-9 - Geometria do modelo <i>pinhole</i> com o plano de imagem à frente e atrás do centro óptico, respectivamente i) e ii). ....	16
Figura 2-10 - A interpolação por um polinómio de grau elevado é mais precisa (linha) do que um de menor grau (tracejado) [11]. ....	17
Figura 2-11 - Aproximação polinomial de 2º grau (a), 3º grau (b), 4º grau (c), 5º grau (d) [12]. ....	18
Figura 2-12 - Derivada do vector unitário tangente (vermelho). ....	18
Figura 3-1 - Procedimento de inserção da curva. ....	21
Figura 3-2 - Simulação do seguimento de um veículo terrestre com velocidade variável [16]. ....	22
Figura 3-3 - Diagrama de estados do sistema de exploração [15]. ....	22
Figura 3-4 - Ilustração da curva a ser seguida e o polinómio de ligação [17]. ....	23
Figura 3-5 - Ilustração dos vectores de campo para uma recta e para uma órbita. ....	24
Figura 3-6 - Modelo cinemático do UAV [21]. ....	25

Figura 3-7 - Referenciais. ....	26
Figura 3-8 - Função sigmoide que relaciona o curso desejado com a distância ao caminho. ..	26
Figura 3-9 - Simulação passo a passo do algoritmo com ponto de referência [23].....	27
Figura 4-1 - Notação das variáveis das equações de movimento [27].....	29
Figura 4-2 - Representação das forças existentes numa volta coordenada. ....	32
Figura 4-3 - Estrutura da informação do algoritmo de controlo do seguimento da estrutura. ....	35
Figura 4-4 - Estrutura da informação do algoritmo de controlo do seguimento da estrutura, com ajustamento da velocidade. ....	36
Figura 5-1 - Geometria do cálculo das coordenadas da projecção do campo de visão da câmara.....	39
Figura 5-2 - Aproximação por polinómio de grau 3, 4 e 6.....	40
Figura 5-3 - Método de alteração do caminho, com aproximações por polinómios.....	40
Figura 5-4 - Aproximação iterativa por polinómios. ....	41
Figura 5-5 - Aproximação por <i>spline</i> cúbica.....	42
Figura 5-6 - Fases da construção da curva viável, por uma <i>spline</i> cúbica, tendo em conta a restrição de um raio de curvatura mínimo de 80m.....	42
Figura 5-7 - Sequência da interpolação de pontos. ....	43
Figura 5-8 - Notação das distâncias. ....	44
Figura 5-9 - Orientação desejada em função da distância $y_{track}$ . ....	45
Figura 5-10 - Ponto de referência. ....	45
Figura 5-11 - Referencial do caminho.....	46
Figura 5-12 - Estrutura de informação do sistema híbrido (modos de operação, parâmetros e condições de transição). ....	49
Figura 5-13 - Estrutura da informação do algoritmo de controlo do seguimento de órbitas. ..	50
Figura 5-14 - Geometria do problema de seguimento de uma órbita circular [18]. ....	50
Figura 5-15 - Diagrama de funcionamento do modo <i>search</i> . ....	53
Figura 5-16 - Área pesquisada pelo UAV em órbitas circulares. Na situação i) a área entre órbitas é pesquisada, enquanto na situação ii) a área entre órbitas não é pesquisada completamente. ....	54
Figura 5-17 - Ângulos <i>tilt</i> e <i>pan</i> da câmara. ....	54
Figura 5-18 - Campo de visão da câmara [29]. ....	55

Figura 5-19 - Modo interceptação. Em 1) transita para a fase de <i>Outbound</i> , em 2) entra na fase de <i>Inbound</i> e em 3) transita para o modo de seguimento do caminho de referência. ....	56
Figura 6-1 - Teste do controlador de seguimento de curvas para várias distâncias do ponto de referência. ....	61
Figura 6-2 - Teste do controlador de seguimento de curvas na presença de vento.....	62
Figura 6-3 - Teste do controlador de seguimento de curvas para vários valores do parâmetro de eliminação de <i>chattering</i> .....	62
Figura 6-4 - Comparação da trajectória com velocidade constante e com ajustamento. ....	63
Figura 6-5 - Velocidade comandada. ....	63
Figura 6-6 - Testes efectuados ao controlador de seguimento de órbitas.....	64
Figura 6-7 - Seguimento de órbita com vento de 15m/s vindo da direcção 180 graus. ....	65
Figura 6-8 - Projecção do campo de visão da câmara para vários ângulos de <i>pitch</i> , <i>bank</i> , <i>yaw</i> , <i>tilt</i> e <i>pan</i> .....	66
Figura 6-9 - Sequência de campos de visão à altura de 100m e velocidade de 20m/s, a começar com uma órbita de raio 100m (verde) e as órbitas seguintes com raios calculados por (5.23) (vermelho e azul). ....	66
Figura 6-10 - Modo interceptação para vários ganhos de distância e com a distância ao ponto de interceptação (centro da órbita) de 300m.....	67
Figura 6-11 - Modo de interceptação para vários ângulos de interceptação. ....	67
Figura 6-12 - Seguimento de um caminho com curvas não viáveis, para uma altura de 100m e uma distância ao ponto de interceptação de 300m. ....	69
Figura 6-13 - Seguimento de um caminho com curvas não viáveis, para uma altura de 100m e uma distância ao ponto de interceptação de 200m. ....	69
Figura 6-14 - Seguimento de um caminho com curvas não viáveis, para uma altura de 200m e uma distância ao ponto de interceptação de 200m .....	70



## Lista de Tabelas

Tabela 1 - Sumário das variáveis de movimento envolvidas. ....	30
Tabela 2 - Transformações homogêneas. ....	37



# Abreviaturas e Símbolos

## Lista de abreviaturas

UAV	<i>Unmanned Aerial Vehicle</i>
FAP	Força Aérea Portuguesa
SMC	<i>Sliding Mode Control</i>
SISO	<i>Single Input Single Output</i>
DSC	<i>Dynamic Surface Control</i>
MSS	<i>Multiple Sliding Surfaces</i>
FOV	<i>Field of View</i>
DOF	<i>Degree of Freedom</i>
PITVANT	Projecto de Investigação e Tecnologia em Veículos Aéreos Não Tripulados



# Capítulo 1

## Introdução

Os veículos aéreos não tripulados (UAV) começaram a ser usados nos anos 80, inicialmente focados em aplicações militares. Desde então, a sua utilização tem crescido e expandido para aplicações civis.

Fonte: Wikipédia



Figura 1-1- MQ-1 Predator e RQ-4 Global Hawk.

Relativamente a aplicações militares, os UAVs podem fornecer dados a uma coluna militar para indicar se o caminho à sua frente está livre de perigo e serem utilizados em missões de combate (Figura 1-1), onde o risco para os humanos é elevado.

As suas aplicações civis vão desde a monitorização (ex.: tráfego, oleodutos), detecção de fogos, procura e salvamento, transporte, investigação (atmosférica, ambiental) (Figura 1-2), seguimento de estruturas naturais (ex.: rios, estradas), missões de vigilância (ex.: patrulhamento de fronteiras), entre muitas outras.

A operação de UAVs por longos períodos de tempo e a baixa altitude permite a execução de missões de monitorização em grande escala, sem o risco de perdas humanas. A investigação de furacões é um desses casos. São missões que envolvem riscos grandes para

pilotos e investigadores, em que a recolha de dados através de sensores especiais ajuda na previsão da trajectória e intensidade do furacão.

Fonte: *Wikimedia*



Figura 1-2 - Aerosonde.

Devido à evolução da tecnologia, nos dias que correm encontram-se sensores e sistemas de controlo que realizam o controlo de baixo nível de forma eficiente e com baixos custos, e assim pode dar-se mais ênfase a sistemas de controlo de alto nível, desde o planeamento de caminhos, a algoritmos de seguimento de estruturas mais robustos.

Com o desenvolvimento de algoritmos de alto nível pode tirar-se partido das capacidades dos veículos autónomos de forma mais eficiente.

Por exemplo, um dos grandes objectivos é que exista apenas um operador a controlar vários UAV's, o que não acontece actualmente. Pelo contrário, grande parte dos UAV's necessitam de várias pessoas para o operar e apenas voam "completamente" autónomos em ambientes simples e pré-planeados. Esta redução no número de operadores permite, por sua vez, reduzir os custos de operação associados.

Grande parte dos pilotos automáticos comerciais para controlo de UAVs funciona com um sistema de navegação baseado em *waypoints*. Contudo, este método de navegação não é o mais adequado para o seguimento de estruturas naturais, pois necessita de longos segmentos de rectas, e não garante que a estrutura se mantenha no campo de visão da câmara.

Este trabalho surge com este problema em mente: projectar um sistema de controlo de manobras para executar o seguimento de estruturas naturais, com ênfase no seguimento de rios, baseado em imagens de vídeo captadas.

O seguimento de estruturas naturais é bastante importante em missões de monitorização e reconhecimento, pois permite uma aquisição de imagens em tempo real e, até mesmo, realizar a monitorização quando esta não é possível com seres humanos. Por exemplo, a monitorização do tráfego permite saber o estado das estradas em tempo real e planear uma rota óptima em caso de emergência.

Este trabalho insere-se num projecto de colaboração em curso, entre a Universidade do Porto e a Força Aérea Portuguesa, denominado por PITVANT (Projecto de Investigação e Tecnologia em Veículos Aéreos Não Tripulados). O projecto consiste no desenvolvimento e operação de várias plataformas de pequena e média dimensão, comandadas e controladas por duas *frameworks*, Dune e Neptus [1], respectivamente, que interagem com os pilotos automáticos Piccolo e Micropilot.

Alguns dos objectivos deste projecto são:

- Desenvolver *know-how* nacional sobre UAVs;
- Formação de operadores de UAVs;
- Desenvolver tecnologia para UAVs;
- Demonstração de sistemas e tecnologia em missões militares/civis;
- Desenvolver conceitos de operação (ex.: procedimentos).

## 1.1 - Objectivos do trabalho

Neste trabalho é necessário o desenvolvimento de controladores para várias fases do voo e o planeamento de caminhos. Seguem-se os objectivos seleccionados:

- Desenvolvimento de um algoritmo de geração de caminhos, a partir de pontos que localizam a estrutura natural (fornecidos pelo processamento da imagem de vídeo). O algoritmo deve ter em conta a restrição imposta pelo UAV (raio de curvatura);
- Implementar um modelo 3-dof do UAV;
- Projectar um controlador *sliding mode* (SM) para seguimento de uma estrutura;
- Estudar a projecção do campo de visão da câmara do veículo no plano horizontal onde está a estrutura (analisar tendo em conta os ângulos de *roll* e *yaw*);
- Formular soluções para situações em que o raio de curvatura mínimo exigido ultrapassa a capacidade do UAV;
- Implementar um controlador híbrido, para que perante um caminho não viável a estrutura continue dentro do campo de visão da câmara.

## 1.2 - Âmbito do trabalho

Este trabalho insere-se na disciplina de Dissertação do Mestrado Integrado em Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto, no 2º semestre do ano lectivo de 2008/2009.

### 1.3 - Estrutura do documento

O documento está estruturado sob a forma de 7 capítulos. O presente capítulo faz uma introdução ao tema em questão e uma apresentação dos objectivos deste trabalho.

No capítulo 2 é realizada uma revisão da literatura sobre vários temas relacionados com o projecto, desde o modelo não linear, passando pela teoria de Lyapunov, pela teoria de *sliding mode control*, dos sistemas híbridos, do modelo da câmara e da aproximação de pontos.

O capítulo 3 é dedicado a trabalhos relacionados existentes, desde o planeamento de caminhos, aos vários tipos de controlo utilizados para seguimento de caminhos.

No capítulo 4 são apresentadas as formulações dos problemas a resolver e o modelo do UAV utilizado.

No capítulo 5 é descrita a abordagem seguida na resolução de cada problema: os controladores projectados baseados em *sliding mode control* e sistemas híbridos, e demonstração da estabilidade.

No capítulo 6 é descrito o ambiente de simulação e realizada uma análise dos resultados obtidos, para os vários testes efectuados.

No capítulo 7 apresentam-se as conclusões deste trabalho, os objectivos que se atingiram e sugestões para desenvolvimento futuro.

# Capítulo 2

## Revisão bibliográfica

Neste capítulo são abordados temas que serão necessários posteriormente para a resolução dos problemas.

Começa com a apresentação do modelo geral não linear, em seguida a teoria de Lyapunov, técnicas de *sliding mode control* [2], sistemas híbridos e o seu controlo [3], modelo *pinhole* de uma câmara e acaba com o tema da aproximação de dados por uma curva (*curve fitting*).

### 2.1 - Modelo geral não linear

Um sistema dinâmico não linear pode ser representado por um conjunto de equações diferenciais não lineares, na seguinte forma,

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad \mathbf{x} \in \mathbb{R}^n \quad (2.1)$$

Onde  $f$  é um vector  $n \times 1$  de funções não lineares, e  $\mathbf{x}$  é o vector  $n \times 1$  de estados. O número de estados define a *ordem* do sistema.

### 2.2 - Teoria de Lyapunov

#### 2.2.1 - Estabilidade em sistemas contínuos

O estudo da estabilidade de um sistema é dos pontos mais importantes, porque sistemas instáveis podem causar danos graves.

A análise de estabilidade segundo Lyapunov é um dos métodos usados na análise de estabilidade de sistemas não lineares.

As próximas definições caracterizam o comportamento do sistema quando aplicadas localmente, ou seja, quando as condições iniciais estão “próximas” do ponto de equilíbrio.

**Definição 1 (Ponto de equilíbrio):** Considerando o sistema dinâmico (2.1), um ponto  $x^* \in \mathbb{R}^n$  é um ponto de equilíbrio do sistema dado por (2.1) em que  $f(x^*, t) \equiv 0$ .

**Definição 2 (Ponto de equilíbrio estável):** O ponto de equilíbrio  $x^*$  é estável, segundo Lyapunov, em  $t=t_0$  se

$$\|x(t_0)\| < \delta \Rightarrow \|x(t)\| < \varepsilon \quad \forall t \geq t_0 \quad (2.2)$$

Ou seja, é estável se para condições iniciais “próximas” de  $x^*$ , as soluções permanecem “perto” de  $x^*$ , para  $t \rightarrow \infty$  (Figura 2-1).

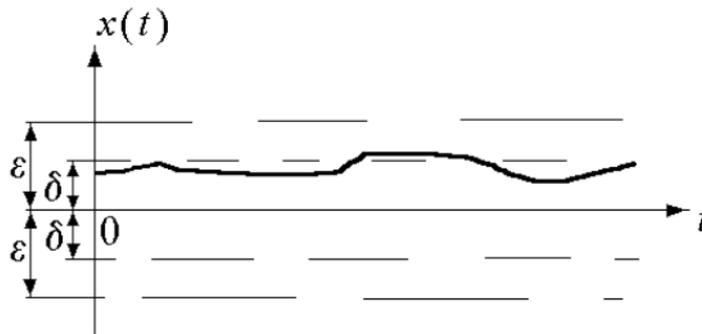


Figura 2-1- Ilustração da definição de estabilidade [4].

**Definição 3 (Ponto de equilíbrio assintoticamente estável):** O ponto de equilíbrio é dito assintoticamente estável, se for localmente estável e todas as soluções que começam “perto” de  $x^*$  tendem para  $x^*$  com  $t \rightarrow \infty$  (Figura 2-2), isto é

$$\delta > 0 \quad \|x(t_0)\| < \delta \Rightarrow \lim_{t \rightarrow \infty} x(t) = 0$$

Existe um **domínio de atractividade** que nos diz que se o estado inicial estiver contido neste, então a solução aproxima-se do ponto de equilíbrio.

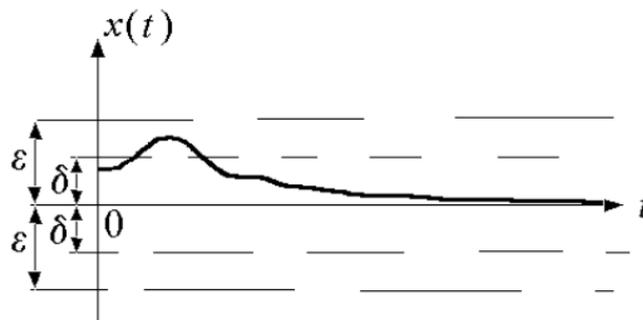


Figura 2-2- Ilustração da definição de estabilidade assintótica [4].

**Definição 4 (Ponto de equilíbrio instável):** O ponto de equilíbrio é instável se existir um estado  $x_0$  tal que a sua trajectória ultrapassa a distância  $\varepsilon$  (Figura 2-3).

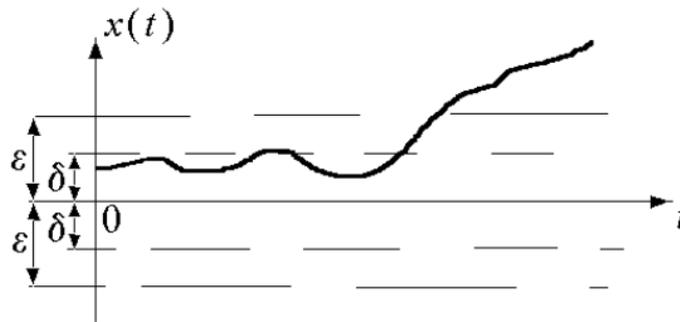


Figura 2-3- Ilustração da definição de instabilidade [4].

Considerando como exemplo um pêndulo, descrito pela seguinte equação:

$$\ddot{\theta} + \frac{g}{l} \sin \theta = 0,$$

Este tem um ponto de equilíbrio localmente estável quando está na posição vertical em baixo, e um ponto de equilíbrio instável quando está na posição vertical em cima. Se o pêndulo oscilar, a posição vertical em baixo torna-se um ponto de equilíbrio assintoticamente estável.

A estabilidade assintótica não nos diz nada relativamente ao tempo que demora a convergir para uma vizinhança do ponto de equilíbrio. Para colmatar esse problema usa-se o conceito de **estabilidade exponencial**, que nos fornece uma razão de convergência da trajectória do sistema.

**Definição 5 (Ponto de equilíbrio exponencialmente estável):** O ponto de equilíbrio é exponencialmente estável, se for assintoticamente estável e se existir um  $\lambda$  e  $\alpha$  positivos, tal que

$$\|x(0)\| < \delta \quad \|x(t)\| \leq \alpha \|x(0)\| e^{-\lambda t} \quad \forall t > 0$$

Isto diz-nos que as soluções convergem a uma velocidade superior a uma função exponencial.

### 2.2.2 - Funções de Lyapunov

O segundo método de Lyapunov permite-nos determinar a estabilidade do sistema sem ser necessário integrar o sistema dado por (2.1). O método baseia-se no conceito de “energia” de um sistema. Se se conseguir medir a variação desta “energia” é possível tirar conclusões sobre a estabilidade do sistema. Contudo, para a maior parte dos sistemas é difícil definir

uma função de “energia” e para contornar esta dificuldade Lyapunov criou a função de Lyapunov  $V(x)$ .

**Definição 6 (Função quadrática):** A função escalar  $V(x) = x^2$  é uma função na forma quadrática e é positiva definida se e apenas se

$$V(x) > 0 \quad \forall x \neq 0.$$

**Definição 7 (Função candidata de Lyapunov):** Seja  $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função contínua com primeira derivada parcial contínua e supondo  $V(x)$  positiva definida. Então  $V(x)$  é uma função candidata de Lyapunov para o sistema dado por (2.1).

**Teorema 1.** O sistema dado por (2.1) é dito assintoticamente estável se existir uma função  $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  tal que

$$\begin{aligned} V(x) &\geq 0 \\ \dot{V}(x(t)) &< 0 \end{aligned} \tag{2.3}$$

onde  $V(x)$  é uma função escalar contínua e positiva definida e a sua derivada é negativa definida.

Tome-se como exemplo um sistema mecânico (mola, pêndulo), numa condição inicial que não é o seu ponto de equilíbrio. Devido ao atrito, a energia vai diminuir com o tempo, e isto quer dizer que  $\dot{V}(x(t)) < 0$ , logo o sistema irá alcançar um ponto de repouso (ponto de equilíbrio).

Em resumo, o segundo método de Lyapunov consiste em dois passos:

- Determinar uma função escalar  $V(x)$ ;
- Calcular a 1ª derivada de  $V(x)$ .

Se a 1ª derivada diminui enquanto o tempo aumenta, então a energia do sistema está a diminuir e, eventualmente, o sistema irá chegar a um ponto de repouso (equilíbrio).

### 2.2.3 - Função de controlo de Lyapunov

Considerando um sistema não linear que satisfaz:

$$\dot{x}(t) = f(\mathbf{x}, t) + g(\mathbf{x}, t)u(t) \tag{2.4}$$

A função de controlo de Lyapunov (CLF)  $V(x, u)$  é uma generalização da noção de função de Lyapunov na análise de estabilidade, referida na secção anterior, para sistemas com entradas.

**Teorema 2.** A CLF testa se um sistema dado por (2.4) é estável. Isto é verdade se para qualquer estado existir um controlo  $u(x,t)$  que torna a derivada de  $V(x,u)$  negativa (a “energia” do sistema diminui). Traduz-se matematicamente por,

$$V(x,u) \geq 0 \quad \dot{V}(x,u) < 0 \quad \forall x \neq 0, \exists u$$

### 2.3 - Controlo por Sliding Mode

O modelo matemático de qualquer sistema, e em particular de um sistema não linear, isto é, um sistema descrito por uma equação diferencial não linear, é sempre um modelo matemático e por isso uma aproximação da realidade. Primeiro, porque o modelo pode resultar de aproximações, segundo, porque os parâmetros do sistema real podem não ser exactamente conhecidos ou variar com o tempo, terceiro, porque pode haver dinâmicas não modeladas e quarto porque os efeitos de perturbações externas podem não ser bem conhecidos. Apesar destes problemas, é fundamental que o controlador continue a ter uma performance aceitável.

Uma das formas de solucionar estes problemas é através de controlo robusto, sendo o *sliding mode control* uma das metodologias mais usadas.

O objectivo é conduzir (*reaching phase*) a trajectória do sistema para uma superfície desejada (*sliding surface*) no espaço de estados e manter esta trajectória (*sliding mode*) (Figura 2-4).

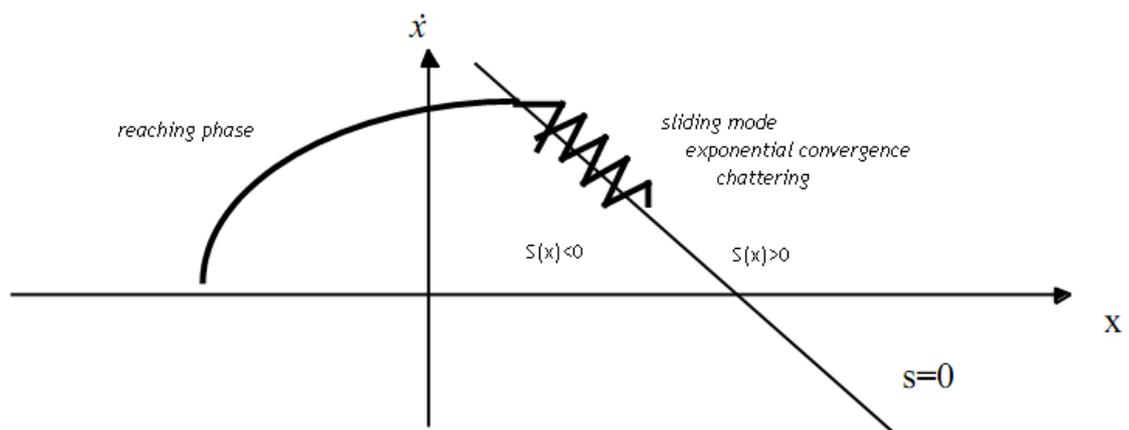


Figura 2-4 - Comportamento do método *sliding mode control*.

