

# WINDIRIGÍVEL – Controlo de um Dirigível Autónomo Robotizado

Paulo Brígida, A. Paulo Coimbra, Manuel Crisóstomo

Instituto de Sistemas e Robótica, Departamento de Engenharia Electrotécnica e de Computadores  
Universidade de Coimbra, Polo 2, 3030-290 Coimbra, Portugal

paulobrigida@yahoo.com.br, acoimbra@deec.uc.pt, mcris@isr.uc.pt

**Resumo.** Este trabalho consistiu na criação de dois softwares. O primeiro, de nome WinDirigível, controla e gere um dirigível e os sensores a ele acoplados, através de uma ligação RS232 wireless. O segundo software, de nome SimuladorDEEC, tem por função simular o comportamento desse dirigível e os sensores a ele acoplados, de forma a testar a transmissão e recepção de comandos e dados por ligação RS232 bem como o algoritmo de controlo e planeamento de trajectórias do WinDirigível.

Os resultados obtidos limitam-se, para já, aos conseguidos nos testes com o simulador, pois o hardware a ser embebido no dirigível ainda não se encontra concluído. Dentro das limitações do algoritmo do comportamento físico de um dirigível integrado no simulador, o qual se encontra ainda pouco desenvolvido, e do algoritmo de controlo usado para planeamento da trajectória, os resultados conseguidos são bastante satisfatórios.

O trabalho já realizado incidiu maioritariamente na construção do interface gráfico e de toda a estrutura de programação que envolve os dois softwares. No futuro, espera-se com este software poder controlar e simular outros tipos de UAV.

## Palavras-chave

UAV, dirigível, simulador, controlo, robô.

## 1. Introdução

De entre as “máquinas voadoras”, o dirigível oferece vantagens, se bem que também algumas desvantagens, comparativamente aos diversos tipos de helicópteros, aviões, planadores, etc.

A capacidade de um dirigível se manter mais ou menos estável a pairar no ar, bem como o controlo suave dos seus movimentos, passando pela diminuta manutenção exigida, tornam-no num veículo que tem suscitado grande interesse por parte da comunidade científica e do grande público [1 a 6].

Pretendeu-se com este projecto criar um software que permitisse o controlo de uma aeronave de escala reduzida

e a gestão dos dados dos sensores a bordo, tendo um interface com o utilizador “user-friendly”, graficamente agradável e atraente.

A figura 1 exemplifica a arquitectura do sistema desenvolvido, aplicado a um dirigível de hélio. O software desenvolvido foi vocacionado para um pequeno dirigível, mas pode ser facilmente adaptado para qualquer outro tipo de veículo móvel.

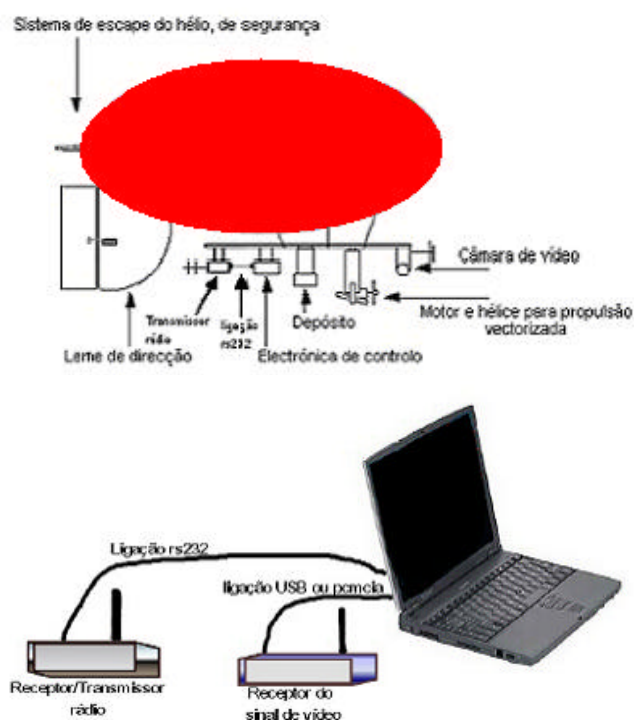


Fig. 1. Arquitectura do sistema desenvolvido

Como aplicação de apoio ao software principal, foi criado um outro programa (SimuladorDEEC) para simular o comportamento do dirigível com vento de direcção e intensidade aleatórias ou constantes, o qual trabalha noutro PC, estando conectado por RS232 ao PC que processa o WinDirigível, de modo a simular e testar

a transmissão de comandos e a recepção de dados dos sensores.

## 2. WinDirigível

Este é o software principal de todo o projecto. É ele que processa as informações transmitidas pela panóplia de equipamento sensorial montado no dirigível, transmitindo os sinais de comando para o dirigível.

A linguagem de programação utilizada foi o C++ [7] tendo sido desenvolvidas cerca de 70 000 linhas de código. Foi ainda utilizado o pacote gráfico OpenGL.

As bases de dados implementadas usam a tecnologia Paradox versão 7.0. Podiam ter sido usadas outras tecnologias alternativas para a composição das tabelas hospedeiras das diversas informações necessárias, desde mysql, dbase, interbase, etc. No entanto a escolha recaiu sobre o Paradox por esta tecnologia ter um bom rácio entre a flexibilidade do tipo de dados a guardar e a robustez dos drivers ODBC de ligação entre a aplicação e a base de dados. É preciso referir que o Paradox não é conhecido pela sua rapidez de acesso de leitura e escrita nas tabelas de dados. Alguma lentidão, em certos momentos de execução do programa, pode dever-se à escrita ou leitura nalguma tabela, nomeadamente na abertura da aplicação em que é executado um processo de inicialização dos valores de variáveis resguardados numa tabela (configuracao.db).

Em computadores rápidos, com discos usando tecnologias de BUS recentes, só em condições extremas se deverá notar algum atraso. O registo dos backups dos dados relativos ao posicionamento do dirigível no seguimento de uma missão ou em modo manual, bem como a salvaguarda da constituição de uma missão, é realizada em ficheiros, precisamente para evitar o congestionamento das leituras que se fizessem no futuro, aumentando a performance do sistema, e melhorando a compatibilidade de outras possíveis aplicações que se possam vir a desenvolver e que pretendessem fazer uso desses dados. A leitura independente que os ficheiros oferecem facilita também que os dados possam ser facilmente abertos por outro tipo de aplicações com vista a serem estudados ou exibidos, ou de poderem dar origem a situações de aprendizagem de redes neuronais ou de lógica difusa em programas como o MatLab.

### A. Características

Listamos a seguir as características principais do software WinDirigível:

- Aquisição, por porta série RS232, dos dados transmitidos pelo simulador ou pelo dirigível real (altitude, altura, GPS e bússola electrónica).



Fig. 2. Ecrã principal do WinDirigível

- Transmissão, por porta série RS232, de comandos para controlar os lemes de altitude e de direcção bem como a aceleração dos motores.
- Visualização gráfica ou numérica, consoante as grandezas, dos valores dos sensores, dos valores calculados da velocidade actual, da distância à estação de base e dos ângulos dos lemes em relação à posição neutra (figura 2).
- Possibilidade de controlo manual através da interface gráfica.
- Output gráfico do erro da trajectória seguida pelo dirigível, da diferença entre a velocidade pretendida e a velocidade real, e da diferença entre a altitude pretendida e a altitude real do dirigível.
- Localização do dirigível num mapa ou fotografia aérea, escolhidos pelo utilizador e lido de um ficheiro ou da base de dados. Representa-se também a sua orientação, bem como a rota percorrida, as etapas constituintes da missão e a possibilidade de se saber a distância entre quaisquer dois pontos localizados no mapa, através do uso do cursor do rato. A rota percorrida pelo dirigível pode ser apresentada por um gradiente de cores representando a percentagem de caminho percorrido até ao término da missão (ver figura 3).

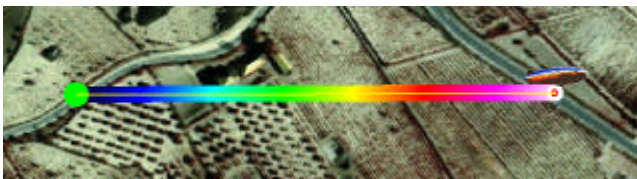


Fig. 3. Localização do dirigível no mapa, com indicação da percentagem da rota percorrida, codificada com cores

- Criação dos projectos de missão, passo a passo, com a possibilidade de serem salvaguardados em ficheiro e carregados no sistema para serem executados. Na figura 4 apresenta-se um dos ecrãs do “wizard mission”.