Considere o sistema dinâmico SISO (*Single Input Single Output*):

$$x^{(n)} = f(\mathbf{x}) + b(\mathbf{x})u, \tag{2.5}$$

onde  $x$  é o valor escalar da saída,  $u$  é o valor escalar da variável de controlo e  $\mathbf{x}$  é o vector de estados. As funções  $f(\mathbf{x})$  e  $b(\mathbf{x})$ , geralmente não lineares, não são exactamente conhecidas, mas as suas imprecisões são limitadas por funções contínuas de  $\mathbf{x}$  [2].

A implementação deste método de controlo consiste, principalmente, em três fases:

- **Escolha da superfície de deslizamento:** esta superfície representa o comportamento dinâmico desejado do sistema. Pode ser representada pela equação geral (2.6), assegurando a convergência da variável para o valor desejado [2].

$$S(x;t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x} \quad (2.6)$$

$$\tilde{x} = x - x_d$$

$\lambda$  : é uma constante positiva que interpreta a largura de banda do controlo desejado.

$n$  : grau relativo, igual ao número de vezes que é necessário derivar a saída para obter a variável de controlo.

$S(x;t)=0$  é uma equação diferencial com a solução singular  $\tilde{x}=0$ .

- **Estabelecimento das condições de convergência:** condições que fazem com que o sistema convirja para a *sliding surface*. Definindo uma função de Lyapunov escalar positiva  $V>0$  (2.7).

$$V(s) = \frac{1}{2} s^2 \quad (2.7)$$

Para testar a estabilidade, ou seja, a convergência para a superfície, a primeira derivada de (2.7) deve ser negativa definida,

$$\dot{V}(s) = s\dot{s} < 0. \quad (2.8)$$

- **Escolha da lei de controlo:** com o objectivo que o estado do sistema permaneça na fase *sliding mode*, projecta-se uma lei que satisfaz as equações

$$\begin{aligned} s(x) &= 0 \\ \dot{s}(x) &= 0 \end{aligned} \quad (2.9)$$

Esta lei contém uma componente de baixa frequência  $u_{eq}$  (valor contínuo que mantém  $\dot{s} = 0$ ) e uma componente de alta frequência  $u_n$  (assegura que qualquer estado fora da *sliding surface* irá convergir para esta), expressa por

$$u(t) = u_{eq}(t) + u_n. \quad (2.10)$$

Onde  $u_n$  é um termo descontínuo, por exemplo, a função sigmóide (Figura 2-5).

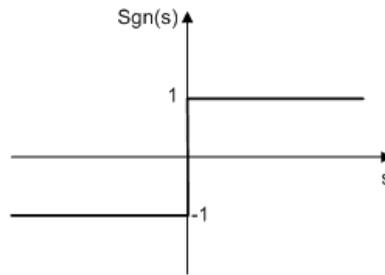


Figura 2-5 - Função sigmóide (sign).

A componente de alta frequência (descontínua) que é inserida para “combater” a presença de perturbações e de incertezas na modelação, provoca um comportamento descrito como *chattering*.

Este comportamento é indesejado porque provoca reduções na precisão do controlo, perdas por calor elevadas em circuitos eléctricos e vibrações altas nas partes mecânicas móveis. Pode ainda excitar modos ressonantes, provocando instabilidade no sistema.

Uma das formas de eliminar este problema é a inserção de uma região limitadora em redor da *sliding surface* (Figura 2-6) [2].

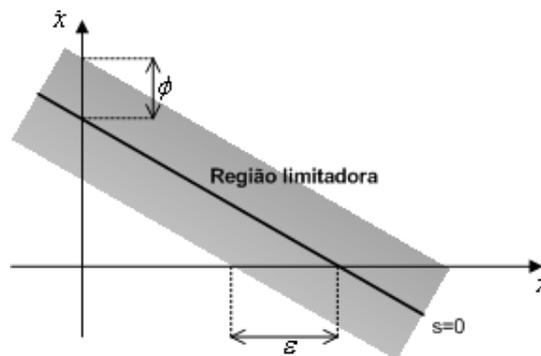


Figura 2-6 - Região limitadora.

Onde  $\phi$  espessura da região e  $\epsilon = \phi / \lambda^{n-1}$  é a largura da região limitadora. Substituindo na função *sign* por  $s / \phi$ , obtemos uma função de controlo mais “suave” (Figura 2-7).

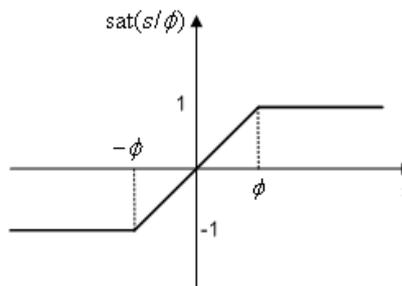


Figura 2-7 - Função de saturação.

### Ideia do Sliding Mode Control:

Transformar um sistema de ordem  $n$  num sistema de 1ª ordem, pois este é mais fácil de controlar, i.e. “se a saída é alta, então aplicamos um controlo para baixo, e vice-versa, se a saída é baixa, então aplicamos um controlo para cima” [5].

A redução para um sistema de 1ª ordem é feita através da definição de uma nova variável de “saída”,  $s$ , com as seguintes propriedades:

- O grau de  $s$  deve ser 1, isto é, a variável de controlo aparece na equação da derivada de  $s$ .
- O objectivo é “levar” a variável  $s$  a zero,  $s \rightarrow 0$ .

### 2.3.1 - Exemplos

#### Integrador Simples

$$\begin{cases} \dot{x} = u \\ y = x \end{cases} \quad (2.11)$$

O objectivo é  $y \rightarrow 0$ .

Escolhendo a lei de controlo:  $u = -k \cdot x$

Pode usar-se qualquer ganho  $k$ , sem que o sistema se torne instável.

Fazendo  $s = x$ , ao derivar obtemos a variável de controlo  $u$ .

#### Integrador Duplo

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = u \\ y = x_1 \end{cases} \quad (2.12)$$

O objectivo é  $y \rightarrow 0$ .

Escolhendo:

$$s(x_1, x_2) = x_2 + \lambda x_1 = \dot{x}_1 + \lambda x_1 \quad \lambda > 0 \quad (2.13)$$

Igualando a zero,

$$s = 0 \Rightarrow \dot{x}_1 + \lambda x_1 = 0 \Rightarrow \dot{x}_1 = -\lambda x_1$$

Isto implica que  $x_1 \rightarrow 0$  exponencialmente.

Fazendo a derivada de (2.13), obtêm-se

$$\dot{s}(x_1, x_2) = \dot{x}_2 + \lambda \dot{x}_1 = u + \lambda x_2. \quad (2.14)$$

Escolhendo  $u = -\lambda x_2 - \text{sgn}(s)$ .

### 2.3.2 - Condições “*matched* e *unmatched*”

Um sistema não linear com incertezas em condições de “matching” é definido matematicamente pela seguinte definição [6].

$$\dot{x} = f(x,t) + \Delta f(x,t) + B(x,t)u + \Delta B(x,t)u \quad (2.15)$$

Onde  $\Delta f(x,t)$  e  $\Delta B(x,t)$  correspondem às incertezas de  $f(x,t)$  e  $B(x,t)$ , respectivamente. O sistema satisfaz as condições de *matching* se

$$\Delta f(x,t) = B(x,t)\Delta f'(x,t) \quad \Delta B(x,t) = B(x,t)\Delta B'(x,t)$$

e se existir uma constante positiva  $\varepsilon$  tal que,

$$\Delta B'(x,t) \leq 1 - \varepsilon$$

Assim, o sistema dado por (2.15) pode ser escrito como:

$$\dot{x} = f(x,t) + B(x,t)[\Delta f'(x,t) + (1 + \Delta B'(x,t))u(x,t)].$$

Na equação anterior pode verificar-se que a incerteza entra no sistema pelo mesmo canal que a variável de controlo  $u$ , logo pode ser “combatida” pela variável de controlo. Caso contrário, estamos perante condições *unmatched*.

### 2.3.3 - Múltiplas *sliding surfaces*

O método de múltiplas *sliding surfaces* surge para colmatar o problema do método de *sliding mode control* não se aplicar a sistemas em condições *unmatched*.

A ideia principal do método é considerar algumas variáveis de estado como variáveis de controlo fictícias e estabilizar cada *sliding surface* passo a passo, começando pela última *sliding surface* [7].

### 2.3.4 - *Dynamic Surface Control* e relação com o *Integrator Backstepping*

O *Integrator Backstepping* é um método passo-a-passo para resolver problemas em que existem incertezas não lineares “*unmatched*”. No entanto, é necessário derivar sucessivamente funções não lineares e estabelecer um limite nas funções que representam as incertezas.

Este problema levou ao desenvolvimento dos métodos *Dynamic Surface Control* (DSC) e *Multiple Sliding Surfaces* (MSS) [8], que eliminam a necessidade de derivar o modelo do sistema, através da passagem da variável desejada por um banco de filtros de 1ª ordem.

## 2.4 - Modelos de sistemas híbridos e o seu controlo

### 2.4.1 - Sistemas híbridos

Os sistemas híbridos combinam as características dos sistemas contínuos com as características dos sistemas discretos (Figura 2-8). Os vários modos (dinâmicas diferentes) são controlados através de um conjunto de transições.

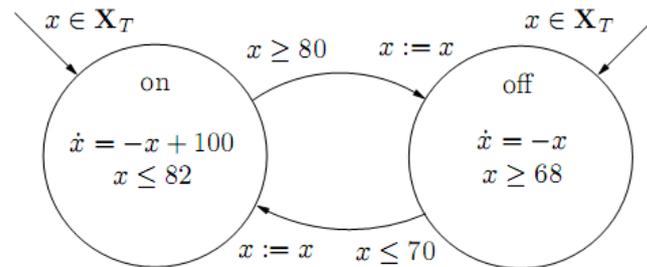


Figura 2-8- Sistema híbrido  $H$  [3].

**Definição 8 (Sistema híbrido):** O sistema híbrido  $H$  é um conjunto de  $H=(Q, X, \text{Init}, f, \text{Dom}, R)$  onde,

- $Q = \{q_1, q_2, \dots\}$  é o conjunto de estados discretos;
- $X = \mathbb{R}^n$  é o conjunto de estados contínuos;
- $\text{Init} \subseteq Q \times X$  é o conjunto de estados iniciais;
- $f(\cdot, \cdot): Q \times X \rightarrow \mathbb{R}^n$  é um vector de campo;
- $\text{Dom}(\cdot): Q \rightarrow 2^X$  é um domínio;
- $R(\cdot, \cdot): Q \times X \rightarrow 2^X$  é uma relação de reset.

Ou seja, o estado de um sistema híbrido pode evoluir de várias formas. Começando num valor inicial  $(q_0, x_0) \in \text{Init}$ , o estado contínuo evolui conforme a equação diferencial  $\dot{x} = f(q_0, x)$ ,  $x(0) = x_0$ , enquanto o estado discreto,  $q$ , permanece constante em  $q_0$ . Esta evolução continua enquanto  $x(t) \in \text{Dom}(q_0)$ . Se em algum ponto  $R(\cdot, \cdot) \neq \emptyset$ , então uma transição é executada. Durante a transição o estado contínuo e o discreto podem tomar um valor em  $R(q_0, x)$ . Depois da transição, o processo repete-se.

### 2.4.2 - Estabilidade em sistemas híbridos

Tal como se define a estabilidade para sistemas contínuos na secção 2.2.1, agora vão ser apresentadas as definições de estabilidade para sistemas híbridos [9].

Considerando o sistema híbrido  $H$ .

**Definição 9 (Ponto de equilíbrio):** O estado  $x_e = 0 \in \mathbb{R}^n$  é um ponto de equilíbrio de  $H$  se,

- $f(q, 0) = 0$  para  $\forall q \in Q$
- $R(q, 0) \subset Q \times 0$

Isto quer dizer que são possíveis transições discretas desde que o estado se mantenha no ponto de equilíbrio.

**Definição 10 (Ponto de equilíbrio estável):** Um sistema híbrido  $H$  tem um ponto de equilíbrio  $x_e$  estável se, para  $\varepsilon > 0$  existe  $\delta > 0$  tal que para todos  $(\tau, (q, x))$  que saem de  $(q_0, x_0)$

$$\|x_0\| < \delta \Rightarrow \|x(t)\| < \varepsilon \quad \forall t \in \tau$$

**Definição 11 (Ponto de equilíbrio assintoticamente estável):** Um sistema híbrido  $H$  tem um ponto de equilíbrio  $x_e$  assintoticamente estável se, for estável e existir um  $\delta$  tal que para todos  $(\tau, (q, x))$  começados em  $(q_0, x_0)$

$$\|x_0\| < \delta \Rightarrow \lim_{t \rightarrow \tau_\infty} \|x(t)\| = 0$$

**Teorema 3. (Teorema de estabilidade de Lyapunov para sistemas híbridos)** Considere o sistema híbrido  $H$  e  $x_e = 0$  um ponto de equilíbrio. Assumindo que existe um conjunto aberto  $D \subset Q \times \mathbb{R}^n$  tal que  $(q, 0) \in D$  para  $q \in Q$ . Considere  $V: D \rightarrow \mathbb{R}$  uma função contínua diferenciável em  $x$  tal que para todos  $q \in Q$ :

1.  $V(q, 0) = 0$ ;
2.  $V(q, x) > 0$  para todo o  $x, (q, x) \in D \setminus \{0\}$ , e
3.  $\frac{\partial V(q, x)}{\partial x} f(q, x) \leq 0$  para todo o  $x, (q, x) \in D$

Se para todos  $(\tau, q, x)$  começados a partir de  $(q_0, x_0) \in \text{Init} \cap D$ , e todos  $q' \in Q$ , a sequência  $\{V(q(\tau_i), x(\tau_i)) : q(\tau_i) = q'\}$  é não crescente (ou vazia), então  $x_e = 0$  é um equilíbrio estável de  $H$ .

## 2.5 - Modelo de uma câmara

A projecção de pontos num espaço 3D para um plano 2D (plano da imagem) pode ser efectuada recorrendo a um modelo da câmara.

O modelo da câmara *pinhole* permite fazer esta projecção através de uma relação matemática.

No modelo *pinhole* o plano de imagem pode estar referenciado à frente ou a atrás do centro óptico da câmara. A Figura 2-9 ilustra a geometria envolvida neste modelo para os dois casos. Quando o plano de imagem está à frente do centro óptico, a imagem aparece invertida no plano de imagem.

A projecção de um ponto **P** no plano da imagem é realizada recorrendo à relação matemática dos triângulos similares.

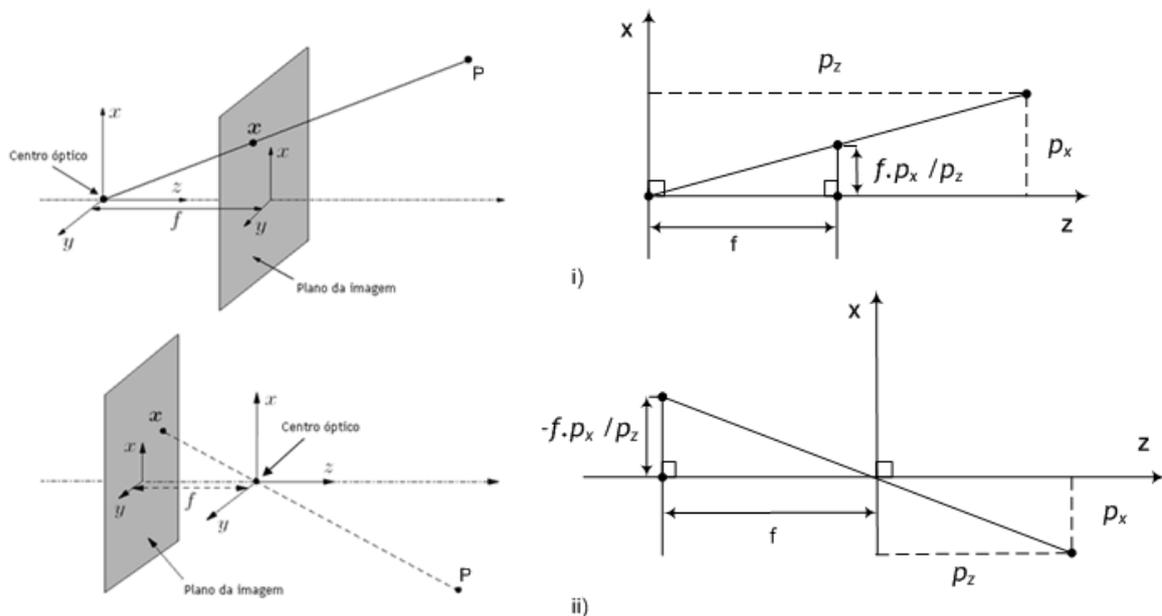


Figura 2-9 - Geometria do modelo *pinhole* com o plano de imagem à frente e atrás do centro óptico, respectivamente i) e ii).

As coordenadas do ponto **X** no plano da imagem são

$$x = f \frac{p_x}{p_z}; y = f \frac{p_y}{p_z} \quad e \quad x = -f \frac{p_x}{p_z}; y = -f \frac{p_y}{p_z},$$

com plano de imagem à frente e a atrás, respectivamente.

O referencial da câmara tem a origem coincidente com o centro óptico e o eixo z está alinhado com o eixo óptico da câmara. O plano da imagem é paralelo ao plano (x,y) a uma distância *f* da origem, onde *f* é a distância de focal da câmara.

A relação entre um ponto  $P$  no referencial inercial e o ponto no sistema de coordenadas da câmara é descrita por um conjunto de transformações nas coordenadas. A rotação e translação de um referencial  $i$  para um referencial  $j$  são representadas por uma matriz de transformação homogênea [10], com a forma

$$H = \begin{bmatrix} R & d \\ 0 & 1 \end{bmatrix}, \quad R \in SO(3), \quad d \in \mathbb{R}^3.$$

A sua inversa é

$$H^{-1} = \begin{bmatrix} R^T & -R^T d \\ 0 & 1 \end{bmatrix}, \quad R \in SO(3), \quad d \in \mathbb{R}^3.$$

Onde  $R \in SO(n)$  representa um grupo ortogonal especial de ordem  $n \times n$ , em que o seu determinante é 1 e a transposta da matriz  $R$  é igual à sua inversa.

## 2.6 - Aproximação de pontos

Em muitos casos temos um conjunto de pontos  $(x_1, y_1) \dots (x_N, y_N)$  e queremos encontrar uma fórmula  $y=f(x)$  que relacione estes dados. Existem várias formas para alcançar este objectivo, de seguida apresentam-se duas das formas: aproximação por um polinómio e interpolação por uma *spline*.

### 2.6.1 - Aproximação por um polinómio

Como se vê na Figura 2-10 a aproximação por um polinómio de grau mais elevado é mais precisa.

Contudo, o aumento da ordem, não significa sempre melhor precisão. Como se pode observar na Figura 2-11, se houver pontos distantes, um polinómio de grau mais elevado pode introduzir oscilações altas entre os pontos.

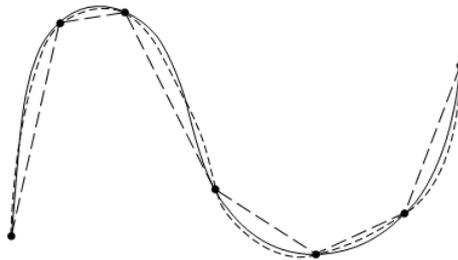


Figura 2-10 - A interpolação por um polinómio de grau elevado é mais precisa (linha) do que um de menor grau (tracejado) [11].

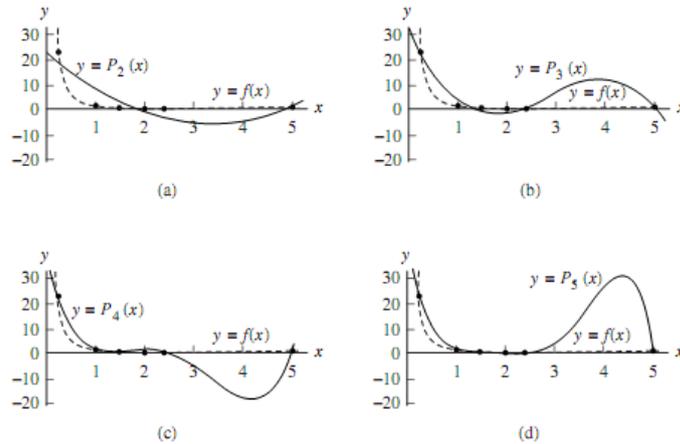


Figura 2-11 - Aproximação polinomial de 2º grau (a), 3º grau (b), 4º grau (c), 5º grau (d) [12].

### 2.6.2 - Interpolação por uma *spline*

Em situações onde a continuidade das derivadas é importante, a utilização da função *spline* é uma das hipóteses. As *splines* cúbicas são polinómios cúbicos entre cada dois pontos, em que a segunda derivada é contínua. Em comparação com os polinómios, as *splines* tendem a ser mais estáveis, com menos possibilidades de apresentarem grandes oscilações entre pontos [11].

### 2.6.3 - Cálculo da curvatura

A curvatura é um indicador do quanto uma curva é “apertada”, em cada ponto da trajectória. Se o valor da curvatura for zero, significa que a trajectória é uma recta. Se o valor for constante, significa um movimento circular.

A curvatura está relacionada com a razão a que o vector tangente unitário muda em relação a  $s$ , representado por

$$\kappa(s) = \left\| \frac{dT}{ds} \right\|, \quad (2.16)$$

onde  $T$  é o vector tangencial unitário e  $s$  é o comprimento do arco.

Na Figura 2-12 nota-se que o sentido da derivada do vector unitário tangente aponta para o interior da curva, na direcção que o vector unitário tangente está a virar.

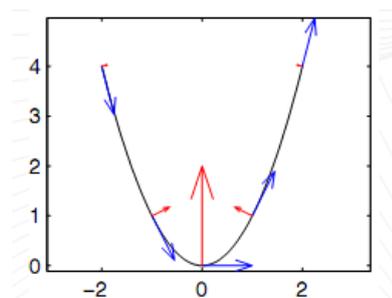


Figura 2-12 - Derivada do vector unitário tangente (vermelho).

O raio de curvatura, em metros, é

$$R(s) = \frac{1}{\kappa(s)} . \quad (2.17)$$

Para um curva, em 2-D, escrita na forma  $y=f(x)$ , a equação da curvatura é

$$\kappa = \frac{\frac{d^2 y}{dx^2}}{\left[ 1 + \left( \frac{dy}{dx} \right)^2 \right]^{3/2}} . \quad (2.18)$$



## Capítulo 3

### Trabalho relacionado

O interesse sobre veículos autónomos tem crescido imenso, encontrando-se vários trabalhos realizados neste tema, quer para veículos autónomos terrestres, quer subaquáticos ou aéreos.

Neste capítulo vão ser expostos alguns dos trabalhos executados nas várias vertentes desta temática, desde vários métodos de controlo aplicados, até ao planeamento de caminhos.

#### 3.1 - Planeamento do caminho

As estruturas naturais apresentam curvaturas variadas e em alguns casos descontínuas, sendo que a maioria dos controladores existentes são apenas indicados para realizar o seguimento de caminhos simples constituídas por *waypoints* interligados por linhas e círculos [13, 14].

Outra abordagem apresentada em [15] é a inserção de uma curva *spline* cúbica (Figura 3-1). O processo consiste na divisão da zona do rio detectada em regiões e depois é inserida uma curva *spline* que interliga os centros de massa dessas regiões.