Fig. 4. Último ecrã de criação e planeamento de missão

- Salvaguarda em ficheiro dos dados dos sensores obtidos no cumprimento de uma determinada missão, com a possibilidade de serem carregados no sistema para uma posterior visualização e análise, usando para isso a funcionalidade “mission player”, que permite avançar, passo a passo ou a velocidade variável, o decorrer de uma missão anteriormente guardada.
- Opções configuráveis como o raio de precisão dos pontos de passagem.
- Opções de ida directa para o final da missão ou para um local de emergência definido aquando a criação da missão.

### B. Os Gráficos das Diferenças

O módulo dos gráficos das diferenças (ou erros de trajectória, de velocidade e de altitude) tem uma grande importância, pois serão eles que comprovarão com maior exactidão os resultados do controlo imposto ao dirigível numa missão, pelo WinDirigível.

Qualquer um dos três tipos de gráficos permite que se altere a profundidade do efeito de três dimensões, bem como aumentar ou diminuir o efeito de rotação e translação do seu eixo, através da interacção com o menu disposto na parte de baixo do módulo.

### C. O Erro da Trajectória

O valor do erro de trajectória definiu-se como sendo a distância do ponto onde o dirigível se encontra ao segmento de recta da etapa corrente (ver figura 5).

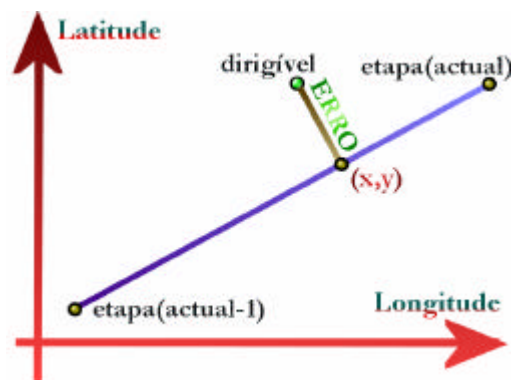


Fig. 5. Definição do erro de trajectória

No gráfico da figura 6 podemos ver a evolução do erro de trajectória ao longo de uma missão.



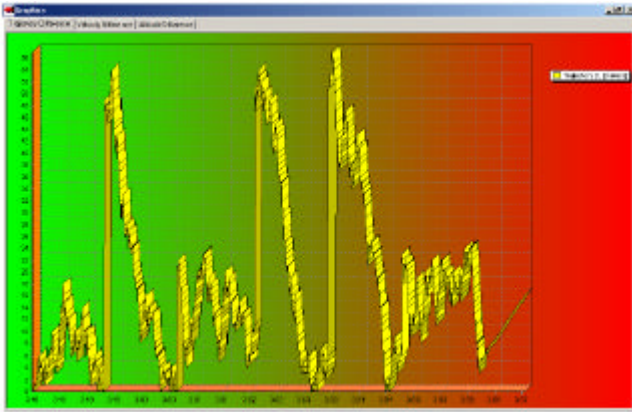


Fig. 6. Evolução do erro de trajectória

#### D. O Erro da Velocidade

O erro da velocidade é definido com o sendo a diferença entre a velocidade pretendida, fixada pelo utilizador, e a velocidade real que o dirigível tem em determinado momento. Estas velocidades são definidas relativamente ao solo.

#### E. O Erro da Altitude

De modo semelhante à obtenção do erro da velocidade atrás definido, o erro da altitude é calculado pela diferença entre a altitude pretendida pelo utilizador e a altitude que o dirigível tem em determinado instante.

#### F. Sistema de Controlo

Após a criação de um projecto de missão ou do seu carregamento através de um ficheiro, o WinDirigível está pronto a controlar o voo do dirigível real ou virtual. No modo de simulação, o processamento é muito semelhante ao que é efectuado no modo RS232, substituindo-se os dados da comunicação via rádio por valores simulados.

No modo RS232, os dados dos sensores são adquiridos através de recepção via RS232 (wireless), advindo estes de um simulador a correr noutra computador ou do dirigível real. Na figura 7 mostra-se o fluxograma resumido do algoritmo de controlo numa situação de execução de uma missão no modo RS232.

#### G. Controlo da Direcção, Velocidade e Altitude

Para cada uma destas grandezas foi implementado um controlador do tipo proporcional derivativo (PD). Este controlador é muito simples e foi apenas implementado para se realizarem testes preliminares de verificação dos softwares.

No futuro será inserido um controlo por lógica difusa, por redes neuronais ou, ainda, neuro-difuso, o que trará, seguramente, melhores resultados. Uma outra possibilidade será o uso de filtros de Kalman.