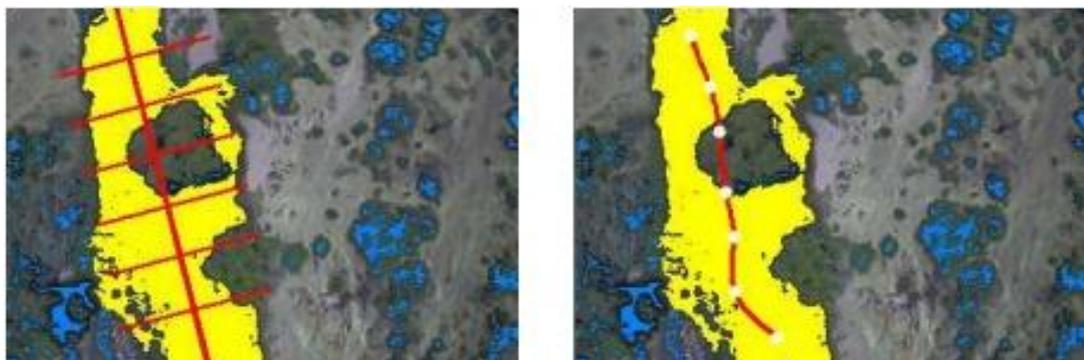


Figura 3-1 - Procedimento de inserção da curva.

O planeamento de caminhos em problemas de seguimento de veículos terrestres móveis é outra vertente encontrada no controlo de UAV's. Em [16] é apresentada uma estratégia que envolve dois modos de funcionamento (*loitter* ou sinusoidal), dependendo da velocidade do veículo terrestre. Se este tem uma velocidade muito menor que o UAV, o modo *loitter* é activado. Caso contrário, é o modo sinusoidal (Figura 3-2).

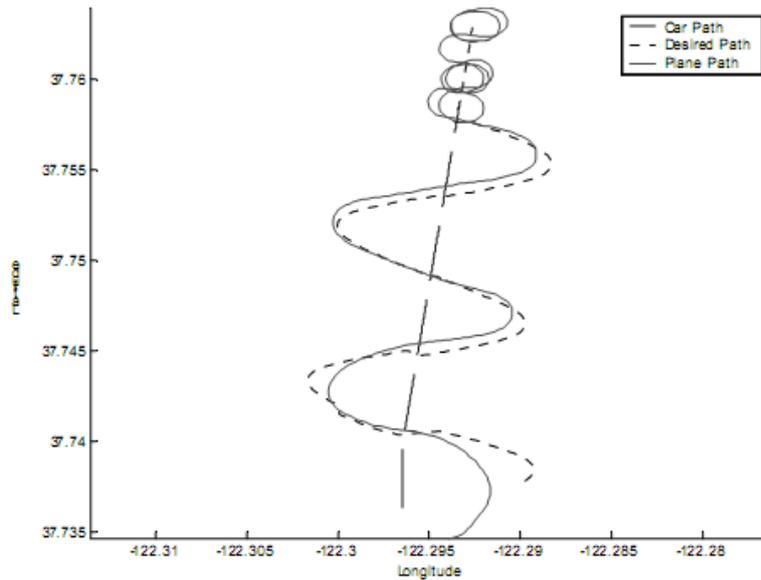


Figura 3-2 - Simulação do seguimento de um veículo terrestre com velocidade variável [16].

### 3.2 - Seguimento de estruturas

Em [15] é apresentado um diagrama de estados do sistema de exploração e seguimento de um rio (Figura 3-3) e as transições envolvidas.

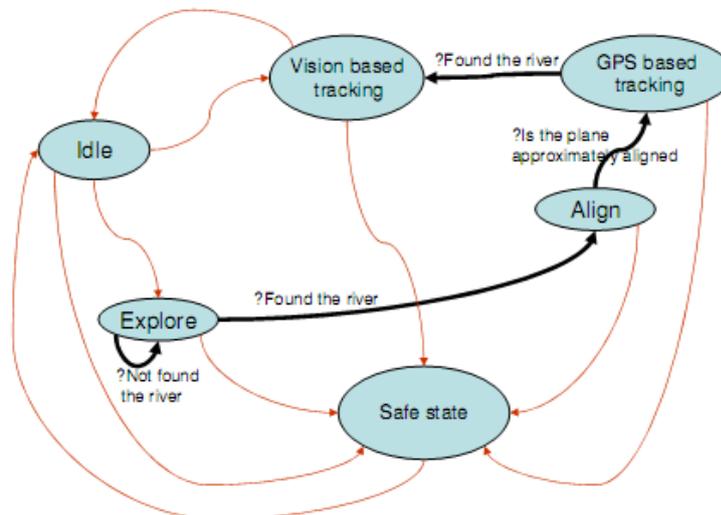


Figura 3-3 - Diagrama de estados do sistema de exploração [15].

O estado *Idle* é o estado inicial, onde o UAV pode estar a seguir uma órbita ou um valor de velocidade de volta fixo. O estado *Explore* é activo pelo operador, com o objectivo do UAV seguir uma órbita em busca do ri. Assim que é encontrado são calculadas as coordenadas GPS do rio, com base nos ângulos *roll*, *pitch* e *yaw*, e posição do UAV. Depois transita para o estado *Align*, onde o UAV se alinha com uma direcção definida pelo operador. Quando alinhado transita para *GPS based tracking* onde segue a curva definida pelas coordenadas GPS, até que encontra o rio e muda para *Vision based tracking* onde o rio é seguido com base na curva calculada pelo algoritmo em tempo real.

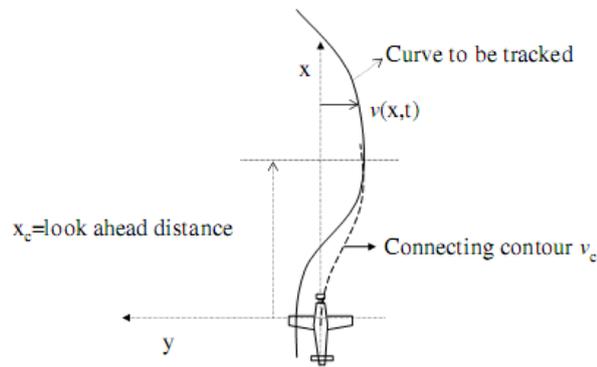


Figura 3-4 - Ilustração da curva a ser seguida e o polinómio de ligação [17].

O problema de controlo consiste no seguimento de uma curva, através do comando da velocidade de volta. Em [17] é apresentada a solução, baseada na definição de um ponto de referência e num polinómio cúbico que liga este ponto à posição do UAV (Figura 3-4), tendo a seguinte forma

$$v_c(x,t) = \alpha(t)x^3 + \beta(t)x^2$$

$$\alpha(t) = -2\frac{v_c(x_c,t)}{x_c^3} + \frac{v'_c(x_c,t)}{x_c^2}.$$

$$\beta(t) = 3\frac{v_c(x_c,t)}{x_c^3} - \frac{v'_c(x_c,t)}{x_c^2}$$

A velocidade de volta comandada é proporcional à curvatura do polinómio na origem. Assim, a lei de controlo é

$$\omega(t) = v \frac{d^2 v_c(0,t)}{dt^2}$$

$$= 2v \left( 3 \frac{v_c(x_c,t)}{x_c^3} - \frac{v'_c(x_c,t)}{x_c^2} \right)$$

Em [18, 19] são desenvolvidas leis de controlo, baseadas em campos vectoriais, para seguimento de rectas e órbitas. Os vectores de campo indicam o curso desejado na direcção do caminho a ser seguido (Figura 3-5).

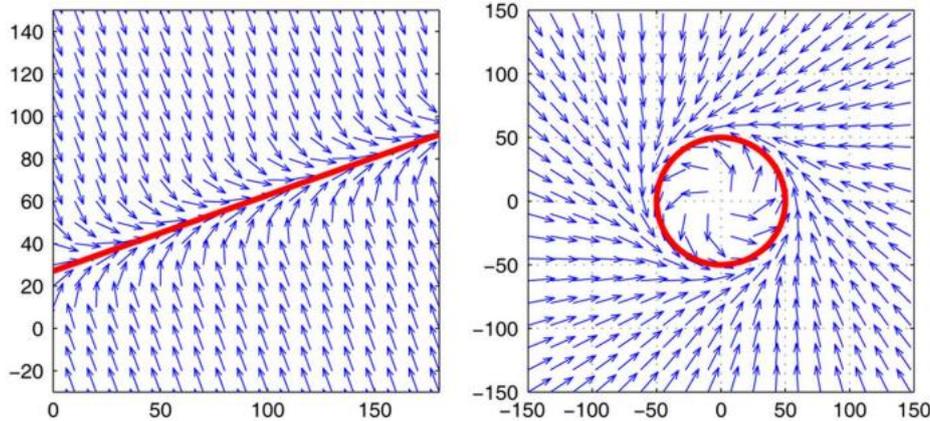


Figura 3-5 - Ilustração dos vectores de campo para uma recta e para uma órbita.

O curso desejado para o seguimento de rectas é definido como:

$$\chi^d(y) = -\chi^\infty \frac{2}{\pi} \tan^{-1}(ky), \quad (3.1)$$

onde  $\chi^\infty$  é o curso de aproximação quando  $y$  (distância lateral do UAV ao caminho) é grande e  $k$  é uma constante positiva que influencia a transição de  $\chi^\infty$  para zero.

Para seguimento de órbitas, a orientação desejada é definida em [19] como

$$\begin{aligned} \chi^d &= \gamma - \pi + \sin^{-1}\left(\frac{r}{d}\right) && \text{quando } d > 2r \\ \chi^d &= \gamma - \frac{\pi}{2} - \frac{\pi}{3} \left(\frac{d-r}{r}\right)^k && \text{quando } d \leq 2r \end{aligned}$$

onde  $d$  é a distância do UAV ao centro da órbita,  $r$  é o raio da órbita e  $\gamma$  é a orientação da linha que liga o centro da órbita à posição do UAV.

Em [20] é apresentada uma solução baseada em *sliding mode control*, para um robô móvel efectuar o seguimento de um caminho. A lei de controlo usa duas *sliding surfaces* com o objectivo de extinguir os três erros (orientação, posição  $x$  e  $y$ ).

Em [21] é exposta uma arquitectura de controlo hierárquica para patrulhamento de fronteiras. A arquitectura é composta pela camada do piloto automático, pela camada que coordena as manobras (geração de caminhos e controladores de seguimento) e pelas camadas de supervisão (veículo e equipa). A estratégia de controlo é baseada em *sliding mode control*. Considera-se o seguinte modelo em coordenadas polares (Figura 3-6),

$$\begin{aligned}\dot{r} &= u_1 \cos(\psi - \theta) \\ \dot{\theta} &= \frac{u_1}{r} \sin(\psi - \theta), \\ \dot{\psi} &= u_2\end{aligned}$$

onde  $u_1$  e  $u_2$  são a velocidade linear e velocidade de rotação, respectivamente. Assume-se que  $u_1$  é constante e  $u_2$  é a variável de controle.

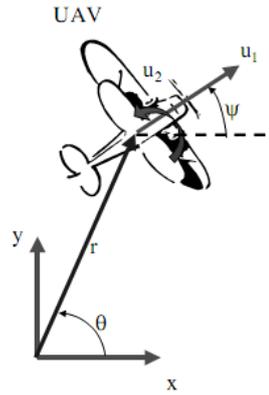


Figura 3-6 - Modelo cinemático do UAV [21].

Desenvolveu-se um controlador com múltiplas *sliding surfaces*, considerando a primeira *sliding surface*

$$S = R - R_{des},$$

pretendendo-se a seguinte dinâmica

$$\dot{S} = -\lambda S.$$

Substituindo a derivada, obtêm-se

$$u_1 \cos(\psi - \theta) - \dot{R}_{des} = -\lambda(R - R_{des})$$

Definindo  $\alpha_{des} = \psi_{des} - \theta$ , obtêm-se

$$\alpha_{des} = \arccos\left(\frac{-\lambda_r(R - R_{des}) + \dot{R}_{des}}{u_1}\right).$$

A segunda *sliding surface* definida é

$$S_2 = \alpha - \alpha_{des},$$

com a mesma dinâmica da primeira, obtêm-se

$$u_2 - \frac{u_1}{r} \sin(\psi - \theta) - \dot{\alpha}_{des} = -\lambda_2(\alpha - \alpha_{des}).$$

Nesta equação aparece a variável de controle, assim a lei de controle é

$$u_2 = -\lambda_2(\alpha - \alpha_{des}) + \frac{u_1}{r} \sin(\psi - \theta) - \dot{\alpha}_{des}.$$

Uma lei de controlo baseada no comportamento ‘Good Helmsman’ é desenvolvida em [22], com o objectivo de converter o *relative course error*, dado por (3.2), e a distância ao caminho,  $y_s$ , para zero (Figura 3-7), simultaneamente. Isto é obtido com o acoplamento do curso comandado e a distância ao caminho, isto é  $\tilde{\chi}_c(y_s)$ .

$$\tilde{\chi} \triangleq \chi - \chi_s \quad (3.2)$$

Onde  $\chi$  é o curso no momento e  $\chi_s$  é a direcção do caminho.

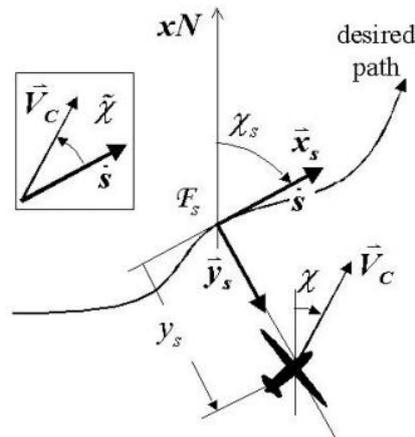


Figura 3-7 - Referenciais.

O comportamento ‘Good Helmsman’ é expresso colocando o  $\chi_c$  em função de  $\chi_s$  e  $y_s$ , obtendo

$$\begin{aligned} \tilde{\chi}_c &\triangleq \chi_c - \chi_s = \sigma(y_s) \\ \chi_c(y_s, \chi_s) &= \sigma(y_s) + \chi_s \end{aligned}$$

onde  $\sigma(y_s)$  representa uma função sigmoide (Figura 3-8) que tem a função de estabelecer  $\tilde{\chi}_c \approx 0$  quando está perto do caminho e  $\tilde{\chi}_c = \pm \tilde{\chi}_{icpt}$  quando está longe.

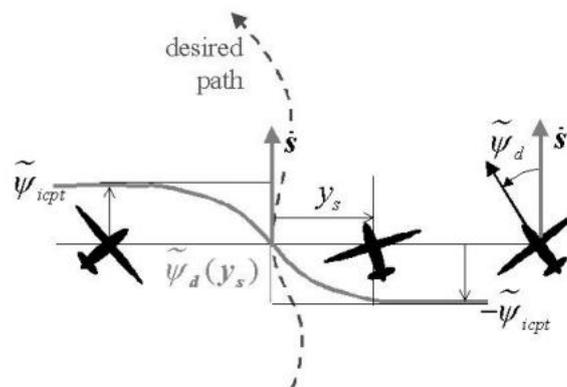


Figura 3-8 - Função sigmoide que relaciona o curso desejado com a distância ao caminho.

Outras técnicas não lineares, apresentadas em [14, 23], seguem uma lógica semelhante a um controlador PD aplicado no seguimento de linhas rectas, mas com a inclusão de um elemento de controlo de antecipação. Este elemento consiste no uso de um ponto de referência que se move ao longo do caminho desejado, funcionando como um pseudo-alvo (Figura 3-9). O método usa a velocidade inercial na computação da aceleração lateral e tem a capacidade de se adaptar à presença de vento.

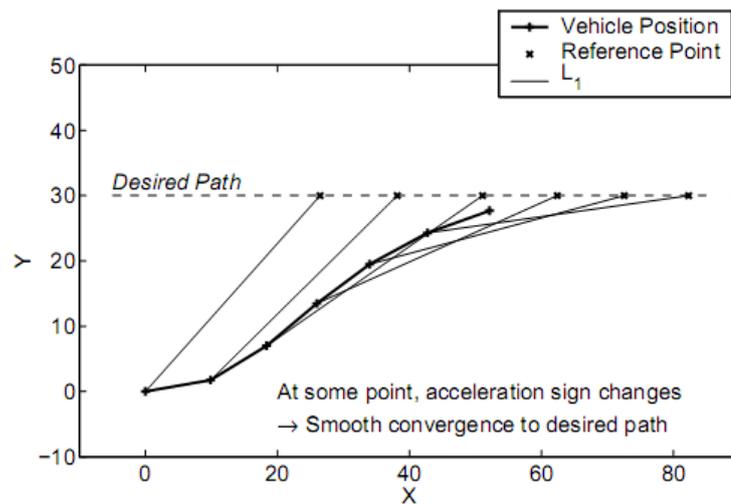


Figura 3-9 - Simulação passo a passo do algoritmo com ponto de referência [23].

Outros tipos de controlo praticados são o  $H_{inf}$  e PI [24], controlo adaptativo [25] e funções de controlo de Lyapunov [26].



# Capítulo 4

## Formulação dos problemas

Após uma análise do tema deste trabalho foram identificados vários problemas para estudo e resolução.

Neste capítulo são formulados os problemas sobre o modelo simplificado do UAV utilizado neste trabalho; a aproximação/interpolação recursiva de pontos; seguimento de uma curva viável com velocidade constante e ajustável; e o seguimento de uma curva não viável.

### 4.1 - Modelo 6-dof

O movimento de um veículo é descrito em termos de forças, momentos, velocidades lineares e angulares e ângulos Euler (Figura 4-1).

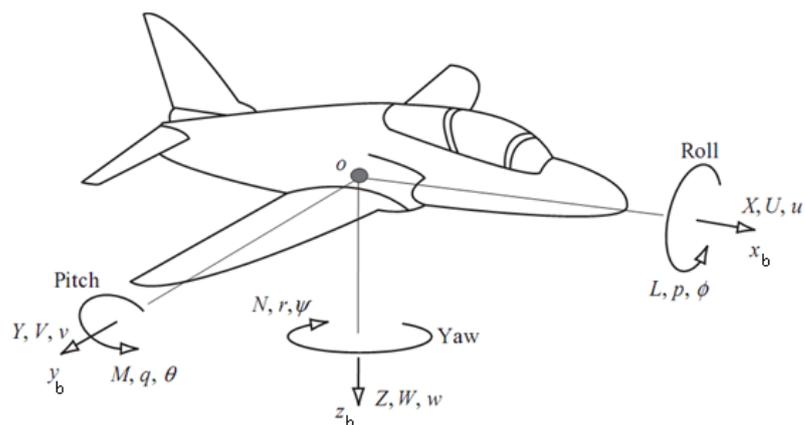


Figura 4-1 - Notação das variáveis das equações de movimento [27].

Tabela 1 - Sumário das variáveis de movimento envolvidas.

Nome	Variável
Referencial inercial	x y z
Referencial do UAV	X <sub>b</sub> Y <sub>b</sub> Z <sub>b</sub>
Força	X Y Z
Momento	L M N
Velocidade linear	U V W
Velocidade angular	p q r
Atitude	$\phi$ $\theta$ $\psi$

Os ângulos de Euler que definem a atitude do UAV são definidos pela rotação de um referencial fixo (inercial), segundo a regra da mão direita. O sentido e a ordem das rotações são importantes, porque os ângulos não obedecem à regra da comutatividade. Assim, existem 12 sequências possíveis.

A ordem das rotações seguida foi: *roll*, *pitch* e *yaw*. Assim, obtêm-se a matriz de rotação (4.1), que transforma as coordenadas do referencial do UAV para o referencial inercial.

$$T_{\psi} = \begin{bmatrix} c_{\psi} & -s_{\psi} & 0 \\ s_{\psi} & c_{\psi} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad T_{\theta} = \begin{bmatrix} c_{\theta} & 0 & s_{\theta} \\ 0 & 1 & 0 \\ -s_{\theta} & 0 & c_{\theta} \end{bmatrix} \quad T_{\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{\phi} & -s_{\phi} \\ 0 & s_{\phi} & c_{\phi} \end{bmatrix}$$

$$T_{\psi}T_{\theta}T_{\phi} = \begin{bmatrix} c_{\psi}c_{\theta} & -s_{\psi}c_{\phi} + c_{\psi}s_{\theta}s_{\phi} & s_{\psi}s_{\phi} + c_{\psi}c_{\phi}s_{\theta} \\ s_{\psi}c_{\theta} & c_{\psi}c_{\phi} + s_{\psi}s_{\theta}s_{\phi} & -c_{\psi}s_{\phi} + s_{\psi}s_{\theta}c_{\phi} \\ -s_{\theta} & c_{\theta}s_{\phi} & c_{\theta}c_{\phi} \end{bmatrix}, \quad (4.1)$$

onde  $c = \cos$  e  $s = \sin$ .

As equações da dinâmica descrevem os efeitos aerodinâmicos e as forças de propulsão que actuam no veículo aéreo, relativamente ao referencial do veículo. Enquanto as equações cinemáticas fornecem informação da posição e orientação do veículo no referencial inercial.

As equações da cinemática e da dinâmica do modelo de 6-dof apresentam-se de seguida [27]:

#### Equações cinemáticas:

A velocidade do UAV no referencial inercial pode ser calculada pela seguinte transformação da velocidade no referencial do veículo:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = T_\psi T_\theta T_\phi \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

A relação entre as derivadas dos ângulos Euler e as velocidades angulares é estabelecida da seguinte forma:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -s_\phi \\ 0 & c_\phi & s_\phi c_\theta \\ 0 & -s_\phi & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (4.2)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{pmatrix} 1 \\ c_\theta \end{pmatrix} \begin{bmatrix} c_\theta & s_\phi s_\theta & c_\phi s_\theta \\ 0 & c_\phi c_\theta & -s_\phi c_\theta \\ 0 & s_\phi & c_\phi \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Quando os ângulos  $\phi, \theta, \psi$  têm variações de amplitudes pequenas, as equações (4.2) podem ser aproximadas por

$$\begin{aligned} p &= \dot{\phi} \\ q &= \dot{\theta} \\ r &= \dot{\psi} \end{aligned}$$

#### Equações generalizadas das forças:

As equações (4.3) representam as forças, obtidas pela aplicação da segunda lei de Newton, num corpo rígido.

$$\begin{aligned} m(\dot{U} - rV + qW) &= X \\ m(\dot{V} - pW + rU) &= Y \\ m(\dot{W} - qU + pV) &= Z, \end{aligned} \quad (4.3)$$

onde  $m$  é a massa do corpo rígido.

#### Equações generalizadas dos momentos:

$$\begin{aligned} I_x \dot{p} - (I_y - I_z)qr - I_{xz}(pq + \dot{r}) &= L \\ I_y \dot{q} + (I_x - I_z)qr + I_{xz}(p^2 - r^2) &= M \\ I_z \dot{r} - (I_x - I_y)qr + I_{xz}(qr - \dot{p}) &= N \end{aligned} \quad (4.4)$$

As equações (4.4) descrevem o momento de *roll*, *pitch* e *yaw*, respectivamente.

## 4.2 - Modelo simplificado do UAV

O modelo 6-dof é bastante complexo, por isso adoptou-se um modelo simplificado. Apesar de ser um modelo simples, cumpre os requisitos necessários para os tipos de problemas que se colocam. É de referir que, por exemplo, para problemas como a evasão de obstáculos/ameaças seria necessário o uso de um modelo 6-dof (seis graus de liberdade).

Listam-se algumas das simplificações:

- Volta coordenada (ângulo *sideslip* igual a zero);
- Altitude constante;
- Inércias não modeladas;
- Controlo apenas da velocidade de volta e da velocidade linear.

Será um bom teste para a robustez dos controladores a transição para um modelo 6-dof. Como os controladores desenvolvidos se baseiam na teoria de *sliding mode control*, a transição pode ser entendida como uma perturbação a que os controladores são sujeitos.

As equações cinemáticas do modelo adoptado, no plano x-y, são representadas por

$$\begin{aligned}\dot{x} &= V \cos(\psi) + W_x = V_g \cos(\chi) \\ \dot{y} &= V \sin(\psi) + W_y = V_g \sin(\chi) , \\ \dot{\psi} &= \omega\end{aligned}\quad (4.5)$$

onde  $V_g$  representa a velocidade relativa ao solo em m/s,  $\omega$  é a velocidade de volta em rad/sec,  $\chi$  é curso seguido pelo veículo, em rad.

**Definição 12 (Raio de curvatura do UAV):** Resolvendo as duas equações que definem as forças horizontais e verticais, dadas por (4.6), eliminando a componente de sustentação ( $L$ ), chega-se à fórmula (4.7) que define o raio de curvatura.

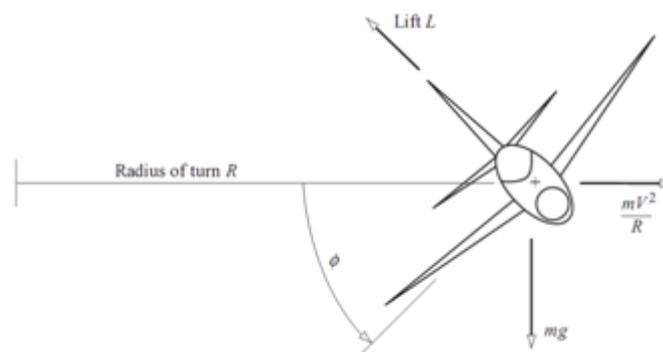


Figura 4-2 - Representação das forças existentes numa volta coordenada.

Assumindo que as voltas são coordenadas (Figura 4-2), então as equações das forças envolvidas são

$$\begin{cases} mg = L \cos \phi \\ \frac{mV^2}{R} = L \sin \phi \end{cases} \quad (4.6)$$

Resolvendo o sistema, resulta

$$R_{UAV} = \frac{V^2}{g \tan(\phi)} \quad (4.7)$$

O tempo para completar uma volta é dado por

$$t = \frac{2\pi R}{V} = \frac{2\pi V}{g \tan(\phi)} \quad (4.8)$$

Assim a velocidade de volta está relacionada com o ângulo *roll* por

$$\omega = \frac{2\pi}{t} = \frac{g \tan(\phi)}{V} \quad (4.9)$$

Neste modelo existem três estados  $(x, y, \psi)$  e duas variáveis de controlo  $(V, \omega)$ , sendo estas limitadas.

A velocidade  $V$  tem um valor mínimo e máximo, que depende das características do UAV. Abaixo do valor mínimo o UAV não gera sustentação suficiente para se manter no ar de forma controlável.

A velocidade de volta, dada pela fórmula (4.9), está limitada, para que o ângulo de pranchamento  $\phi$  não ultrapasse os 25 graus.

**Definição 13 (Curva viável):** Designa-se por curva viável, uma curva que possui um raio de curvatura, dado por (2.17), superior ao do veículo UAV, ao longo de todo o seu percurso. Considerando as restrições impostas pela velocidade linear e velocidade de volta.

Isto é, o raio de curvatura mínimo exigido ao UAV durante todo o percurso deve ser superior ao raio de curvatura que o UAV pode efectuar, calculado com base numa velocidade e num ângulo de pranchamento máximo.

A definição pode ser representada matematicamente por,

$$R_{curva} \geq R_{UAV} \quad \forall t$$

$$\Leftrightarrow \frac{\left[1 + \left(\frac{dy}{dx}\right)^2\right]^{3/2}}{\frac{d^2y}{dx^2}} \geq \frac{V^2}{g \cdot \tan(\phi_{\max})} \quad (4.10)$$

**Definição 14 (Velocidade mínima):** Designa-se por velocidade mínima aquela à qual a sustentação criada para manter o voo nivelado (sem perder altura) está no limiar.

**Definição 15 (Curva não viável):** Designa-se por curva não viável, uma curva que possui um raio de curvatura inferior ao do UAV, em alguns pontos do seu percurso. Traduzindo-se matematicamente por,

$$R_{curva} < R_{UAV}$$

### 4.3 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo

**Problema 1.** Através das imagens de vídeo captadas são obtidos pontos que localizam a estrutura a ser seguida. Perante isto, é necessário gerar um caminho de referência, realizando uma aproximação recursiva, para o UAV seguir.

A aproximação recursiva deve “ajustar” a curva para que esta seja viável (ver Definição 11). Sendo um método iterativo (necessário efectuar várias aproximações até estar de acordo com a Definição 11) e necessitar do conhecimento *a priori* de pontos distanciados do UAV para proceder ao ajustamento da curva antes de lá chegar, a sua utilização considera-se apenas em casos em que o UAV não está a seguir esses pontos. Por exemplo, está a fazer órbitas enquanto adquire pontos e corre o algoritmo.

Este método dá prioridade à construção de uma curva viável para o UAV, em vez de uma curva que garanta a passagem pelos pontos.

**Problema 2.** Através de pontos obtidos das imagens de vídeo é necessário criar um caminho para o UAV seguir. A criação e evolução do caminho tem que ser executada enquanto o UAV segue-o.

Neste problema a prioridade é ter um algoritmo rápido que realize a interpolação dos pontos de localização da estrutura que vão chegando do processamento das imagens, em vez da geração de uma curva viável.

#### 4.4 - Seguimento de uma curva viável com velocidade constante

Após realizada a aproximação dos pontos e encontrada a curva de referência que o UAV deve seguir, coloca-se o problema de o UAV executar o seguimento desta.

**Problema 3.** O UAV tem que efectuar o seguimento de uma curva viável (ver definição 11). É assumido que o UAV possui uma velocidade linear constante e é controlado através da velocidade de volta ( $\omega$ ).

A Figura 4-3 apresenta a informação que o algoritmo de controlo recebe e envia.

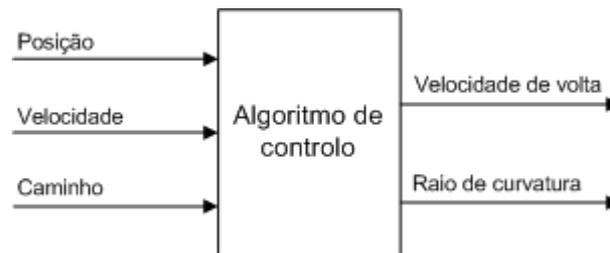


Figura 4-3 - Estrutura da informação do algoritmo de controlo do seguimento da estrutura.

#### 4.5 - Seguimento de uma curva viável com ajustamento da velocidade

Como já foi referido anteriormente o UAV tem uma velocidade mínima, e analisando a fórmula (4.7) verifica-se que o raio de curvatura do UAV varia linearmente com o quadrado da velocidade. Ao considerar, por exemplo, uma velocidade mínima (ver definição 12) de 15m/s e uma velocidade de cruzeiro de 20m/s, obtêm-se raios de curvatura de respectivamente,

$$R_{min} = \frac{15^2}{9.8 \cdot \tan(25 \cdot \pi / 180)} \approx 49m$$

e

$$R_{min} = \frac{20^2}{9.8 \cdot \tan(25 \cdot \pi / 180)} \approx 88m$$

Ou seja, a redução da velocidade de 5 m/s, reduz o raio de curvatura em quase 40 m.

Verifica-se assim que através da variação da velocidade é possível realizar o seguimento de curvas mais “apertadas”.

**Problema 4.** Realizar o seguimento de uma curva viável, considerando a possibilidade do ajustamento da velocidade. Assim, o controlo é efectuado através das duas variáveis ( $V, \omega$ ). A Figura 4-4 apresenta os parâmetros de entrada e saída do algoritmo de controlo.

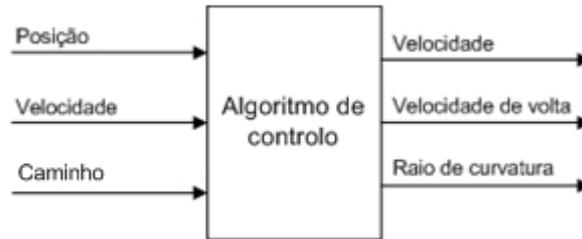


Figura 4-4 - Estrutura da informação do algoritmo de controlo do seguimento da estrutura, com ajustamento da velocidade.

Neste caso, a velocidade mínima estabelece o raio de curvatura mínimo da curva, de forma a garantir que continua a ser uma curva viável.

Pretende-se, portanto, que o UAV percorra o caminho a uma velocidade de cruzeiro, diminuindo a velocidade quando encontrar uma curva que necessita de uma velocidade menor para ser executada.

#### 4.6 - Seguimento de uma curva não viável

**Definição 16 (Área pesquisada):** Área pesquisada é toda aquela percorrida pelo campo de visão da câmara.

**Problema 5.** Quando é exigido ao UAV o seguimento de uma curva não viável, vai existir uma área do caminho que não vai ser pesquisada. Como o objectivo é garantir que a área pesquisada inclua todo o caminho, torna-se necessário desenvolver um conjunto de manobras para manter o caminho dentro do campo de visão da câmara. Para o efeito é assumido que a altitude e velocidade são constantes.

# Capítulo 5

## Abordagem

Neste capítulo apresentam-se as abordagens seguidas na resolução dos problemas descritos no capítulo 4.

### 5.1 - Projecção do campo de visão da câmara

A projecção do campo de visão da câmara tem em conta os ângulos *pitch*, *roll* e *yaw* do UAV, os ângulos *pan* e *tilt*, e a posição da câmara [28].

As coordenadas que definem a projecção no referencial inercial são obtidas através de duas transformações homogéneas (Tabela 1).

Tabela 2 - Transformações homogéneas.

Transformação	Descrição
$T_{AC}^{Inercial}$	Transformação das coordenadas no referencial do UAV para o referencial Inercial
$T_{Cam}^{AC}$	Transformação das coordenadas no referencial da Câmara para o referencial do UAV

#### Transformação do referencial da Câmara para o referencial da UAV

Esta transformação tem em conta os ângulos *pan* e *tilt* da câmara. Considera-se que o centro óptico e de rotação da câmara situa-se na origem do referencial da câmara.

A transformação do referencial da câmara para o referencial do UAV é dada por:

$$T_{Cam}^{AC} = \begin{bmatrix} R_{Cam}^{AC} & d_{Cam}^{AC} \\ 0 & 1 \end{bmatrix}$$

$$R_{Cam}^{AC} = \begin{bmatrix} C_{Pan}C_{Tilt} & -S_{Pan} & C_{Pan}S_{Tilt} \\ S_{Pan}C_{Tilt} & C_{Pan} & S_{Pan}S_{Tilt} \\ -S_{Tilt} & 0 & C_{Tilt} \end{bmatrix},$$

onde  $c = \cos$  e  $s = \sin$  e  $d_{Cam}^{AC}$  é a posição da câmara no referencial do UAV.

### Transformação do referencial UAV para o referencial Inercial

Esta transformação tem em conta os ângulos Euler do UAV (*pitch*, *roll*, *yaw*) e é definida por:

$$T_{AC}^{Inercial} = \begin{bmatrix} R_{AC}^{Inercial} & d_{AC}^{Inercial} \\ 0 & 1 \end{bmatrix}$$

$$R_{AC}^{Inercial} = \begin{bmatrix} C_{Pitch}C_{Yaw} & S_{Roll}S_{Pitch}C_{Yaw} - C_{Roll}S_{Yaw} & C_{Roll}S_{Pitch}C_{Yaw} + S_{Roll}S_{Yaw} \\ C_{Pitch}S_{Yaw} & S_{Roll}S_{Pitch}S_{Yaw} + C_{Roll}C_{Yaw} & C_{Roll}S_{Pitch}S_{Yaw} - S_{Roll}C_{Yaw} \\ -S_{Pitch} & S_{Roll}C_{Pitch} & C_{Roll}C_{Pitch} \end{bmatrix},$$

onde  $d_{AC}^{Inercial}$  é a posição do UAV no referencial inercial.

As coordenadas da projecção são calculadas da seguinte forma (Figura 5-1):

i) Cálculo de  $\lambda$  (distância do centro óptico ao ponto em causa) através de,

$$\lambda = \text{Alt\_AC} - z\_Cam\_ACRef / Pxl\_EarthRef(3).$$

ii) Cálculo das coordenadas do ponto (2), no referencial inercial, através de,

$$x\_Cnr = Pxl\_EarthRef(1) \cdot \lambda$$

$$y\_Cnr = Pxl\_EarthRef(2) \cdot \lambda$$

Alt\\_AC = altitude do UAV.

z\\_Cam\\_ACRef = coordenada z da posição da câmara no referencial do UAV, relativamente ao referencial inercial.

Pxl\\_EarthRef = vector de coordenadas [x y z] do ponto de interesse do plano de imagem (1), no referencial inercial.

x\\_Cnr = coordenada x do canto, no referencial inercial.

y\\_Cnr = coordenada y do canto, no referencial inercial.

O cálculo dos outros três cantos é efectuado de forma semelhante.

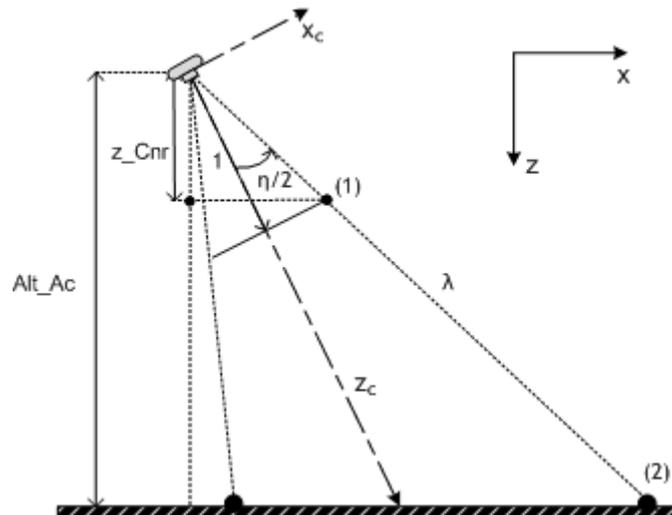


Figura 5-1 - Geometria do cálculo das coordenadas da projecção do campo de visão da câmara.

## 5.2 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo

A aproximação por um polinómio ou a interpolação por uma *spline* resolve os Problemas 1 e 2, descritos no Capítulo 4.

As funções de aproximação por polinómios e interpolação por uma *spline* cúbica foram retiradas do livro *Numerical Methods Using Matlab* [12]. Optou-se por implementar as funções, em vez das proporcionadas pelo Matlab<sup>TM</sup>, porque uma próxima linha de desenvolvimento será a conversão do código para C++, para testar os algoritmos na prática.

### 5.2.1 - Aproximação por um polinómio

A função *lspoly* aproxima os pontos por um polinómio, pelo método dos mínimos quadrados.

O grau do polinómio é transmitido como parâmetro da função, junto com as coordenadas  $x$  e  $y$ .

Na Figura 5-2 ilustra-se a aproximação de um conjunto de pontos por polinómios de vários graus. Verifica-se que quanto maior o grau do polinómio, melhor é a aproximação. Mas tem a desvantagem de introduzir oscilações maiores, como é o caso entre o intervalo [1800 2000]. Este efeito é maior, quanto maior for o grau do polinómio.

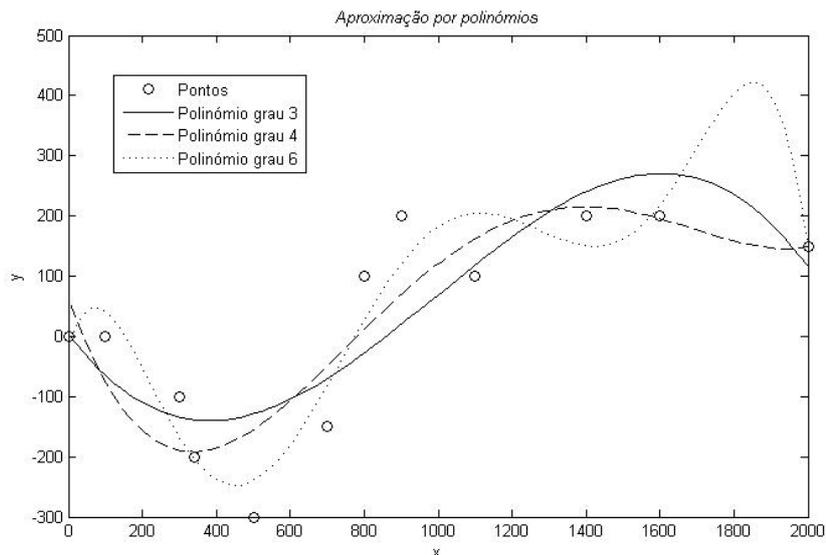


Figura 5-2 - Aproximação por polinômio de grau 3, 4 e 6.

O método iterativo para obter um caminho viável consiste em detectar os pontos que não obedecem à definição de caminho viável (ver Definição 13) e acrescentar um ponto na direcção do vector normal nesse local (Figura 5-3).

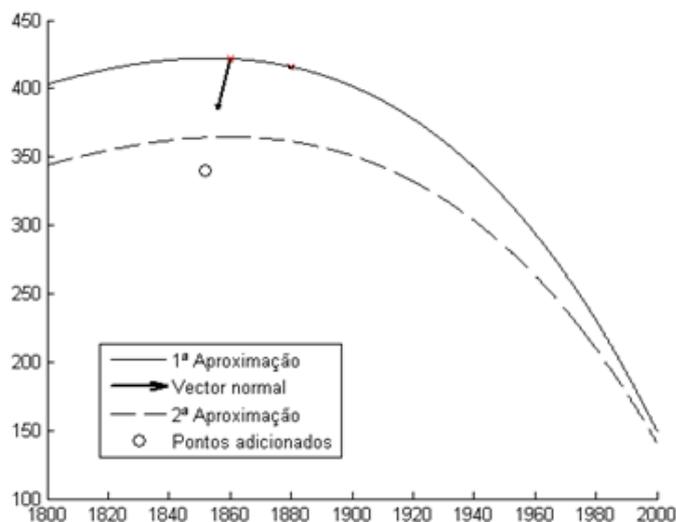


Figura 5-3 - Método de alteração do caminho, com aproximações por polinômios.

Na próxima aproximação, os pontos acrescentados vão fazer diminuir a curvatura nessa zona, porque são tidos em conta no método dos mínimos quadrados que tenta diminuir o erro.

A Figura 5-4 ilustra o método iterativo para criar uma curva viável. Neste exemplo, a restrição é para pontos com um raio de curvatura menor que 80m (simbolizados pela cruz vermelha), caso passe este valor são adicionados pontos (pontos vermelhos), que faz com que a próxima aproximação tenha um raio de curvatura maior nessa zona.

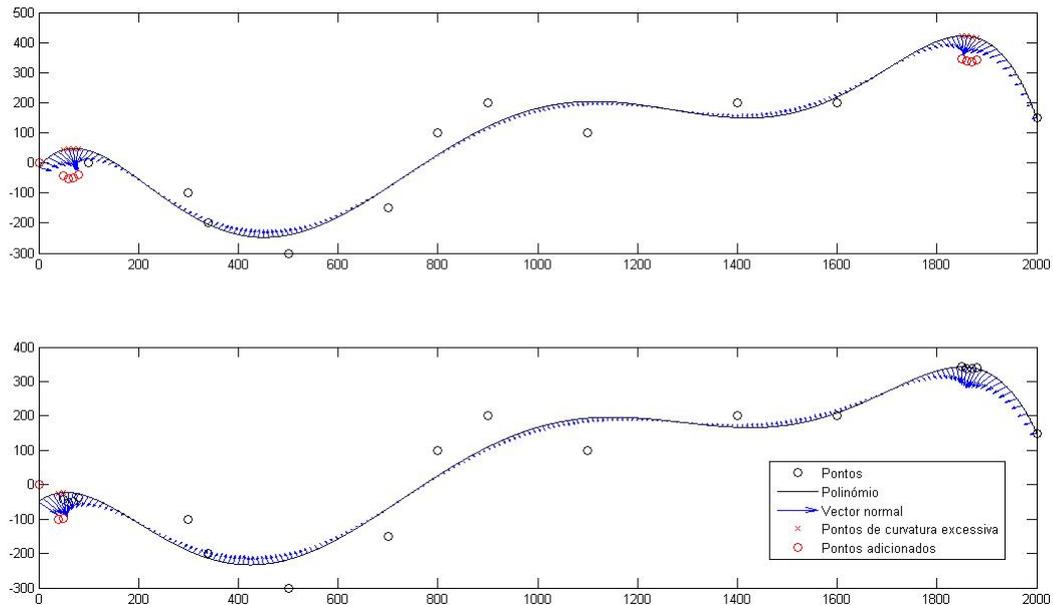


Figura 5-4 - Aproximação iterativa por polinómios.

Este método pode aplicar-se a situações em que já se conhecem os pontos e deseja-se encontrar um caminho que melhor aproxime os pontos e respeite a restrição da curvatura.

### 5.2.2 - Interpolação por uma *spline* cúbica

A utilização de *splines* aplica-se em dois casos:

- 1) Quando já se conhecem os pontos e pretende-se o mesmo objectivo da aproximação por polinómios, ou seja, uma curva viável.
- 2) O outro caso aplica-se quando os pontos vão sendo adicionados à medida que aparecem no campo de visão da câmara.

#### Caso 1

A Figura 5-5 apresenta o resultado da interpolação por uma *spline* cúbica. Verifica-se que existem três zonas onde a curvatura é excessiva.

A forma para realizar a alteração do caminho, de maneira a cumprir a restrição da curvatura, consiste em detectar os pontos que não obedecem à restrição e movê-los no sentido do vector normal (o que aponta para dentro da curva).

Ao mover o ponto, se este ultrapassar a componente do eixo dos  $xx$  ou  $yy$ , do ponto anterior ou seguinte, é apagado.

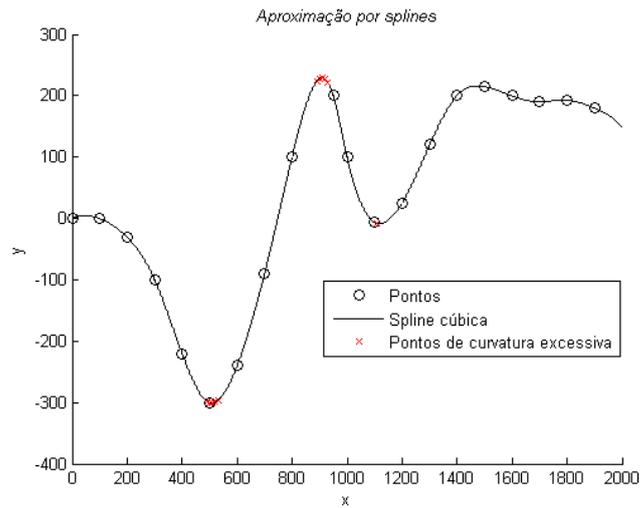


Figura 5-5 - Aproximação por *spline* cúbica.

A sequência da alteração dos pontos até obter-se um caminho viável é ilustrada na Figura 5-6.