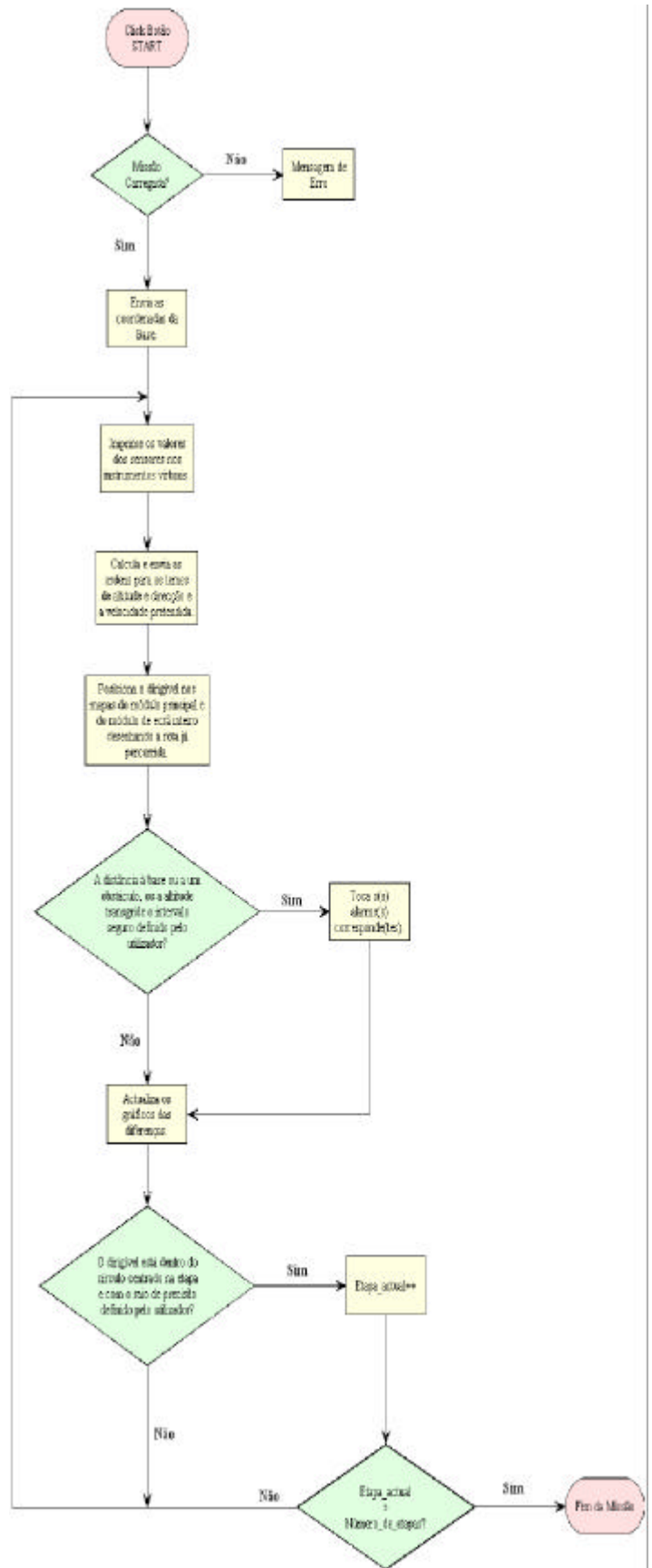


Fig. 7. Fluxograma simplificado do controlo do dirigível numa missão

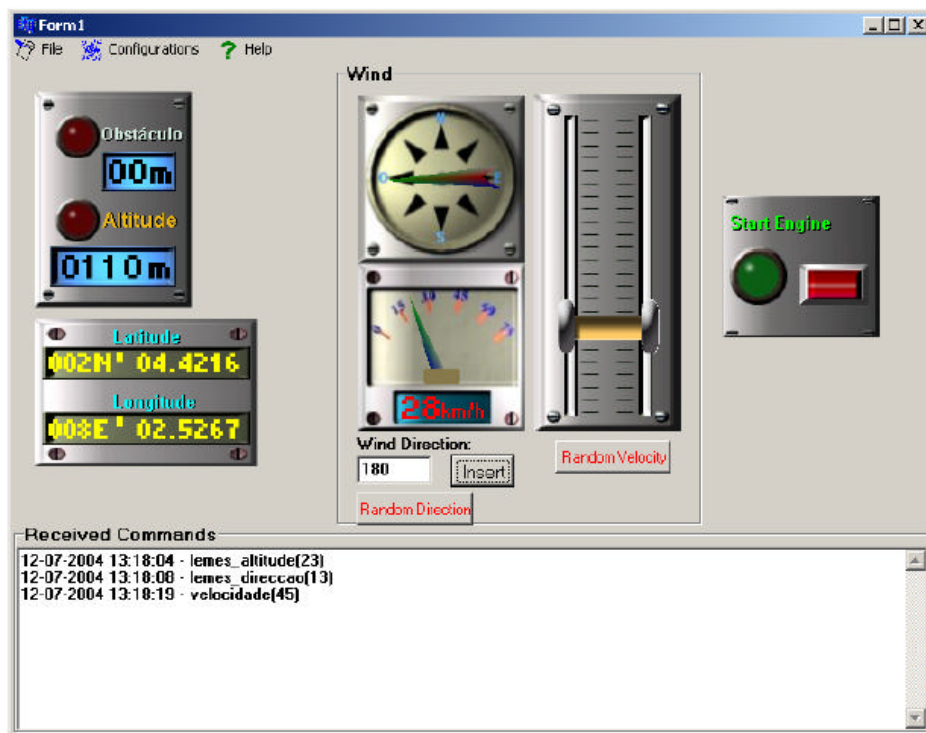


Fig. 8 – Ecrã principal do SimuladorDEEC

### 3. SimuladorDEEC

Como aplicação de apoio ao software central WinDirigível, foi criado este simulador, com o objectivo de permitir simular o comportamento de um dirigível perante determinados factores meteorológicos e com uma aproximação razoável a um modelo físico real (ver ecrã principal na figura 8). Além da capacidade de simulação de um dirigível, será possível, no futuro, simular também outro tipo de veículos, bastando adicionar o modelo matemático do comportamento físico desses veículos e que se inclua essa escolha no menu das opções do programa.

#### A. Características

Listamos a seguir as características principais do software SimuladorDEEC.