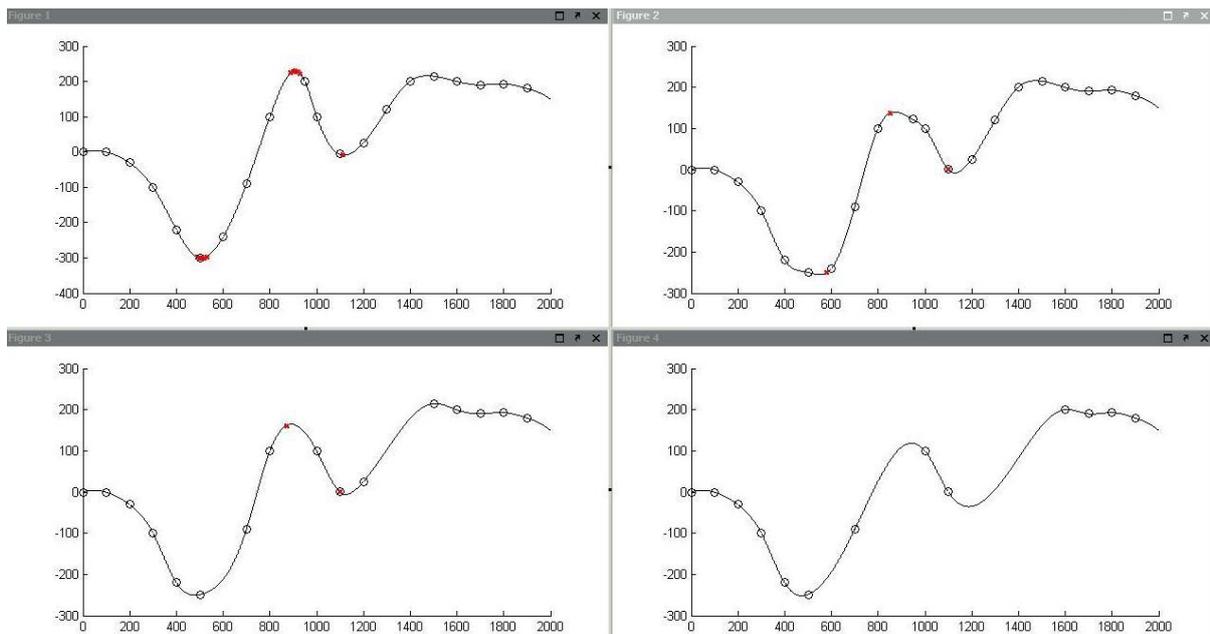


Figura 5-6 - Fases da construção da curva viável, por uma *spline* cúbica, tendo em conta a restrição de um raio de curvatura mínimo de 80m.

## Caso 2

Neste caso, a interpolação é executada assim que entra um novo ponto na “memória” (vector com dimensão fixa e que simula os últimos pontos localizados). A Figura 5-7 ilustra a sequência de interpolações executadas sempre que é adicionado um ponto novo.

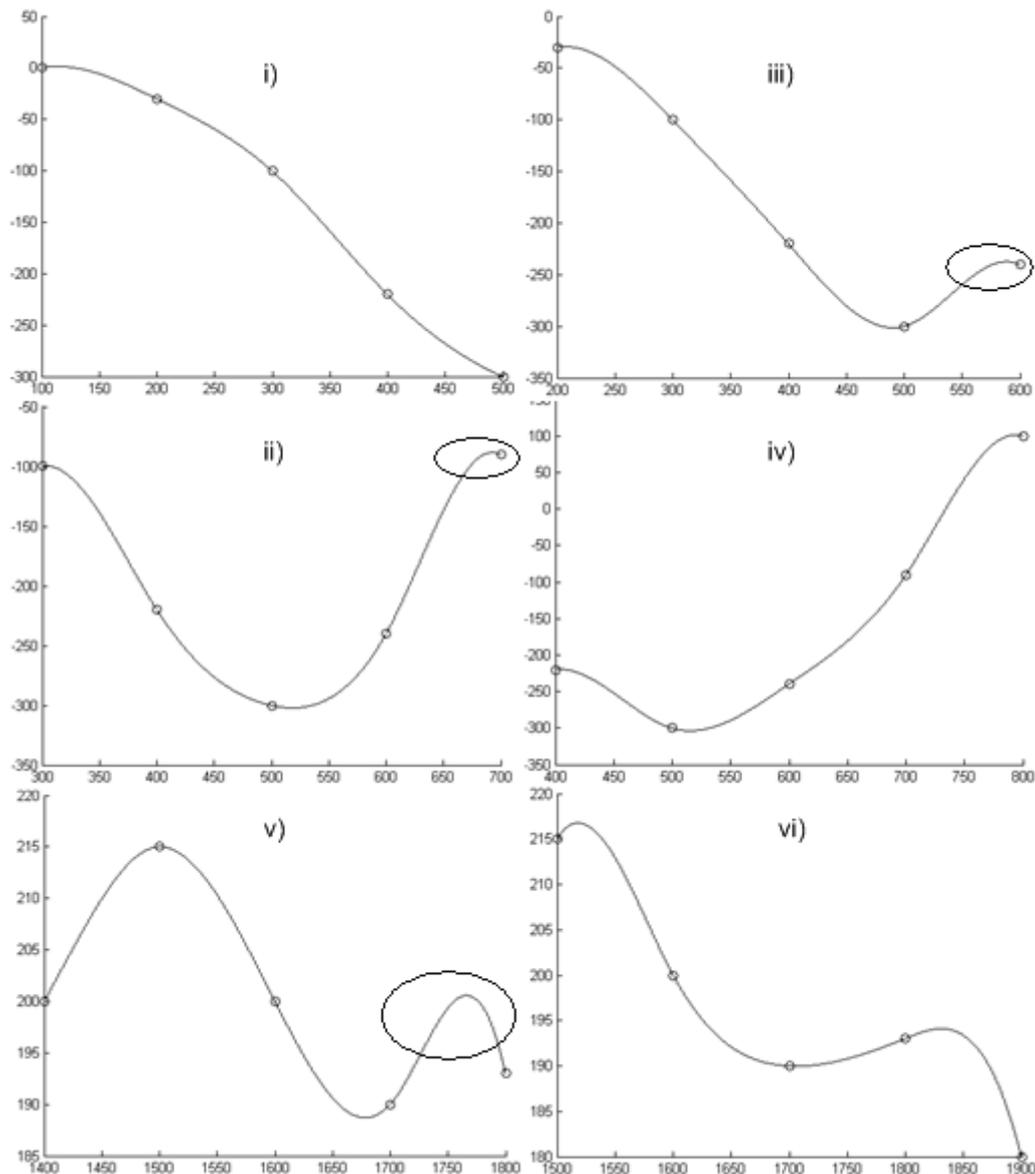


Figura 5-7 - Sequência da interpolação de pontos.

Na Figura 5-7 verifica-se que o último segmento a ser adicionado tem uma curvatura grande, nas seqüências ii), iii) e v). Isto deve-se à condição da primeira derivada no último ponto ter um valor definido, passado como parâmetro à função *csfit*. Esta função tem quatro parâmetros de entrada (coordenadas x e y dos pontos e os valores da 1ª derivada no primeiro e último ponto) e efectua a interpolação cúbica de um conjunto de pontos.

Este excesso de curvatura vai fazer com que o UAV “entenda” que está perante uma curva apertada e entra no modo *search* (modo do controlador híbrido explicado na secção 5.5), o que nem sempre é correcto. Por exemplo, a seqüência iii) apresenta excesso de curvatura na zona do ponto de coordenadas (700,-100) e verifica-se que quando adicionado o próximo ponto (seqüência iv) a zona apresenta uma curvatura viável. O mesmo acontece para a seqüência v).

Para evitar esta comutação desnecessária para o modo *search*, impõem-se uma condição relativamente à distância entre os pontos que são obtidos do processamento da imagem.

A distância entre pontos tem que garantir que existem sempre pelo menos três pontos à frente do UAV.

A distância entre pontos é calculada pela seguinte fórmula,

$$DistPts < DistAhead / 2,$$

onde *DistPts* é a distância entre pontos obtidos do processamento da imagem de vídeo e *DistAhead* é a distância entre o UAV e o ponto mais à frente no campo de visão (Figura 5-8).

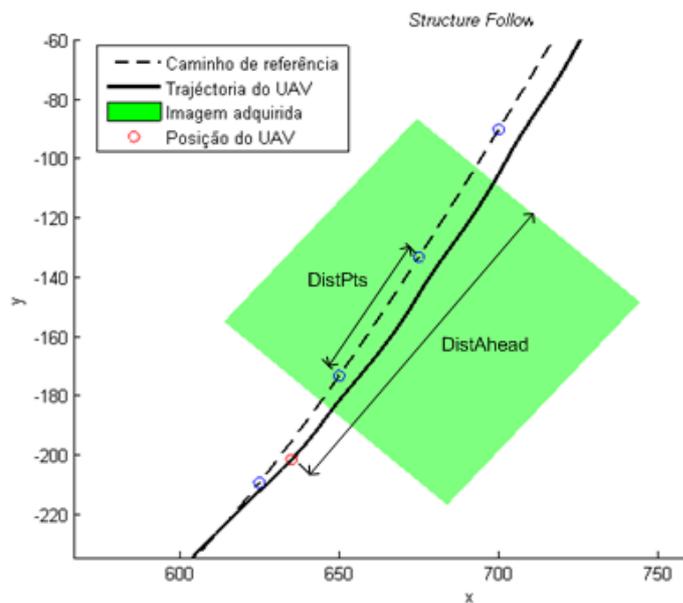


Figura 5-8 - Notação das distâncias.

Outra forma de resolver o problema é aumentar a altura a que o UAV efectua o seguimento, pois quanto maior a altura, maior o campo de visão. No entanto, esta forma depende das condições de operação do UAV, enquanto a alteração da distância entre pontos é uma questão de programação do algoritmo de processamento de imagem.

### 5.3 - Seguimento de uma curva viável com velocidade constante

O objectivo é projectar um controlador que direcciona o UAV na direcção do caminho de referência, proporcional à distância entre o UAV e o caminho de referência, ou seja, quando está longe o UAV deve ir pelo caminho mais curto (ângulo de intercepção aproximadamente 90°), quando está “perto”, começa a convergir para a orientação desejada, de forma “suave” (Figura 5-9).

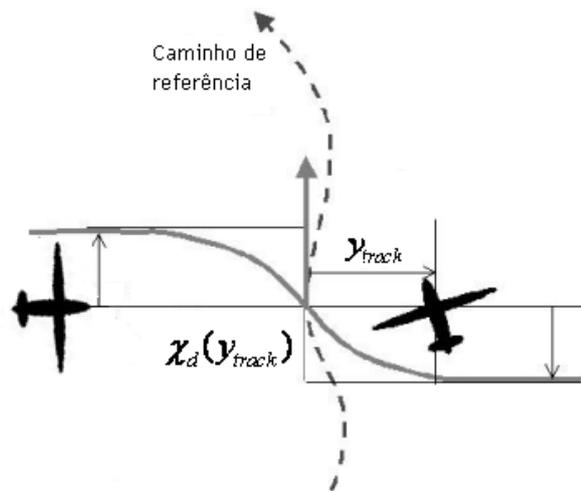


Figura 5-9 - Orientação desejada em função da distância  $y_{track}$ .

Uma função capaz de implementar estas características é  $\tan^{-1}$ . Assim, a orientação desejada é definida pela fórmula

$$\chi_d(y_{track}) = -\tan^{-1}(k \cdot y_{track}), \quad (5.1)$$

onde  $k = 1/\text{DistRef}$  é uma constante positiva. O parâmetro *DistRef* é a distância do ponto de referência no referencial do UAV. Este ponto é depois projectado na curva da estrutura (Figura 5-10).

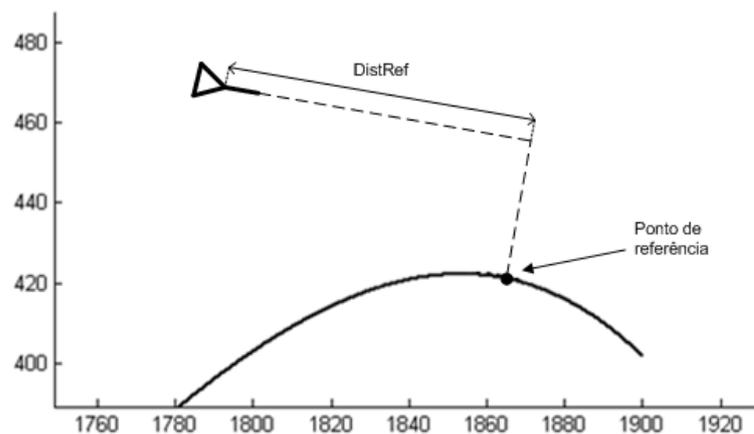


Figura 5-10 - Ponto de referência.

O ponto de referência funciona como um factor de antecipação e proporciona a transição “suave” do ângulo de intercepção, de aproximadamente  $90^\circ$ , para a orientação desejada (orientação do caminho no ponto de referência).

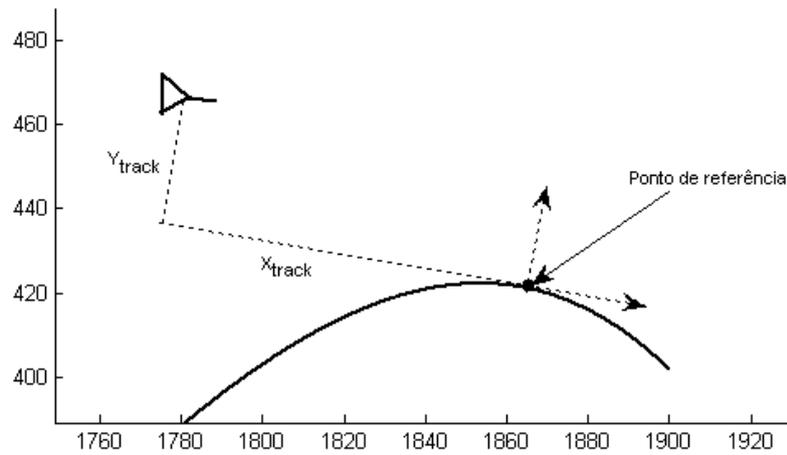


Figura 5-11 - Referencial do caminho.

A variável  $y_{track}$  é a posição do UAV no referencial do caminho (offset), com origem no ponto de referência (Figura 5-11).

**Teorema 4.** Se  $\chi = \chi_d(y_{track})$ , então  $y_{track} \rightarrow 0$ .

*Demonstração:* Utilizando a função candidata de Lyapunov

$$V = \frac{1}{2} y_{track}^2 \quad (5.2)$$

Fazendo a sua derivada,

$$\begin{aligned} \dot{V} &= y_{track} \dot{y} \\ \dot{V} &= y_{track} V_g \sin(\chi) \\ \dot{V} &= -y_{track} V_g \sin(\tan^{-1}(k \cdot y_{track})) \quad \blacksquare \end{aligned}$$

Como a derivada é negativa para  $y_{track} \neq 0$ , então  $y_{track} \rightarrow 0$ , num tempo finito, segundo o teorema de Lyapunov.

**Teorema 5.** A lei de controlo (5.3), desenvolvida através do método *sliding mode control*, é estável e  $y_{track} \rightarrow 0$ . Considerando a orientação desejada dada pela fórmula (5.1).

$$\dot{\chi} = -V_y \frac{k}{1 + (k \cdot y_{track})^2} - \kappa \cdot \text{sat}\left(\frac{s}{\varepsilon}\right) \quad (5.3)$$

*Demonstração:* A vantagem de utilizar o modelo cinemático do UAV é a sua ordem. Como  $\theta$  é um estado controlável e a variável de controlo  $\omega$  influencia linearmente este estado, é possível projectar um controlador *sliding mode* simples.

Assim a *sliding surface* (5.4) é a escolhida para o efeito desejado, ou seja, a convergência da orientação para o valor desejado.

$$s = \chi - \chi_d \quad (5.4)$$

Fazendo a derivada de (5.4) obtêm-se

$$\dot{s} = \dot{\chi} - \dot{\chi}_d \quad (5.5)$$

$$\dot{s} = \dot{\chi} + V_y \frac{k}{1 + (k \cdot y_{track})^2} \quad (5.6)$$

Sendo,

$$V = \frac{1}{2} s^2$$

E a sua derivada,

$$\dot{V} = s\dot{s}$$

$$\dot{V} = s(\dot{\chi} - \dot{\chi}_d)$$

$$\dot{V} = s\left(\dot{\chi} + V_y \frac{k}{1 + (k \cdot y_{track})^2}\right)$$

Igualando (5.6) a zero obtêm-se a componente equivalente da lei de controlo e acrescentando a componente descontínua, resulta em,

$$\dot{\chi} = -V_y \frac{k}{1 + (k \cdot y_{track})^2} - \kappa \operatorname{sgn}(s) \quad (5.7)$$

onde  $\kappa > 0$  e

$$\operatorname{sgn}(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$$

Então,

$$\dot{V} \leq -\kappa |s|$$

Podemos concluir, pelo teorema de estabilidade de Lyapunov, que o sistema é estável e  $s \rightarrow 0$ .

Considerando também o Teorema 5 concluí-se que,

$$s \rightarrow 0 \Rightarrow \chi \rightarrow \chi_d \Rightarrow y_{track} \rightarrow 0$$

Como referido na secção 2.3, a função *sign* produz *chattering*. Para reduzir este comportamento a função *sign* é substituída pela função *sat* (5.8), com o objectivo de “suavizar” a função *sign*.

$$sat(x) = \begin{cases} x & se |x| \leq 1 \\ sign(x) & outro \end{cases} \quad (5.8)$$

Assim a lei de controlo passa a ser representada por

$$\dot{\chi} = -V_y \frac{k}{1 + (k \cdot y_{track})^2} - \kappa \cdot sat\left(\frac{s}{\varepsilon}\right), \quad \blacksquare \quad (5.9)$$

onde  $\varepsilon > 0$  define a região limitadora em redor da *sliding surface*, com a finalidade de reduzir o comportamento *chattering*.

## 5.4 - Seguimento de uma curva viável com ajustamento da velocidade

Quando o UAV detecta à sua frente (distância pré-determinada e tendo em consideração a distância/tempo que demora a diminuir de velocidade) uma curva com um raio de curvatura inferior às capacidades actuais, e sabendo a velocidade mínima, logo também o raio de curvatura mínimo, se o raio de curvatura não passar esse limite é calculada a velocidade necessária para efectuar a curva.

Caso o raio de curvatura passe o limite então estamos perante o problema discutido no seguimento de uma curva não viável (secção 4.6).

A velocidade comandada é calculada em função do raio de curvatura exigido, pela fórmula (5.10), considerando um ângulo de pranchamento máximo de 25 graus.

$$V_c = \sqrt{g \cdot \tan(\phi) \cdot R_{curva}} \quad V_{\min} < V_c < V_{\max} \quad (5.10)$$

## 5.5 - Seguimento de uma curva não viável

**Teorema 6.** O Problema 5 é resolvido com um controlador híbrido, constituído pelos modos de operação: seguimento, *search*, intercepção, garantindo assim que o caminho é todo pesquisado.

*Demonstração:* Com o objectivo da área pesquisada incluir o caminho completo, projectou-se um controlador híbrido com três modos de operação:

- Seguimento
- *Search*
- Intercepção

A Figura 5-12 apresenta a informação necessária em cada modo, assim como a transição que o activa.



Figura 5-12 - Estrutura de informação do sistema híbrido (modos de operação, parâmetros e condições de transição).

### **Modo seguimento**

Quando este modo está activo, o UAV segue o caminho definido pelos pontos adquiridos, utilizando o controlador projectado para o seguimento de uma curva viável (secção 5.2).

O raio de curvatura do caminho é calculado uma distância à frente da posição actual do UAV. Caso este valor seja inferior ao valor limite do UAV, ou seja uma curva não viável (ver Definição 15), o sistema transita para o modo *search*.

### **Modo search**

Este modo, com a estrutura de informação ilustrada na Figura 5-13, realiza a pesquisa numa área com o objectivo de encontrar pontos que identifiquem a estrutura e criem um caminho viável, para depois calcular um ponto de intercepção e proceder ao seguimento da estrutura.

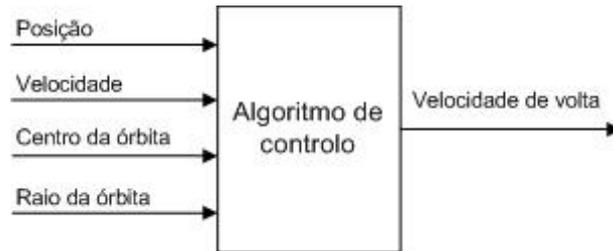


Figura 5-13 - Estrutura da informação do algoritmo de controlo do seguimento de órbitas.

A pesquisa é efectuada através de órbitas circulares, com centro na posição onde o limite do raio de curvatura foi ultrapassado.

Baseado em [18] é construída uma lei de controlo, pelo método *sliding mode control*, para executar o seguimento de órbitas circulares, para qualquer posição e orientação inicial do UAV.

A Figura 5-14 ilustra a geometria envolvida no seguimento de uma órbita circular.

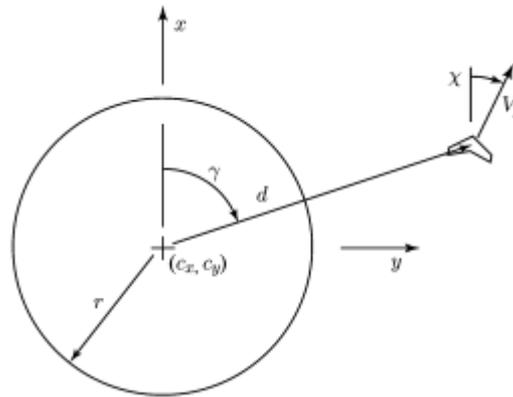


Figura 5-14 - Geometria do problema de seguimento de uma órbita circular [18].

Onde  $d$  é a distância do UAV ao centro da órbita, em metros;  $r$  é o raio da órbita pretendido, em metros;  $c_x$  e  $c_y$  é a posição do centro da órbita;  $\gamma$  é o ângulo entre a posição do UAV e o centro da órbita.

A lei de controlo deve comandar o UAV para o centro da órbita quando a distância ao centro da órbita é muito maior que o seu raio ( $d > 2r$ ) e quando passar este limite comandar para o seguimento da órbita. Estas duas condições podem ser expressas como

$$\begin{aligned} \chi_d &= \gamma - \pi \quad \text{quando } d > 2r \\ \chi_d &= \gamma - \frac{\pi}{2} \quad \text{quando } d = r \end{aligned} \quad (5.11)$$

Com base nisto, o curso desejado para seguimento no sentido horário é dado por

$$\chi_d = \gamma - \frac{\pi}{2} - \tan^{-1}(k(d - r)) \quad (5.12)$$

e para seguimento no sentido anti-horário dado por

$$\chi_d = \gamma + \frac{\pi}{2} + \tan^{-1}(k(d-r)), \quad (5.13)$$

onde  $k > 0$  define a razão de transição de  $\gamma - \pi \rightarrow \gamma - \frac{\pi}{2}$ .

Estando a trabalhar com órbitas circulares é conveniente alterar as coordenadas para o sistema polar, considerando a origem o centro da órbita. Observando a Figura 5-14 chega-se a:

$$\begin{aligned} x &= c_x + d \cos \gamma \\ y &= c_y + d \sin \gamma \end{aligned}$$

Fazendo a derivada e substituindo em (4.5), obtêm-se

$$\begin{aligned} \dot{d} &= V_g \cos(\chi - \gamma) \\ \dot{\gamma} &= \frac{V_g}{d} \sin(\chi - \gamma). \end{aligned} \quad (5.14)$$

**Teorema 7.** Considerando  $\tilde{d} = d - r$  e  $\chi = \chi_d(d)$ , então  $\tilde{d} \rightarrow 0$  assintoticamente.

*Demonstração:* Para provar, considere-se a seguinte função candidata de Lyapunov

$$V = \frac{1}{2} \tilde{d}^2.$$

Fazendo a sua derivada e tendo em conta (5.12), obtêm-se

$$\begin{aligned} \dot{V} &= \tilde{d} \dot{\tilde{d}} \\ \dot{V} &= \tilde{d} V_g \cos(\chi - \gamma) \\ \dot{V} &= \tilde{d} V_g \cos\left(-\frac{\pi}{2} - \tan^{-1}(k\tilde{d})\right). \end{aligned} \quad (5.15)$$

Sabendo que  $\sin(\theta) = \cos\left(\frac{\pi}{2} - \theta\right)$ , então (5.15) fica,

$$\dot{V} = -\tilde{d} V_g \sin(\tan^{-1}(k\tilde{d})) \quad (5.16)$$

Como a função  $\tan^{-1}$  e  $\sin$  tem valores entre  $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$ , então para qualquer valor  $\tilde{d} \neq 0$ , a equação (5.16) é negativa. Logo pode-se concluir pelo Teorema de Lyapunov que  $\tilde{d} \rightarrow 0$  assintoticamente, num tempo finito. ■

**Teorema 8.** Considerando a orientação desejada dada por (5.12), a lei de controlo (5.17), desenvolvida pelo método *sliding mode*, executa o controlo do UAV para órbitas circulares, no sentido horário. É estável e  $d \rightarrow r$ .

$$\dot{\chi} = \frac{V_g}{d} \sin(\chi - \gamma) - \frac{k}{1 + (k(d - r))^2} V_g \cos(\chi - \gamma) - \kappa \text{sat}\left(\frac{s}{\varepsilon}\right) \quad (5.17)$$

**Teorema 9.** Considerando a orientação desejada dada por (5.13), a lei de controlo (5.18), desenvolvida pelo método *sliding mode*, executa o controlo do UAV para órbitas circulares, no sentido anti-horário. É estável e  $d \rightarrow r$ .

$$\dot{\chi} = \frac{V_g}{d} \sin(\chi - \gamma) + \frac{k}{1 + (k(d - r))^2} V_g \cos(\chi - \gamma) - \kappa \text{sat}\left(\frac{s}{\varepsilon}\right) \quad (5.18)$$

*Demonstração:* A demonstração do Teorema 9 e 10 são semelhantes, por isso será apenas demonstrado o Teorema 9.

Considere-se a seguinte *sliding surface*,

$$s = \chi - \chi_d$$

A sua derivada é

$$\dot{s} = \dot{\chi} - \dot{\chi}_d \quad (5.19)$$

A derivada de (5.12) é

$$\dot{\chi}_d = \dot{\gamma} - \dot{d} \frac{k}{1 + (k(d - r))^2} \quad (5.20)$$

Igualando a zero a equação (5.19) e substituindo pelas equações (5.14) e (5.20), obtêm-se a componente equivalente da lei de controlo.

$$\dot{\chi} = \frac{V_g}{d} \sin(\chi - \gamma) - \frac{k}{1 + (k(d - r))^2} V_g \cos(\chi - \gamma) \quad (5.21)$$

Acrescentando a componente descontínua, resulta,

$$\dot{\chi} = \frac{V_g}{d} \sin(\chi - \gamma) - \frac{k}{1 + (k(d - r))^2} V_g \cos(\chi - \gamma) - \kappa \text{sat}\left(\frac{s}{\varepsilon}\right) \quad (5.22)$$

Escolhendo a seguinte função candidata de Lyapunov,

$$V = \frac{1}{2} s^2$$

A sua derivada é,

$$\dot{V} = s\dot{s}$$

$$\dot{V} = s(\dot{\chi} - \dot{\chi}_d)$$

Substituindo pelas fórmulas (5.20) e (5.22) resulta,

$$\dot{V} \leq -\kappa |s|$$

Pelo teorema de Lyapunov, concluí-se que o sistema é estável e  $s \rightarrow 0$ , num tempo finito.

Logo,

$$s \rightarrow 0 \Rightarrow \chi \rightarrow \chi_d \Rightarrow d \rightarrow r \quad \blacksquare$$

Agora que temos a lei de controlo para efectuar as órbitas circulares, é preciso responder às questões:

- Quando se transita para uma órbita maior?
- Quanto é o raio da nova órbita?
- Qual é o evento que deve acontecer para ocorrer a transição para o modo de intercepção?
- Quantos novos pontos são necessários?

A Figura 5-15 apresenta o diagrama lógico do funcionamento do modo *search*.

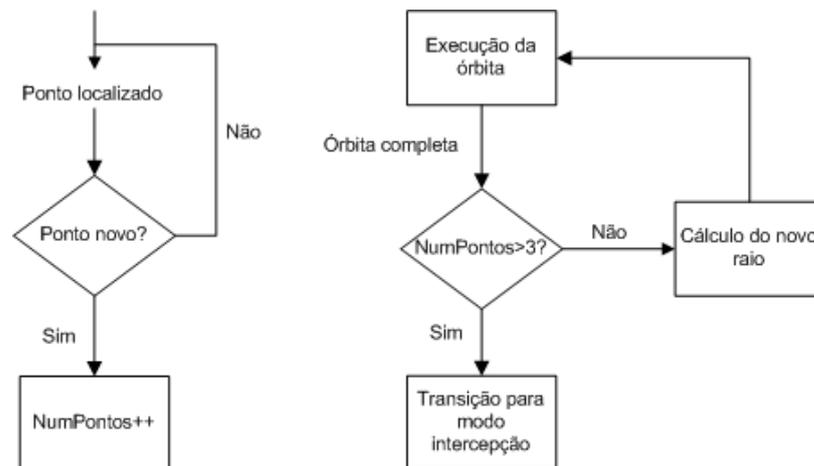


Figura 5-15 - Diagrama de funcionamento do modo *search*.

No modo *search* existem dois algoritmos a correrem em paralelo. Um encarregado de identificar pontos novos e outro a executar a órbita.

O algoritmo de identificação aumenta um contador (*NumPontos*) sempre que forem adquiridos pontos e estes ainda não estarem numa “memória” que guarda os pontos

adquiridos. Caso o ponto já se encontre nessa “memória”, o contador permanece com o mesmo valor. O valor do contador é colocado a zero sempre que é iniciado o modo *search*.

O raio da órbita começa com o valor do raio de curvatura mínimo do UAV e a transição para uma órbita maior ocorre quando, ao fim do UAV completar uma órbita, o número de pontos novos (*NumPontos*) seja insuficiente para calcular um novo caminho viável. Caso contrário, ocorre a transição para o modo de interceptação.

A justificação para o número de pontos novos necessário ter que ser maior que três é apresentada na secção 5.2.

O aumento do raio da órbita deve garantir que a área entre órbitas é pesquisada, como é ilustrado na Figura 5-16.

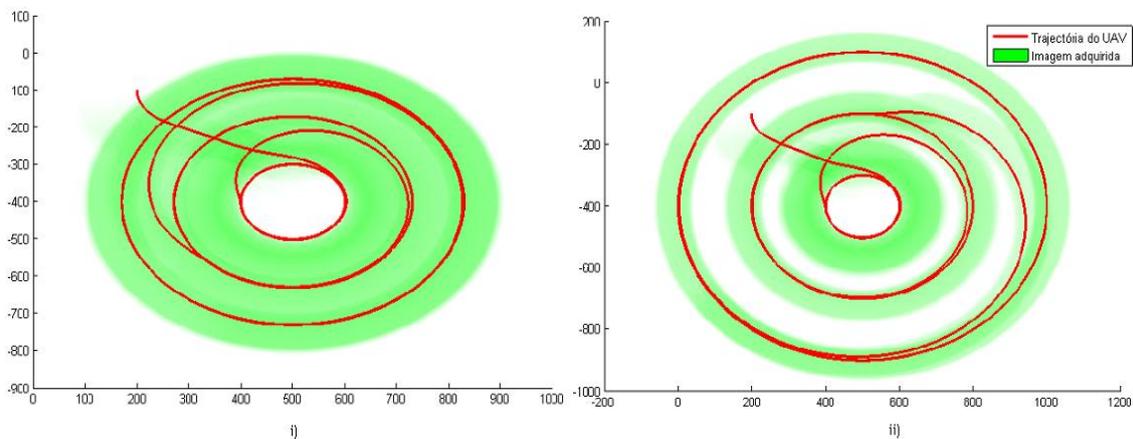


Figura 5-16 - Área pesquisada pelo UAV em órbitas circulares. Na situação i) a área entre órbitas é pesquisada, enquanto na situação ii) a área entre órbitas não é pesquisada completamente.

Logo, este aumento tem que ter em conta o campo de visão da câmara, que por sua vez é influenciado pela altura e pelos ângulos de *roll*, *pitch* e *yaw* do UAV, e também pelos ângulos *pan* e *tilt* da câmara (Figura 5-17).

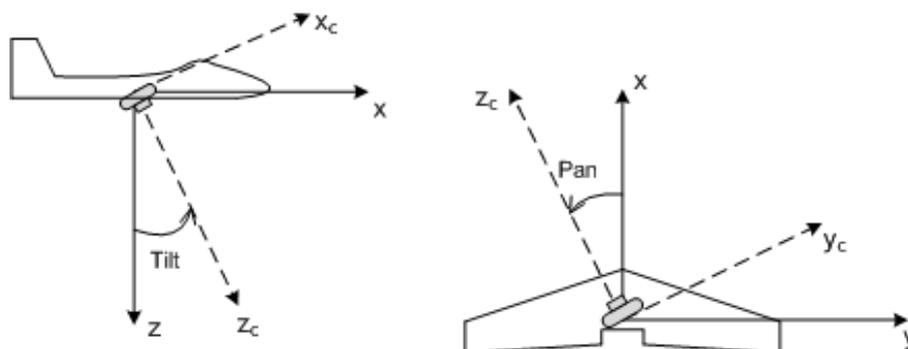


Figura 5-17 - Ângulos *tilt* e *pan* da câmara.

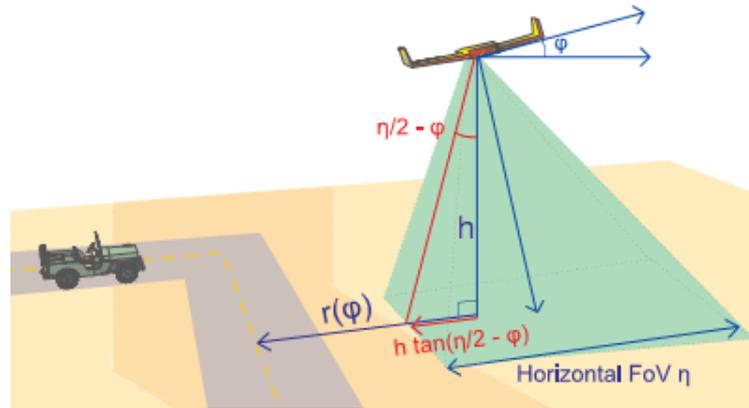


Figura 5-18 - Campo de visão da câmara [29].

**Teorema 10.** A fórmula (5.23) calcula o raio de órbita seguinte e garante que a área entre órbitas é totalmente pesquisada.

$$r_{novo} = r_{actual} + h \cdot \tan\left(\frac{\eta}{2} + \phi_{actual}\right) + h \cdot \tan\left(\frac{\eta}{2} - \phi_{actual}\right) \quad (5.23)$$

Onde  $\eta$  é o ângulo de abertura horizontal da câmara,  $h$  é a altura do UAV e  $\phi$  é o ângulo de pranchamento necessário para executar a órbita actual.

*Demonstração:* Observando a Figura 5-18, sabemos que a distância horizontal pesquisada no interior e exterior da órbita é dada por, respectivamente,

$$d_{interior} = h \cdot \tan\left(\frac{\eta}{2} - \phi\right) \quad e \quad d_{exterior} = h \cdot \tan\left(\frac{\eta}{2} + \phi\right) \quad (5.24)$$

Para garantirmos que a área entre órbitas é pesquisada, tem que verificar-se a seguinte condição,

$$r_{novo} - r_{actual} \leq d_{exterior(actual)} + d_{interior(novo)} \quad (5.25)$$

Substituindo (5.24) em (5.25), resulta

$$r_{novo} - r_{actual} \leq h \cdot \tan\left(\frac{\eta}{2} + \phi_{actual}\right) + h \cdot \tan\left(\frac{\eta}{2} - \phi_{novo}\right) \quad (5.26)$$

Como a órbita seguinte é sempre de raio maior, logo pela relação (4.7), o ângulo de pranchamento necessário para a seguir vai ser menor. Assim, minorando (5.26), substituindo  $\phi_{novo} = \phi_{actual}$ , resulta

$$r_{novo} = r_{actual} + h \cdot \tan\left(\frac{\eta}{2} + \phi_{actual}\right) + h \cdot \tan\left(\frac{\eta}{2} - \phi_{actual}\right) \quad \blacksquare$$

## Modo interceptação

O modo de interceptação é activado quando é detectado um ou mais segmentos do caminho viáveis, posterior(es) ao ponto onde se “perdeu contacto” com o caminho. É calculado o ponto de interceptação e a orientação desejada neste.

O método de interceptação consiste, principalmente, em duas fases, denominadas por *Outbound* e *Inbound*, afastamento e aproximação do ponto de interceptação, respectivamente (Figura 5-19).

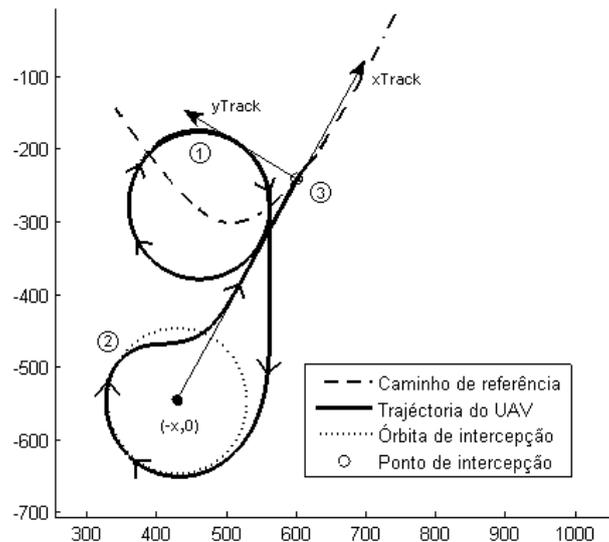


Figura 5-19 - Modo interceptação. Em 1) transita para a fase de *Outbound*, em 2) entra na fase de *Inbound* e em 3) transita para o modo de seguimento do caminho de referência.

### **Outbound**

Esta fase tem o objectivo de facilitar o alinhamento posterior do UAV com a orientação da curva, no ponto de interceptação.

Nesta fase são calculadas as coordenadas do ponto  $(-x,0)$ , representado no referencial da curva (com origem no ponto de interceptação e o eixo horizontal coincidente com a orientação no ponto), no referencial inercial. A coordenada  $x$  é o parâmetro que estabelece a distância entre o ponto de interceptação e o centro da órbita que o UAV irá executar para voltar na direcção do ponto de interceptação.

Quando o UAV se encontrar a executar a órbita, com a orientação aproximada do ponto de interceptação e à distância  $x$  deste, ocorre a transição para a fase de *Inbound* (número 2 na Figura 5-19).

### **Inbound**

Nesta fase é activado um controlador para seguir uma recta com a orientação do caminho no ponto de interceptação e que passa por este. Este controlador é desenvolvido através da adaptação do controlador de seguimento de um caminho (secção 5.2).

A adaptação consiste na utilização de um ponto de referência fixo (ponto de intercepção), enquanto o controlador para o seguimento do caminho utiliza um ponto de referência móvel (a uma distância constante à frente do UAV). Outra diferença é a utilização de um ganho variável com a distância.

Quando o UAV estiver a uma distância predefinida do ponto é activado o modo de seguimento, transição número 3 na Figura 5-19.

**Teorema 11.** Considerando a orientação desejada dada pela fórmula (5.27), então a lei de controlo (5.28), desenvolvida através do método *sliding mode control*, efectua o seguimento de uma linha com uma orientação  $\psi_{track}$ .

$$\chi_d(y_{track}) = \psi_{track} - \tan^{-1}(k \cdot y_{track}) \quad (5.27)$$

$$\dot{\chi} = -V_y \frac{k}{1 + (k \cdot y_{track})^2} - \kappa \cdot \text{sat}\left(\frac{s}{\varepsilon}\right) \quad (5.28)$$

Onde  $k = 1 / \text{Dist}$  e o parâmetro  $\text{Dist}$  é a distância a que o UAV está do ponto de intercepção e  $\psi_{track}$  é a orientação a ser seguida.

**Demonstração:** A demonstração é igual à do Teorema 6.



# Capítulo 6

## Simulação e resultados

Neste capítulo descreve-se o ambiente de simulação e apresenta-se um plano de testes e resultados obtidos, para os vários problemas descritos no capítulo 4.

Em cada secção, correspondente a um problema, será realizada uma análise e discussão dos resultados obtidos.

### 6.1 - Ambiente de simulação

O ambiente de simulação decorre no software *Matlab/Simulink*<sup>TM</sup>.

Inicialmente, utilizaram-se a *Curve Fitting Toolbox*<sup>TM</sup> e a *Spline Toolbox*<sup>TM</sup> como ferramentas de estudo dos vários métodos de aproximação de pontos.

A *Curve Fitting Toolbox*<sup>TM</sup> é uma interface gráfica e um conjunto de funções M-file, com funcionalidades de:

- Pré-processamento de dados: exclusão, selecção e filtragem de dados;
- Processamento: modelos paramétricos (polinómios, exponenciais) e não paramétricos (interpolação);
- Pós-processamento: análise estatística, interpolação, extrapolação.

No desenvolvimento e simulação dos sistemas híbridos utilizou-se a ferramenta *Stateflow*<sup>TM</sup>, em conjunto com o *Simulink*<sup>TM</sup>. Esta permite simular sistemas guiados por eventos (*event-driven systems*), baseados na teoria das máquinas de estados finitos.

Para a análise dos dados gerados na simulação criaram-se algoritmos em ficheiros M-file.

O diagrama de blocos que simula o modelo do UAV está ilustrado no Anexo A. Este modelo tem como sinais de entrada:

- Condições iniciais (posição, altura, velocidade e os ângulos Euler);
- Velocidade de volta;
- Direcção e velocidade do vento.

E sinais de saída:

- Posição;
- Altura;
- Velocidade ( $V_x$ ,  $V_y$ ) relativa ao solo;
- Yaw;
- Ângulo de pranchamento.

A modelação do vento é composta por uma componente contínua e uma componente aleatória, quer para a direcção, quer para a velocidade. A componente aleatória simula rajadas. No Anexo D estão ilustrados os vários blocos constituintes do modelo do vento.

## 6.2 - Aproximação recursiva de pontos obtidos de uma imagem de vídeo

A simulação da imagem adquirida da câmara e da aproximação recursiva de pontos é realizada em dois blocos, *Camera* e *Curve fitting* respectivamente (ver Anexo A).

A simulação da câmara corre um ciclo que verifica se algum dos pontos que identificam a estrutura (pré-definidos) está dentro da projecção da câmara (ver secção 5.1). Se algum dos pontos estiver contido na projecção, as suas coordenadas são adicionadas em vectores “memória” (guarda as coordenadas dos últimos pontos localizados), denominados por *CamDataX* e *CamDataY*, coordenadas  $x$  e  $y$  dos pontos, respectivamente. Estes vectores estão definidos com dimensão sete.

Ao ser adicionado um ponto é também aumentado um contador (*NumPontos*), que tem o objectivo de “informar” o sistema híbrido quantos pontos novos foram adquiridos, quando este está no modo *search*.

No fim, o vector *CamDataX* é ordenado por ordem crescente e o vector *CamDataY*, acompanha a ordenação, para que as coordenadas  $x$  e  $y$  continuem sincronizadas. Esta ordenação é necessária para que a interpolação dos pontos funcione.

O bloco *Curve fitting* recebe como parâmetros a posição do UAV e os vectores *CamDataX* e *CamDataY*. Efectua a interpolação dos pontos por um *spline* cúbica, da qual resulta uma matriz  $6 \times 4$ , correspondente a um polinómio cúbico entre cada dois pontos.

A próxima fase é seleccionar o polinómio/segmento a enviar para o algoritmo de controlo do UAV para efectuar o seu seguimento. Sabendo a posição do UAV e a distância ao ponto de referência (parâmetro do algoritmo de controlo), executa-se um conjunto de *if-else* para saber em qual dos segmentos está contido o ponto de referência. O polinómio/segmento escolhido é enviado para o algoritmo de controlo de seguimento de uma curva.

### 6.3 - Seguimento de uma curva viável com velocidade constante

O ângulo de orientação proveniente do modelo do UAV tem um valor no intervalo  $[-\pi \ \pi]$ , assim os cálculos que envolvam ângulos estão normalizados para este intervalo.

Na Figura 6-1 estão ilustrados os resultados do teste da influência da distância do ponto de referência. Como seria de esperar, distâncias pequenas levam a “oscilações” maiores, pois este parâmetro funciona como um factor de antecipação. Logo, quanto maior a distância, mais “suave” é a aproximação. Contudo, distâncias muito grandes produzem um aumento do erro lateral, como se vê no caso quando a distância é de 30m.

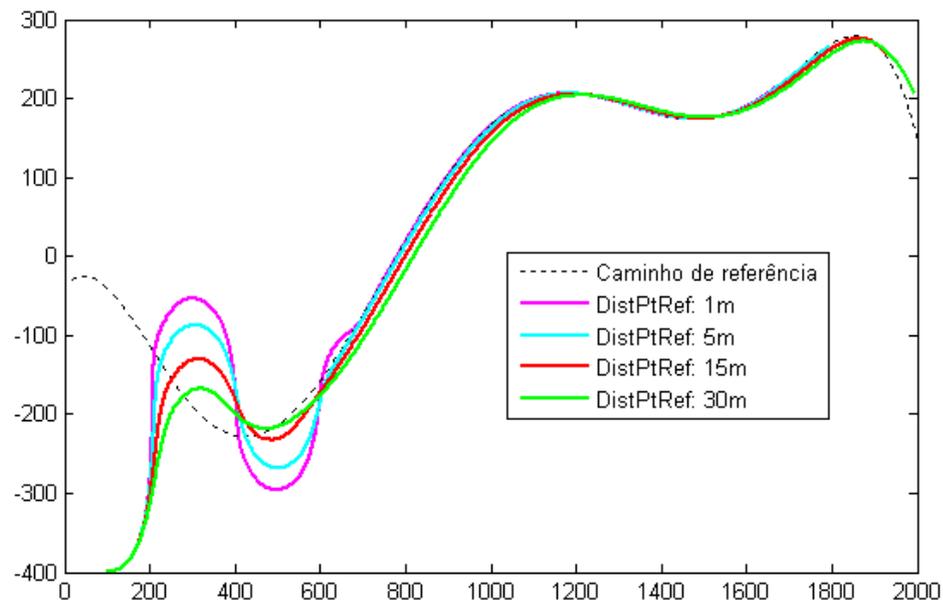


Figura 6-1 - Teste do controlador de seguimento de curvas para várias distâncias do ponto de referência.

No modelo do UAV existem dois tipos de velocidade: velocidade indicada ( $V_{IAS}$ ) e velocidade relativa ao solo ( $V_g$ ). A velocidade indicada é a dada pela seguinte fórmula

$$V_{IAS} = V_g + V_{Vento},$$

sendo uma soma se o vento é de frente e subtração se o vento é de cauda. A velocidade indicada corresponde à velocidade comandada.

Assim, se comandar uma velocidade de 20m/s e existir vento de cauda resulta numa velocidade maior em relação ao solo, logo as curvas do UAV vão ser mais prolongadas. O contrário se passa para vento de frente, em que as curvas são realizadas num espaço mais pequeno.

Este efeito do vento é visível nos resultados ilustrados na Figura 6-2. A simulação foi realizada para uma velocidade comandada de 20m/s e em três situações de vento:

- Sem vento;
- Com vento de cauda (Velocidade: 5m/s e direcção: 225°);
- Com vento de frente (Velocidade: 5m/s e direcção: 45°).