- Modo de simulação do comportamento de um dirigível através da comunicação com um outro PC que esteja a correr o WinDirigível, por porta série RS232, de modo a simular o envio de dados dos sensores que o dirigível real iria fazer se executasse os comandos de controlo enviados pelo WinDirigível.
- Incorpora um modelo matemático simples de simulação do comportamento físico de um dirigível.
- Opção de se seleccionar manualmente diferentes situações de intensidade e direcção do vento, havendo também a possibilidade de o vento ter uma direcção e velocidade aleatórias.

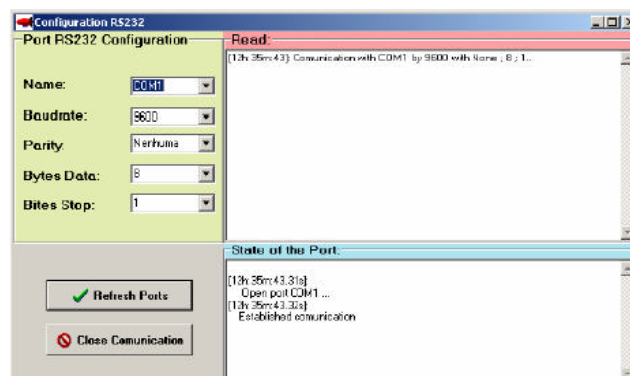


Fig. 9. Ecrã de configuração da porta série.

- Opção de se alterar a configuração da porta série RS232 (ver figura 9).

#### B. Simulação do Vento

Como foi referido, é simulada a interacção do vento com o dirigível. O resultado desta interacção é considerado simplesmente como o vector da velocidade do vento que vai ser somado ao vector da velocidade correspondente à impulsão do motor (ver figura 10).

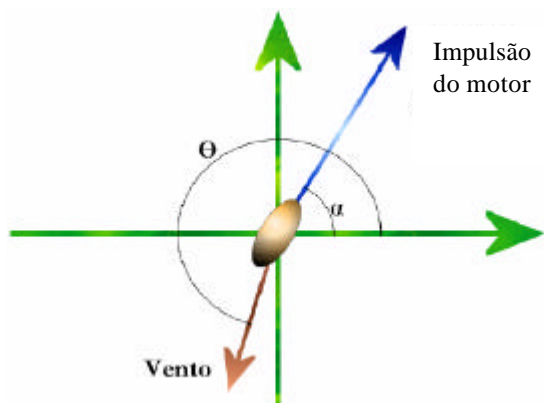


Fig. 10. A acção do vento é somada vectorialmente à impulsão do motor.