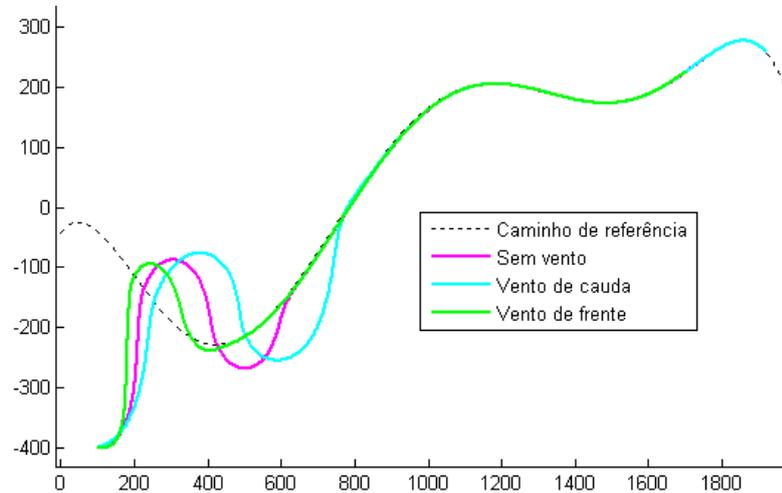


Figura 6-2 - Teste do controlador de seguimento de curvas na presença de vento.

Como já referido na secção 5.3, a função *sign* é aproximada por uma função de saturação. Utilizando-se a ideia descrita em [30], a função escolhida para a aproximação é

$$\text{sgn}(s) = \frac{s}{|s| + \delta},$$

onde  $\delta > 0$ , denominado como o parâmetro de eliminação de *chattering*. Na Figura 6-3 está ilustrado os resultados obtidos na variação deste parâmetro. Conclui-se que quanto maior for o seu valor, mais suave é a aproximação à curva.

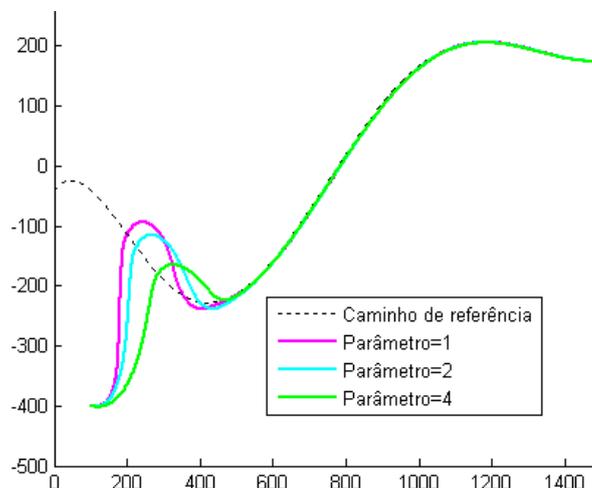


Figura 6-3 - Teste do controlador de seguimento de curvas para vários valores do parâmetro de eliminação de *chattering*.

## 6.4 - Seguimento de uma curva viável com ajustamento da velocidade

Neste problema a velocidade linear é também controlada, utilizando a fórmula (5.10). A velocidade está limitada a um valor mínimo e máximo, definidos na simulação com 12 e 20, respectivamente.

Na Figura 6-4 e 6-5 é ilustrada a trajectória do UAV com velocidade constante e com ajustável, e a velocidade comandada. É visível que das três zonas com curvaturas maiores, apenas uma não é seguida, pois a velocidade exigida é demasiado baixa. Neste caso, a solução seria a apresentada no seguimento de uma curva não viável.

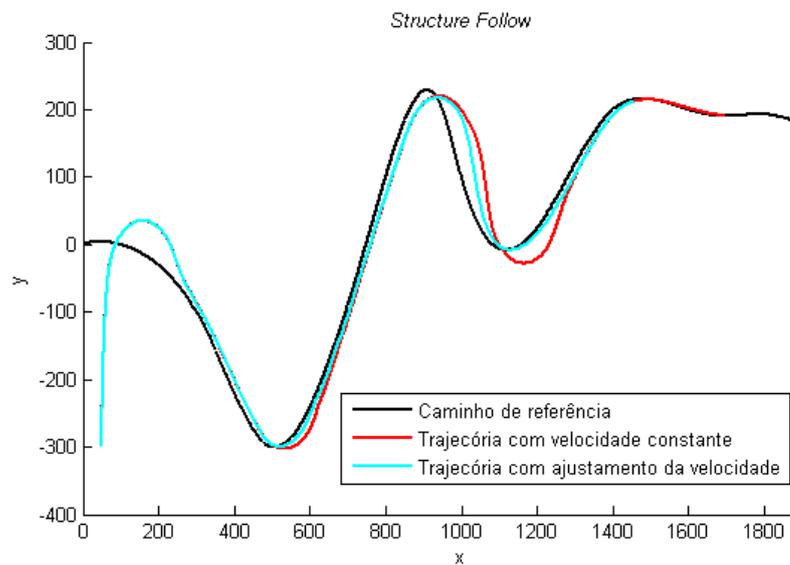


Figura 6-4 - Comparação da trajectória com velocidade constante e com ajustamento.

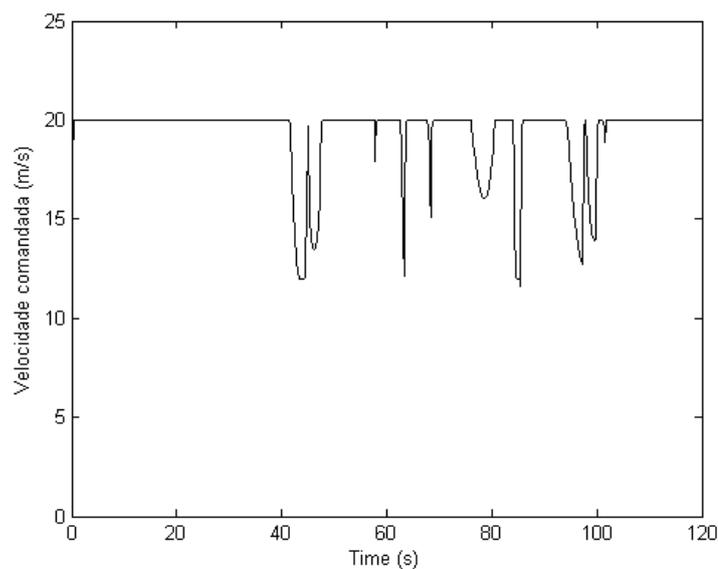


Figura 6-5 - Velocidade comandada.

Como o modelo do UAV não modela as inércias, a velocidade varia quase instantaneamente. Na prática tal não acontece, o que pode levar o UAV a não conseguir reduzir a velocidade antes de chegar à curva. Uma solução é calcular o raio da curvatura a uma distância variável do UAV, em função da velocidade, ou seja, quanto maior for a velocidade maior terá que ser a distância.

## 6.5 - Seguimento de uma curva não viável

Além do controlador para seguimento da estrutura, neste problema desenvolveu-se também um controlador para seguimento de órbitas e outro para seguimento de uma linha com uma orientação desejada. Apresentam-se de seguida alguns testes efectuados aos controladores.

### Modo search

Para testar o controlador do seguimento de órbitas fizeram-se três testes:

- **Teste 1:** Realizar uma órbita com centro nas coordenadas (300,-400) e raio de 100m, no sentido horário. Seguido de uma órbita no mesmo centro, mas com raio de 250m e por fim uma órbita nas coordenadas (400,100), no sentido anti-horário.
- **Teste 2:** Realizar uma órbita com centro longe da posição inicial, com coordenadas (1000,-100).
- **Teste 3:** Começar no centro da órbita, no sentido anti-horário e depois mudar para uma órbita com raio maior e sentido horário.
- **Teste 4:** Seguimento da órbita com vento de 15m/s, direcção 180 graus e rajadas activas.

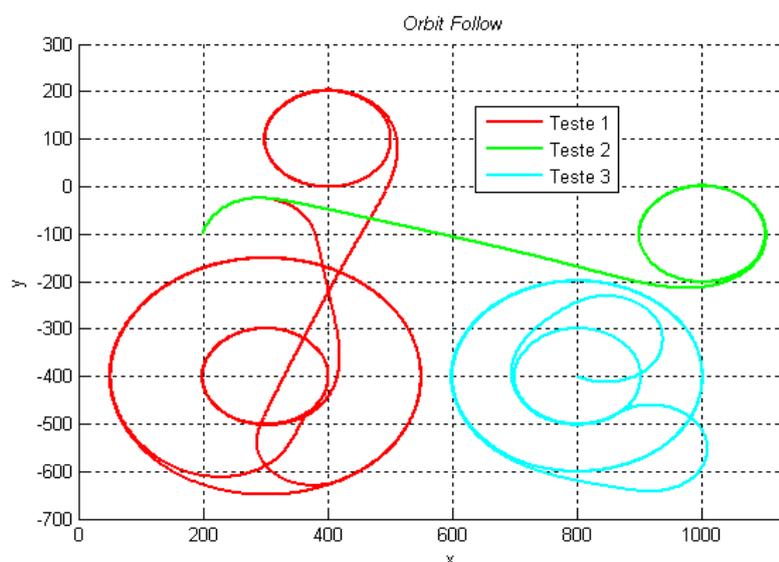


Figura 6-6 - Testes efectuados ao controlador de seguimento de órbitas.

Da análise da Figura 6-6 e 6-7 verifica-se que o controlador produz resultados aceitáveis para todos os testes efectuados. No teste com vento e rajadas (Figura 6-7) verifica-se apenas um ligeiro erro, em comparação ao teste sem vento.

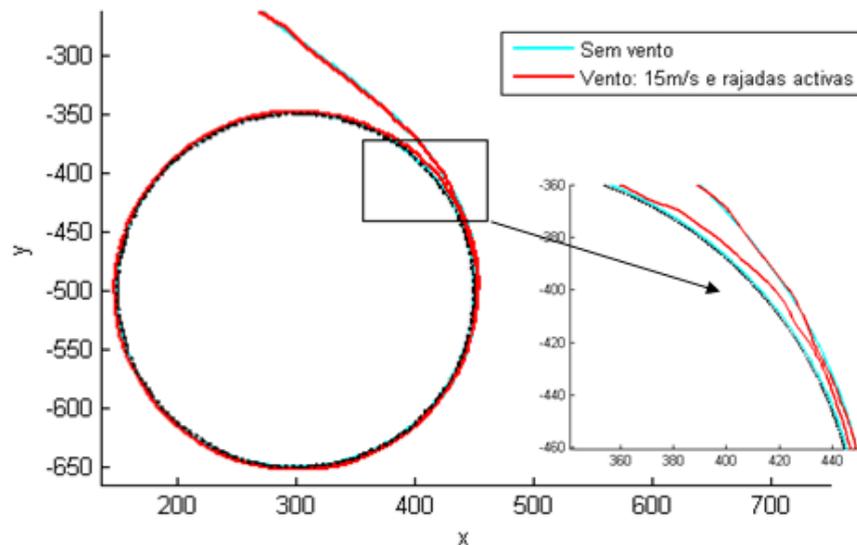


Figura 6-7 - Seguimento de órbita com vento de 15m/s vindo da direcção 180 graus.

Na projecção do campo de visão da câmara fizeram-se os seguintes testes:

- **Teste 1:** Projecção do campo de visão para vários ângulos de *pitch*, *bank*, *yaw*, *tilt* e *pan*.
- **Teste 2:** Projecção para vários raios de órbitas (ilustração do teste ao Teorema 11).

A câmara simulada tem as seguintes características:

- Ângulo de abertura horizontal =  $50^\circ$
- Ângulo de abertura vertical =  $36,5^\circ$

A Figura 6-8 apresenta a projecção do campo de visão para vários ângulos de atitude do UAV e da câmara.

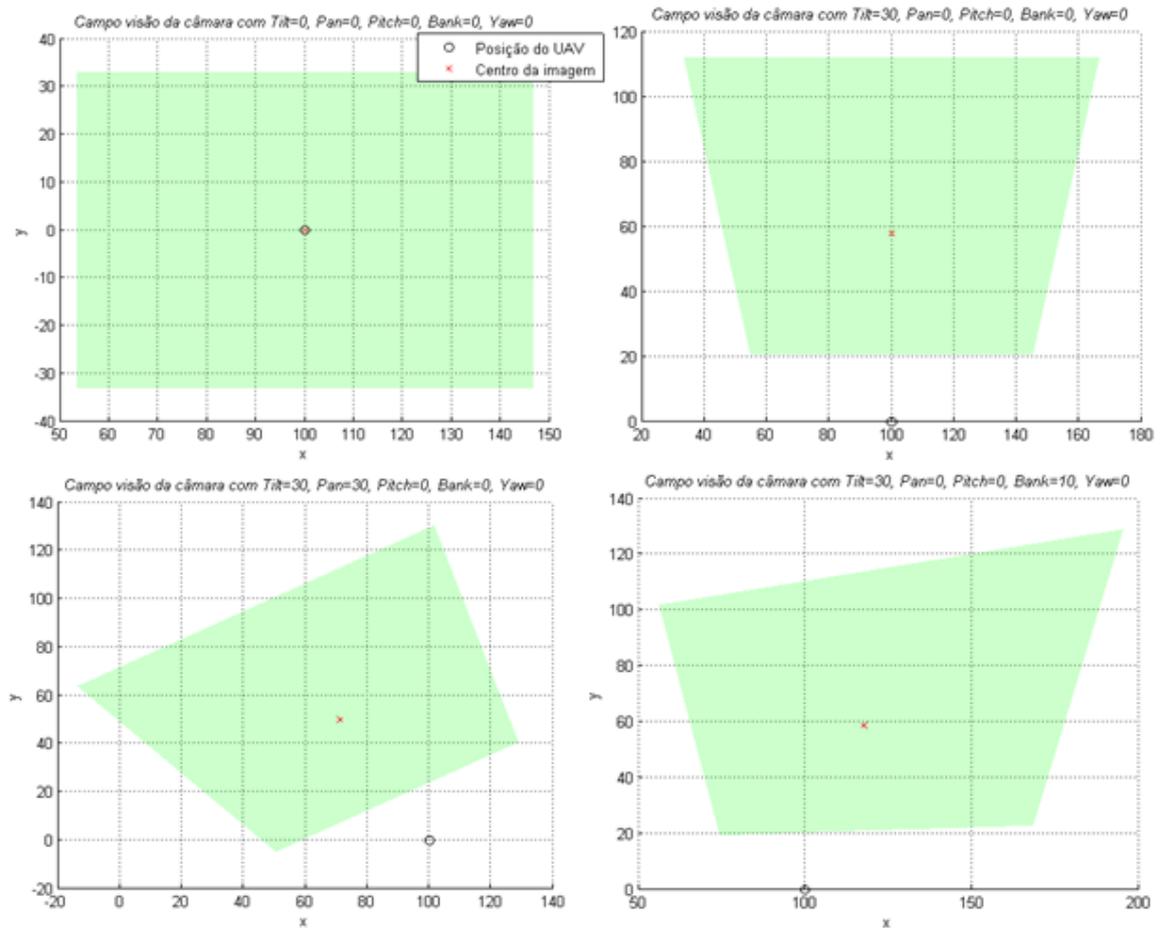


Figura 6-8 - Projecção do campo de visão da câmara para vários ângulos de *pitch*, *bank*, *yaw*, *tilt* e *pan*.

A Figura 6-9 ilustra a área adquirida em três órbitas, sendo o cálculo do raio da órbita seguinte realizado através da fórmula (5.23). Observa-se que o objectivo da área entre órbitas ser pesquisada é cumprido, existindo uma pequena parte que é pesquisada por duas órbitas.

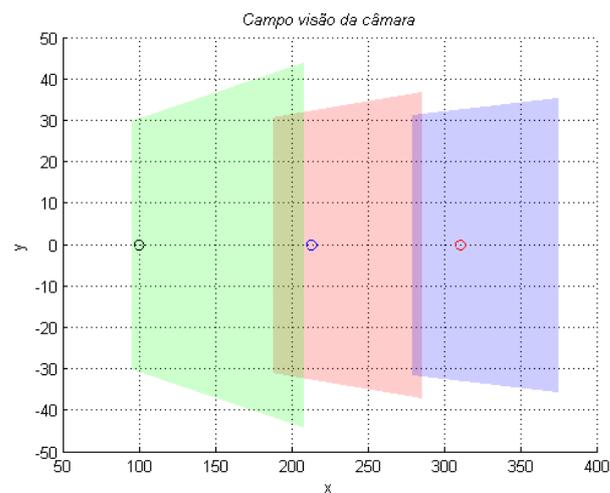


Figura 6-9 - Sequência de campos de visão à altura de 100m e velocidade de 20m/s, a começar com uma órbita de raio 100m (verde) e as órbitas seguintes com raios calculados por (5.23) (vermelho e azul).

### Modo de intercepção

A distância ao ponto de intercepção (centro da órbita) deve ser superior ao raio da órbita do modo de intercepção, de modo a que o UAV tenha uma distância suficiente para se alinhar com a recta que passa no ponto de intercepção.

A Figura 6-10 apresenta o comportamento do UAV para vários ganhos  $k$  (ver fórmula (5.28)). Observa-se que quanto maior o ganho, maior é a tendência para interceptar a linha. Porém a partir de um valor começa a introduzir oscilações, que pode levar o UAV a passar o ponto de intercepção sem estar perfeitamente alinhado com a orientação desejada neste ponto.

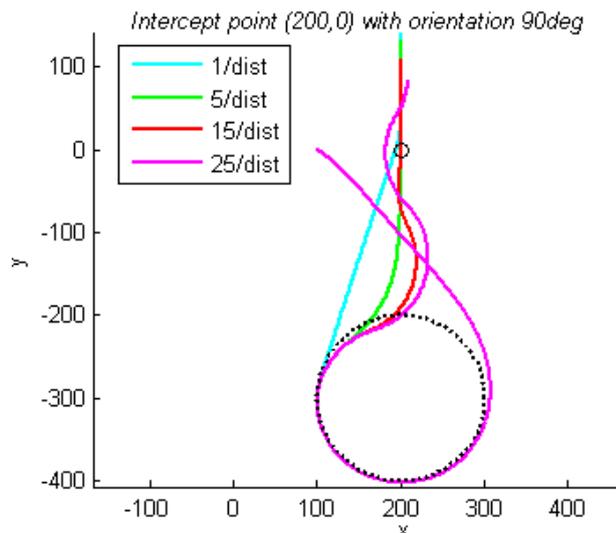


Figura 6-10 - Modo intercepção para vários ganhos de distância e com a distância ao ponto de intercepção (centro da órbita) de 300m.

Na Figura 6-11 é ilustrado o comportamento do controlador de intercepção para linhas com várias orientações.

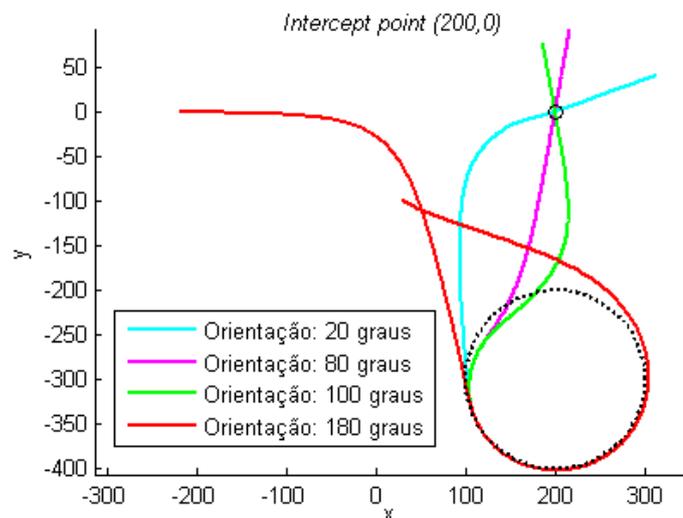


Figura 6-11 - Modo de intercepção para vários ângulos de intercepção.

Ao analisar a figura é visível que o comportamento do controlador no seguimento da linha com uma orientação de 180 graus (linha vermelha) não é desejável, pois ele intercepta a linha num ponto à frente do ponto de intercepção.

Isto acontece porque no momento em que o controlador é activado o UAV encontra-se à frente do ponto de intercepção, isto no referencial da curva (secção 4.6), ou seja, com o eixo dos xx alinhado com a orientação da curva no ponto de intercepção.

É por causa deste comportamento que a escolha do centro da órbita de intercepção situa-se na parte negativa do eixo dos xx.

### **Controlador híbrido**

Para o seguimento de um caminho com curvas não viáveis recorreu-se à ferramenta *Stateflow*<sup>TM</sup> para desenvolver e simular uma máquina de estados, ilustrada no Anexo B. Este controlador faz a comutação entre os vários controladores, conforme a ocorrência de determinados eventos.

As próximas figuras ilustram o resultado da simulação do controlador híbrido desenvolvido. A distância ao ponto de intercepção definida é de 300m (Figura 6-12) e de 200m (Figura 6-13), ambas com o UAV a uma altura de 100m. Em seguida é explicada a ordem de acontecimentos que ocorrem na simulação da Figura 6-12.

- 1- A simulação começa com o UAV nas coordenadas (100,-300), como ainda não tem pontos de localização da estrutura, é dada a ordem para ir procurar na zona das coordenadas (200,0);
- 2- Começa o modo *search* e ao fim de uma volta verifica que já tem o número de pontos suficiente para fazer o seguimento. Entra no modo de intercepção.
- 3- Calcula o ponto de intercepção e a orientação pretendida nele. Com base nestes valores calcula as coordenadas do centro da órbita.
- 4- Quando chega ao ponto de intercepção comuta para o controlador de seguimento do caminho.
- 5- É detectada uma curva não viável e entra no modo *search*. Repete-se os procedimentos descritos nos passos 2, 3 e 4.
- 6- É detectada outra curva não viável e repete-se os procedimentos.
- 7- Neste caso, como ao fim de uma volta ainda não existe um número de pontos novos suficientes para continuar o seguimento, é dada a ordem para fazer outra órbita de raio maior.

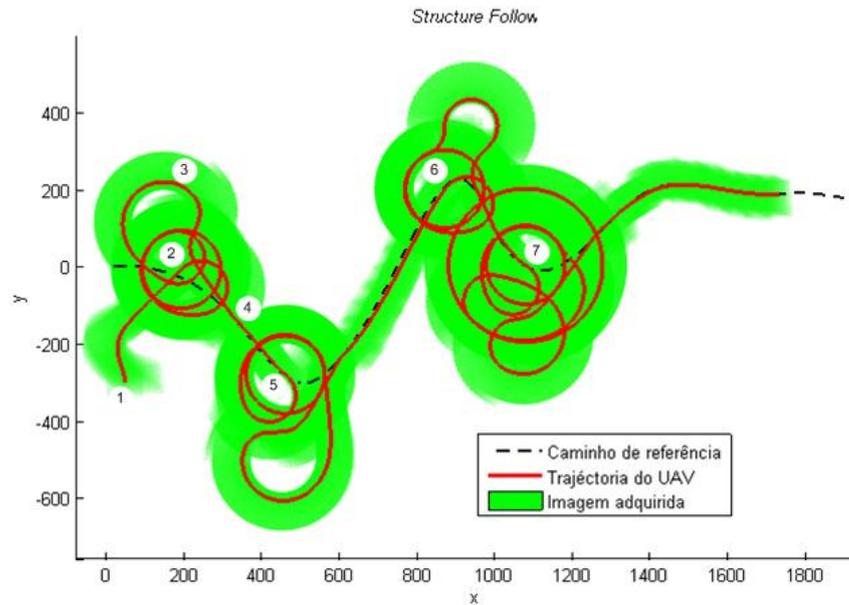


Figura 6-12 - Seguimento de um caminho com curvas não viáveis, para uma altura de 100m e uma distância ao ponto de interceptação de 300m.

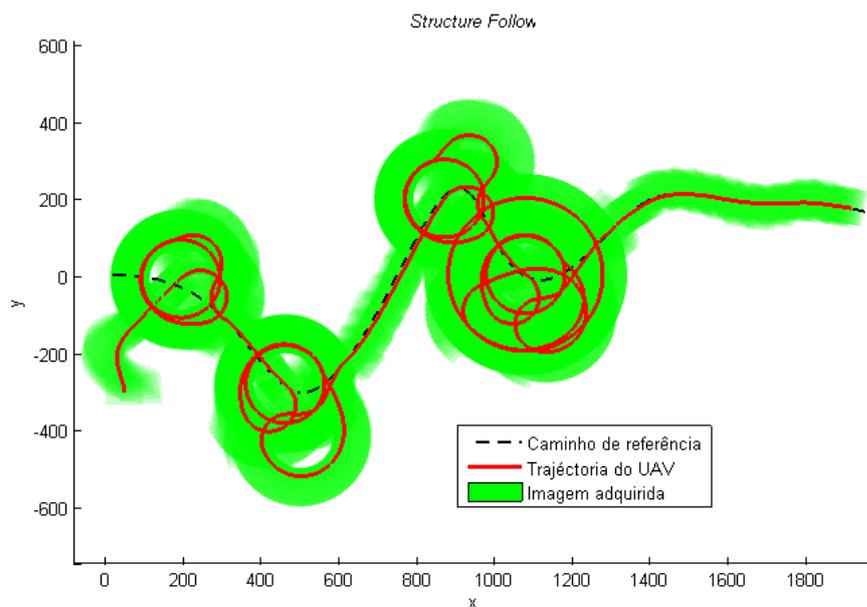


Figura 6-13 - Seguimento de um caminho com curvas não viáveis, para uma altura de 100m e uma distância ao ponto de interceptação de 200m.

Da análise das figuras conclui-se que existem três curvas não viáveis. O que origina a entrada no modo *search* por três vezes.

Na Figura 6-14 é visível que o UAV apenas entrou no modo *search* duas vezes, devido à detecção de curvas não viáveis. Isto deve-se ao facto de o UAV nesta simulação efectuar o seguimento a uma altura de 200m, logo o campo de visão é maior. Por ser maior, quando o UAV está a efectuar a última operação de *search* detecta pontos além da última curva não viável.

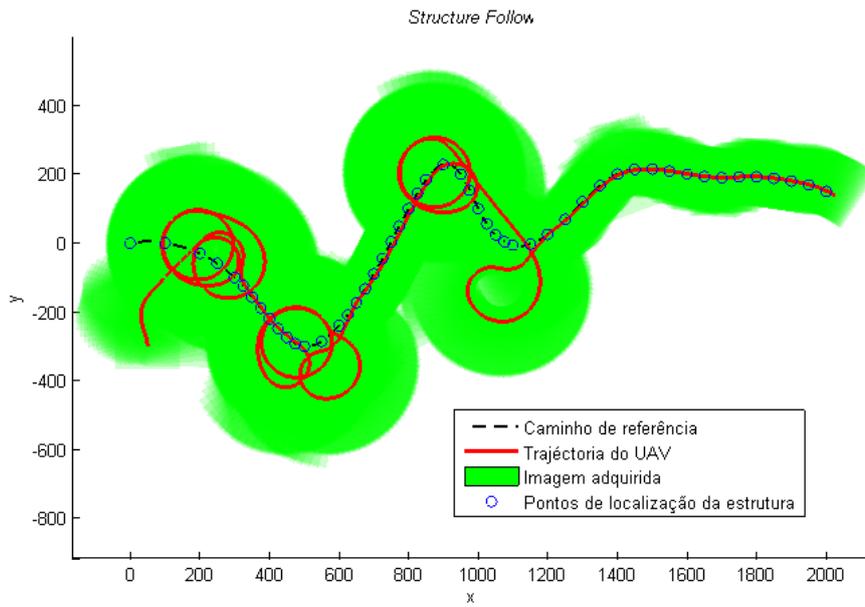


Figura 6-14 - Seguimento de um caminho com curvas não viáveis, para uma altura de 200m e uma distância ao ponto de intercepção de 200m

# Capítulo 7

## Conclusões e trabalho futuro

Neste capítulo são apresentadas as conclusões do trabalho desenvolvido ao longo da dissertação, o que se aprendeu e passos para futuro desenvolvimento.

### 7.1 - Conclusões

O projecto de dissertação desenvolvido abrange principalmente duas áreas: planeamento do caminho e o controlo necessário para o seu seguimento.

Na área do planeamento do caminho estudaram-se alternativas como, a aproximação de pontos por polinómios ou a interpolação por uma *spline*, sempre tendo em conta a restrição da velocidade de volta do UAV. A aproximação por polinómios apesar de ser mais fácil manipulação, não garante a passagem por todos os pontos e não é viável quando se pretende efectuar o seguimento da estrutura ao mesmo tempo que são adquiridos pontos. Por estas razões, o planeamento do caminho incidiu mais no estudo da interpolação por uma *spline*, realizando a interpolação consoante são adquiridos pontos.

No estudo da câmara foi implementada a projecção do campo de visão no plano horizontal, tendo em conta os ângulos *pitch*, *roll* e *yaw* do UAV, e os ângulos *pan* e *tilt* da câmara.

Foram desenvolvidos controladores e demonstrações formais, baseados na teoria de *sliding mode control*, para seguimento de curvas viáveis, seguimento de órbitas circulares e seguimento de uma recta que passa por um ponto com determinada orientação. Para o seguimento de uma curva não viável desenvolveu-se um controlador híbrido com três modos de funcionamento: seguimento, *search* e intercepção.

Os controladores foram simulados em Matlab/Simulink™, com uso da ferramenta *Stateflow* na simulação do sistema híbrido, com obtenção de resultados aceitáveis no seguimento de um caminho com curvas viáveis e não viáveis.

## 7.2 - Trabalho futuro

Os resultados obtidos neste projecto são apenas um pequeno passo nesta área dos veículos aéreos não tripulados. Os próximos passos deste projecto devem ter como referência o estudo realizado nesta dissertação. Assim, passos futuros relacionados com este projecto seriam a optimização dos controladores, pois apesar de apresentarem bons resultados, existe sempre melhorias e extensões que podem ser realizadas.

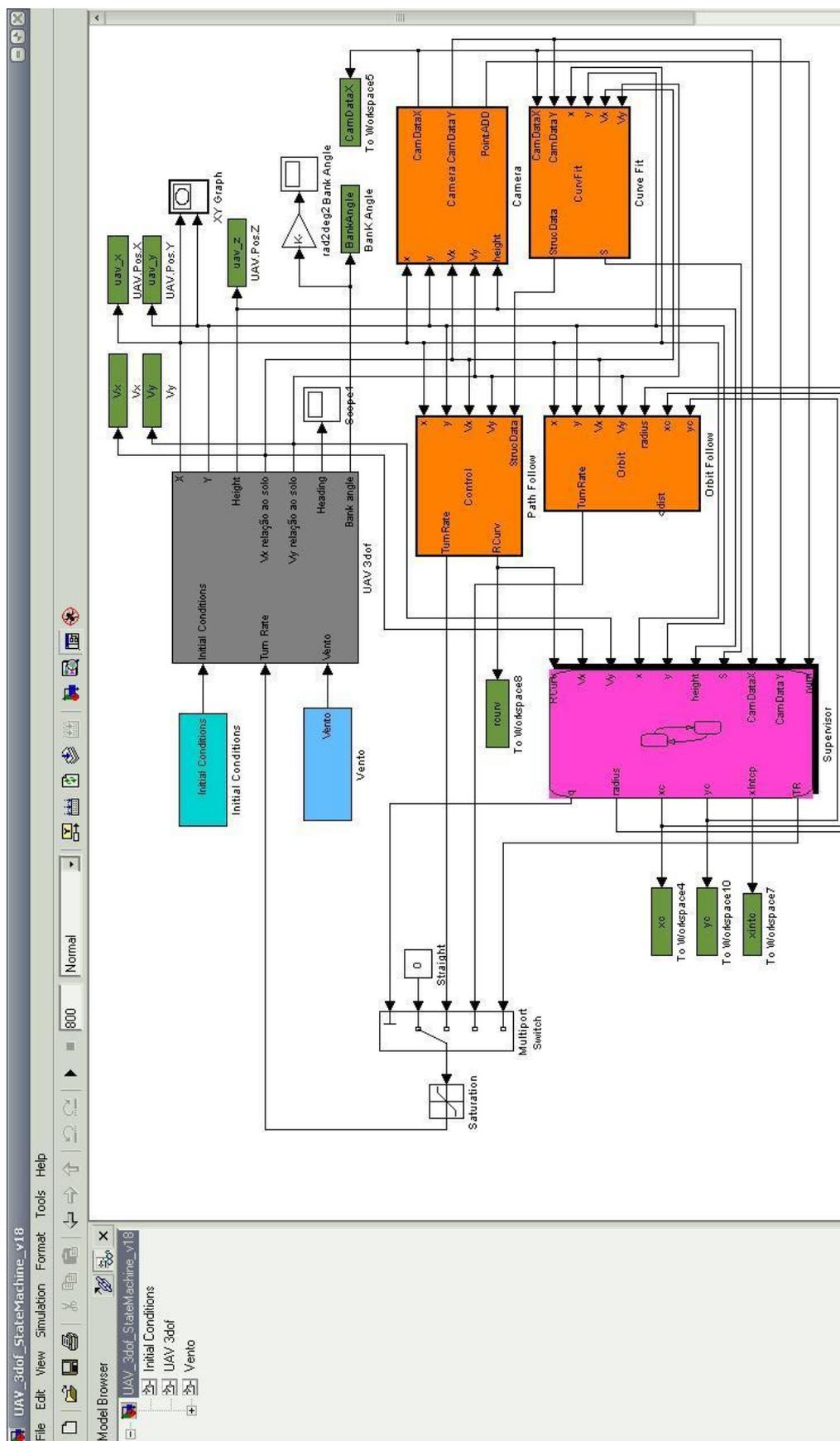
A simulação dos controladores realizou-se com um modelo 3-dof simplificado, a implementação de um modelo de 6-dof produziria resultados mais reais e, ao mesmo tempo, testava a robustez dos controladores desenvolvidos, interpretando a transição de um modelo para outro como uma perturbação.

Outro ponto a trabalhar é a análise de estabilidade do controlador híbrido, usando a análise de estabilidade de Lyapunov.

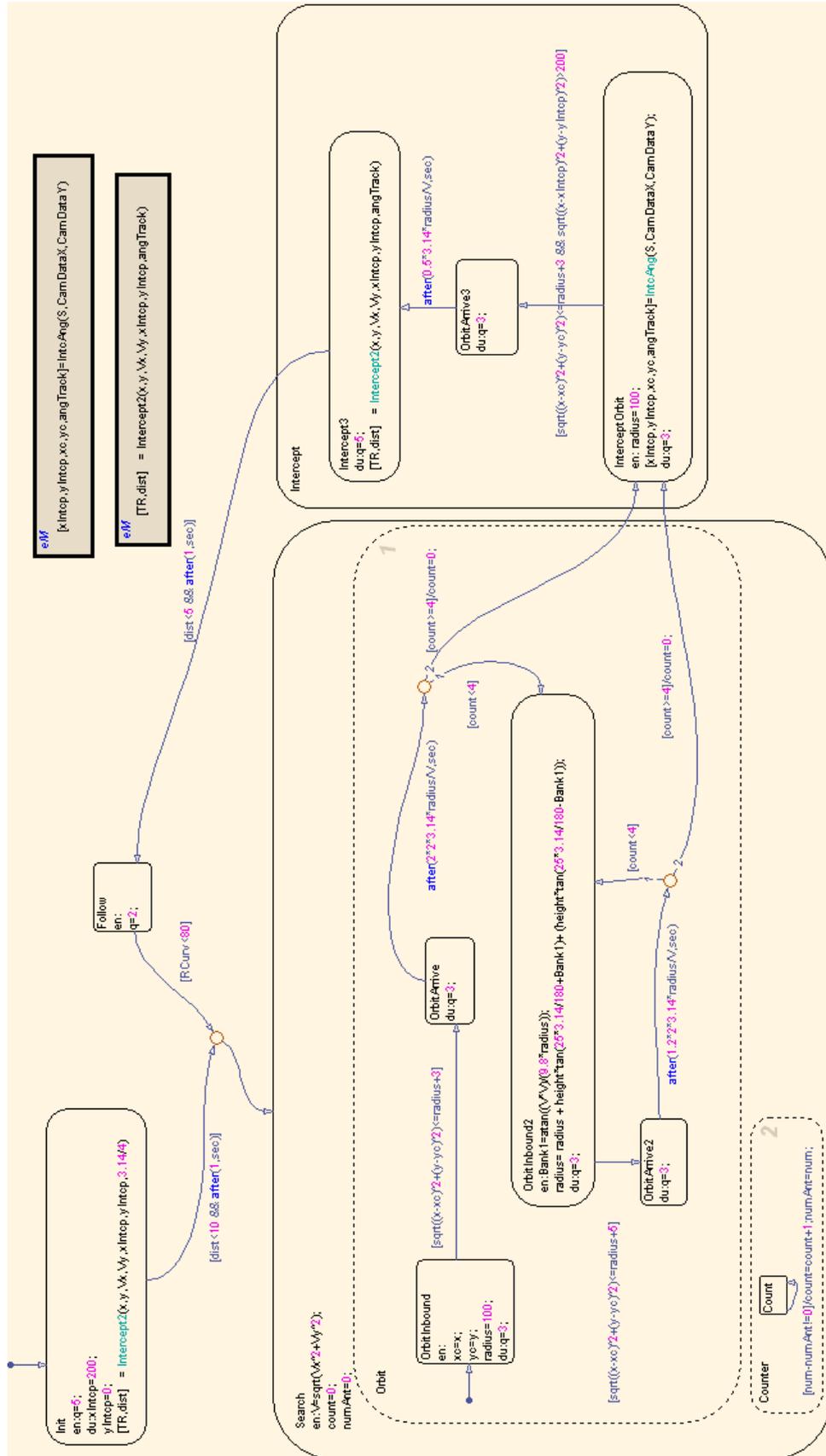
O algoritmo de aproximação de pontos desenvolvido necessita de ser melhorado de forma a tratar o problema quando surgem dois ou mais pontos com coordenadas em  $x$  iguais, no referencial inercial. Uma hipótese de estudo será considerar um referencial que se move ao longo do caminho.

Estudar alternativas para manter a estrutura no campo de visão da câmara em curvas não viáveis. A variação da altitude do UAV é uma das hipóteses a considerar, pois quanto mais alto, maior o campo de visão.

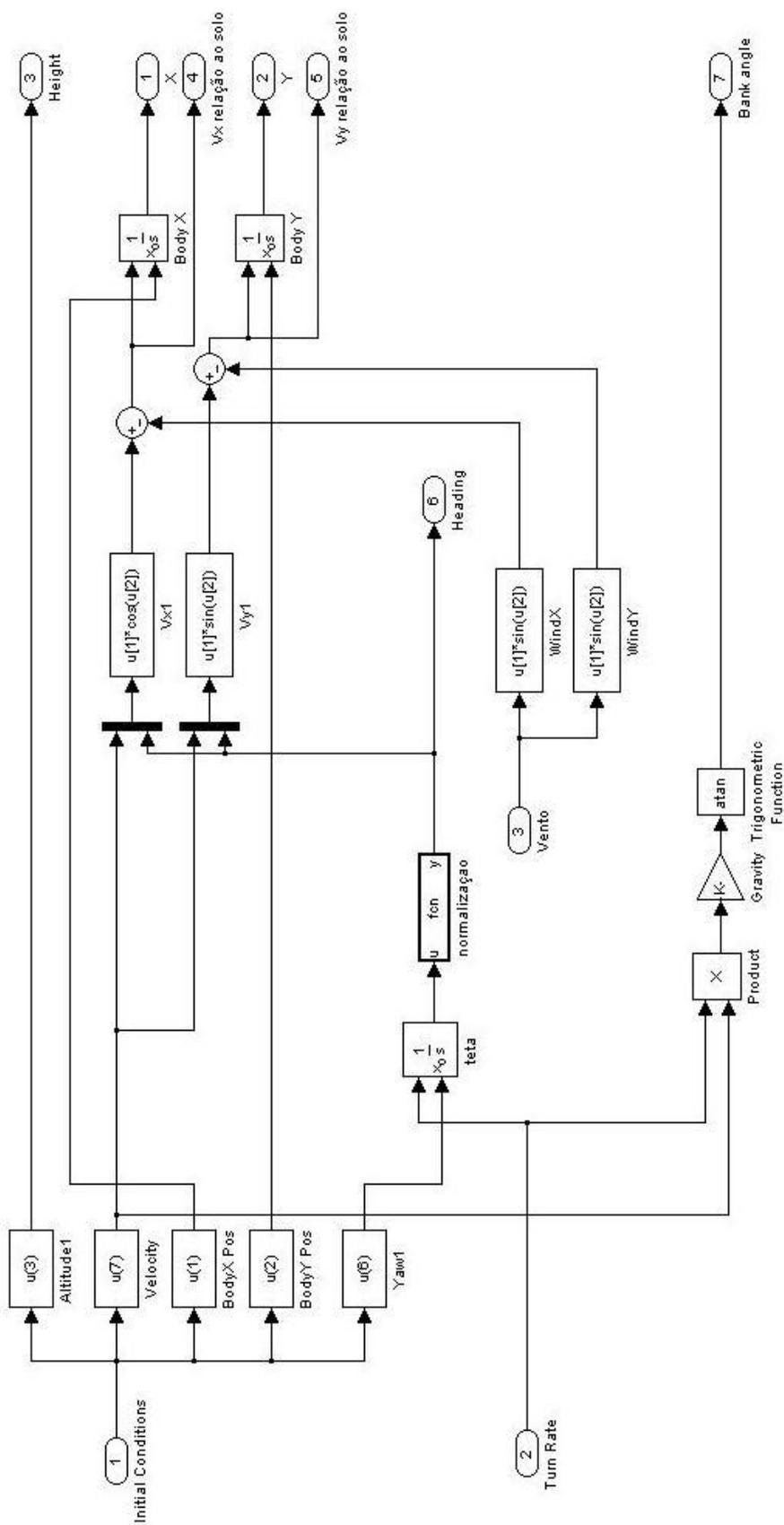
# Anexo A Diagrama de blocos geral do Simulink



## Anexo B Diagrama de estados do controlador híbrido em Stateflow



# Anexo C Modelo do UAV





## Referências

- [1] Almeida, P., R. Bencatel, G. Gonçalves, J.B.d. Sousa, and C. Ruetz. "Experimental results on Command and Control of Unmanned Air Vehicle Systems". in *Proceedings of the IFAC Symposium on Intelligent Autonomous Vehicles 2007*. Toulouse, France.
- [2] Slotine, J.J.E. and W. Li, "Applied Nonlinear Control". 1991, Prentice Hall: New Jersey.
- [3] Hespanha, J.P. "Hybrid and Switched Systems". 2005; Origem: <http://www.ece.ucsb.edu/~hespanha/ece229/>.
- [4] Žak, S.H., "Tools for Analysis of Nonlinear Systems: Lyapunov's Methods". 2008, School of Electrical and Computer Engineering.
- [5] Girard, A., "Control of Nonlinear Dynamic Systems".
- [6] "EML5311 Lyapunov Stability & Robust Control Design". Origem: [http://pegasus.cc.ucf.edu/~cham/eml5311/course\\_lecture/course\\_lecture8.PDF](http://pegasus.cc.ucf.edu/~cham/eml5311/course_lecture/course_lecture8.PDF).
- [7] Mooncheol, W. and J.K. Hedrick. "Discrete time adaptive sliding control of nonlinear systems without matching condition". in *Proceedings of the American Control Conference*. 1995.
- [8] Hedrick, J.K. and P.P. Yip, "Multiple Sliding Surface Control: Theory and Application". *Journal of Dynamic Systems, Measurement, and Control*, 2000. 122(4): p. 586-593.
- [9] Claire, T., "Lecture Notes: Hybrid Systems: Modeling, Analysis, and Control". 2005: Departamento de Aeronáutica e Astronáutica, Universidade de Stanford.
- [10] Spong, M.W., S. Hutchinson, and M. Vidyasagar, "Robot Modeling and Control". 2006: John Wiley & Sons.
- [11] Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, "Numerical Recipes The Art of Scientific Computing". Third ed. 2007: Cambridge University Press.
- [12] Mathews, J.H. and K.D. Fink, "Numerical Methods Using Matlab". Third Edition ed. 1999: Prentice Hall.
- [13] Anderson, E.P., R.W. Beard, and T.W. McLain, "Real-time dynamic trajectory smoothing for unmanned air vehicles". *Control Systems Technology, IEEE Transactions on*, 2005. 13(3): p. 471-477.

- [14] Niculescu, M., "*Lateral track control law for aerosonde UAV*". Proceedings of the 39th AIAA Aerospace Sciences Meeting and Exhibit, 2001.
- [15] Rathinam, S., et al. "*Autonomous Searching and Tracking of a River using an UAV*". in *American Control Conference, 2007. ACC '07.* 2007.
- [16] Hedrick, J.K., J. Lee, R. Huang, A. Vaughn, and X. Xiao. "*Strategies of Path-Planning for a UAV to Track a Ground Vehicle*". in *Proceedings of the AINS Conference.* 2003.
- [17] Rathinam, S., K. Zu, A. Soghikian, and R. Sengupta. "*Vision Based Following of Locally Linear Structures using an Unmanned Aerial Vehicle*". in *44th IEEE Conference on Decision and Control.* 2005.
- [18] Nelson, D.R., D.B. Barber, T.W. McLain, and R.W. Beard, "*Vector Field Path Following for Miniature Air Vehicles*". IEEE Transactions on Robotics, 2007. **23**(3): p. 519-529.
- [19] Nelson, D.R., D.B. Barber, T.W. McLain, and R.W. Beard. "*Vector field path following for small unmanned air vehicles*". in *American Control Conference.* 2006.
- [20] Corradini, M.L. and G. Orlando, "*Robust tracking control of mobile robots in the presence of uncertainties in the dynamical model*". Journal of Robotic Systems, 2001. **18**(6): p. 317-323.
- [21] Girard, A.R., A.S. Howell, and J.K. Hedrick. "*Border patrol and surveillance missions using multiple unmanned air vehicles*". in *43rd Proceedings IEEE Conference on Decision and Control, 2004. CDC.* 2004.
- [22] Rysdyk, R., "*UAV Path following for constant line-of-sight*". University of Washington, Seattle.
- [23] Park, S., J. Deyst, and J.P. How, "*A New Nonlinear Guidance Logic for Trajectory Tracking*". Proceedings AIAA Guidance, Navigation Control Conference, 2004(paper no. AIAA-2004-4900).
- [24] Azinheira, J.R., E. Carneiro de Paiva, J.G. Ramos, Jr., and S.S. Beuno, "*Mission path following for an autonomous unmanned airship*". IEEE International Conference on Robotics & Automation San Francisco, 2000. **2**: p. 1269-1275.
- [25] Ye, Z., P. Bhattacharya, H. Mohamadian, H. Majlesein, and Y. Yongmao. "*Equational dynamic modeling and adaptive control of UAV*". in *IEEE International Conference on System of Systems Engineering Los Angeles.* 2006.
- [26] Wei, R. and R.W. Beard, "*Trajectory tracking for unmanned air vehicles with velocity and heading rate constraints*". IEEE Transactions on Control Systems Technology, 2004. **12**(5): p. 706-716.
- [27] Cook, M.V., "*Flight Dynamics Principles*". Second Edition ed. 2007: Elsevier.
- [28] Barber, D.B., J.D. Redding, T.W. McLain, R.W. Beard, and C.N. Taylor, "*Vision-based Target Geo-location using a Fixed-wing Miniature Air Vehicle*". Journal of Intelligent and Robotic Systems, 2006. **47**(4): p. 361-382.
- [29] Egbert, J. and R.W. Beard. "*Low Altitude Road Following Constraints Using Strap-down EO Cameras on Miniature Air Vehicles*". in *Proceedings of the American Control Conference, 2007. ACC '07.* 2007.

- [30] Liao, X.H., S. Zhao, and Y.D. Song. "*Chattering-free variable structure control with application to flight vehicles*". in *System Theory, 2006. SSST '06. Proceedings of the Thirty-Eighth Southeastern Symposium on*. 2006.