Assim, as equações correspondentes à velocidade resultante são:

$$V_x = |\text{Impulsão Motor}| \cdot \cos(\alpha) + |\text{Vento}| \cdot \cos(\theta) \quad (1)$$

$$V_y = |\text{Impulsão Motor}| \cdot \sin(\alpha) + |\text{Vento}| \cdot \sin(\theta) \quad (2)$$

$$|\text{Velocidade Resultante}| = \sqrt{V_x^2 + V_y^2} \quad (3)$$

#### 4. Resultados Obtidos

Optou-se por correr as simulações com um período de actualização de dados de 500 ms, ou seja, com uma frequência de 2 Hz, pois embora se preveja que o emissor de rádio a bordo do dirigível real tenha uma frequência de emissão de 10 Hz, correr as simulações com o período de actualização equivalente de 100 ms tornar-se-ia muito pesado no esforço computacional necessário à simulação em computadores uniprocessador actuais, podendo haver problemas de “entupimento” de informação. A consequência disto é que, principalmente nas situações em que o dirigível se desloque a grandes velocidades, a sua reacção vai ser um pouco mais demorada do que a que podia eventualmente ser, mas numa situação de simulação contínua, a diferença não se deverá notar. No entanto, mesmo com um período de actualização de dados de 500 ms, o esforço de processamento é significativo. No caso de se pretender correr uma simulação num PC contendo um processador com frequência interna de CPU abaixo dos 2 GHz, recomenda-se que se proceda ao aumento do período de actualização.

##### A. Exemplo de simulação de missão

Neste projecto de missão, o dirigível terá de cumprir seis etapas, dispostas nos seis vértices de um hexágono, estando a estação de base num dos vértices. O local de

emergência situa-se no centro do hexágono. Mostra-se, na figura 11, a trajectória resultante da missão simulada, com vento com velocidade e direcção aleatórias.

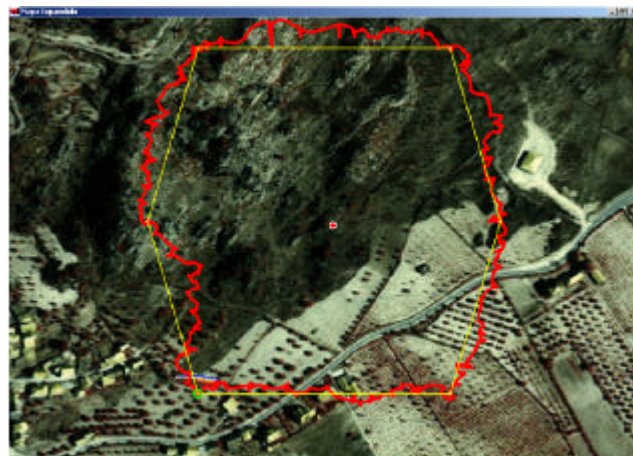


Fig. 11. Trajectória obtida com vento com velocidade e direcção aleatórias

#### 5. Conclusões

Implementaram-se dois programas para fazer o controlo, a monitorização e a simulação de um dirigível autónomo não tripulado.

Foi também desenvolvido o interface com o utilizador que permite definir as missões de um modo simples e interactivo, para além de visualizar a trajectória em tempo útil, sobre o mapa digitalizado.

#### Referências

- [1] S. Lacroix. *Toward autonomous airships: research and developments at LAAS/CNRS*. In 3rd International Airship Convention and Exhibition. Friedrichshafen (Germany), 2000.
- [2] S. Lacroix, IK. Jung, P. Soueres, E. Hygounenc and J-P. Berry. *The autonomous blimp project of LAAS/CNRS: Current status and research challenges*. In 8th International Symposium on Experimental Robotics. Sant'Angelo d'Ischia (Italy), 2002.
- [3] <http://www.rudnei.cunha.nom.br/>, Julho 2004.
- [4] [www.stwing.org/blimp/](http://www.stwing.org/blimp/), Julho 2004.
- [5] [users.aber.ac.uk/dpb/aerobots.html](http://users.aber.ac.uk/dpb/aerobots.html), Julho 2004.
- [6] [www.sp.edu.sg/schools/eee/r\\_d/atic/airship.htm](http://www.sp.edu.sg/schools/eee/r_d/atic/airship.htm), Julho 2004.
- [7] Borland C++ version 6.0 Developer's Guide, 2004.