

# Aplicação da *OpenSteer* na Resolução de Conflitos de Tráfego Aéreo Simulado

Hudson Peçanha Murad, Carlos Henrique Costa Ribeiro

Instituto Tecnológico de Aeronáutica, Programa de Pós-graduação em Eng. Eletrônica e Computação. P. Mal. Eduardo Gomes 50 – São José dos Campos - SP

**Resumo** — As taxas de crescimento do tráfego aéreo mundial, aliadas à implantação do novo sistema de gerência e controle do espaço aéreo nos países signatários da ICAO (CNS/ATM) permitem concluir que um número crescente de aeronaves voarão cada vez mais próximas umas das outras. Soluções que ofereçam algoritmos para resolução de conflitos de tráfego aéreo em ambiente simulado podem contribuir, portanto, para o aumento da segurança das operações quando adaptadas para futuro emprego tanto em sistemas de controle de tráfego aéreo quanto em aeronaves. Este artigo apresenta uma proposta de aplicação de uma adaptação em Java da *OpenSteer*, uma biblioteca livre em C++, visando a resolução de conflitos entre aeronaves em ambiente simulado.

**Palavras-chaves** — Tráfego Aéreo, Comando e Controle, CNS/ATM, *OpenSteer*.

## I. INTRODUÇÃO

O jornal Folha de São Paulo, em outubro de 2007, publicou que o crescimento esperado para o tráfego aéreo mundial entre 2007 e 2011 seria de 5% em média [1].

O Comando da Aeronáutica forneceu à imprensa informações contextualizadas e indicadores nacionais e internacionais que, entre outros, apontavam os índices de incidentes relatados por pilotos, controladores ou ambos [2].

Atualmente, aeronaves que sobrevoam o espaço aéreo brasileiro controlado têm a separação vertical e horizontal definidas segundo normas do DECEA – Departamento de Controle do Espaço Aéreo – de acordo com a categoria (classe) do espaço, a posição relativa aos auxílios à navegação (VOR, NDB) e regras de voo (visual ou instrumento) às quais estejam submetidas. [9]

O Brasil, como signatário da ICAO – Organização de Aviação Civil Internacional – está comprometido com a instalação de um novo sistema de controle e gerenciamento de fluxo e de tráfego aéreo, CNS/ATM – sigla em inglês para Comunicação, Navegação e Vigilância / Gerência de Tráfego Aéreo – cujos objetivos são norteados pelo aumento da capacidade de controle frente à crescente demanda por serviços aeronáuticos, tendo como premissa a segurança das operações, servindo-se de novas tecnologias, tais como o GPS (*Global Positioning System*) e o ADS (*Automatic Dependent Surveillance*).

Este novo sistema, previsto para ser implantado em várias fases, requer diminuição nos mínimos de separação entre as aeronaves, o que pode ser obtido em virtude da maior

precisão das informações relativas à posição das aeronaves no tempo e no espaço. [3]

Considerando-se o aumento do volume de tráfego aéreo e as demandas por diminuição da separação entre as aeronaves, o que de fato já está sendo praticado em condições especiais por alguns aviões num modo de operação conhecido como RVSM – Reduced Vertical Separation Minimum) [4], apesar dos atuais índices de incidentes aeronáuticos, é razoável esperar por sugestões que contribuam para o aumento da segurança relativa à operação de um número crescente de aeronaves voando cada vez mais próximas, suportadas por sistemas de navegação de alta precisão.

Neste sentido, um grupo de pesquisadores vinculados ao projeto iFly, da Comissão Europeia (CE), ligado ao “6th Framework Programme FP6-2005-Aero-4” [5], apresentou no relatório intitulado Estudo Comparativo de Métodos de Resolução de Conflito uma classificação de três categorias para os métodos de resolução de conflitos: Longo, Médio e Curto Prazo [6].

O presente artigo considera apenas conflitos de curto prazo, e propõe que sejam atribuídos comportamentos pré-definidos a cada aeronave simulada, de modo que, baseadas nestes comportamentos, realizem manobras de evasão para resolução dos conflitos de curto prazo, quando as rotas de voo vierem a se interceptar num espaço de tempo de até 6 minutos de voo (pontos de colisão no tempo e no espaço).

Os comportamentos supracitados sofreram adaptações daqueles oferecidos pela biblioteca livre em C++, denominada *OpenSteer* [7], cuja modelagem é discutida em [8].

Inicialmente, vislumbra-se a possibilidade de emprego em sistemas de software embarcados nas aeronaves e/ou nos sistemas de controle de tráfego aéreo localizados em terra.

Os constantes das aeronaves teriam o propósito de auxiliar o processo de tomada de decisão pelos pilotos voando em espaço aéreo não controlado. Aqueles implantados em terra serviriam para auxiliar a tarefa dos controladores de voo nas diretivas empregadas em espaços aéreos controlados.

Este artigo está dividido em quatro seções. A primeira seção procura identificar razões para a proposta baseando-se em fatos e estimativas veiculadas na mídia e nas informações existentes quer em publicações oficiais, quer oriundas de pesquisas. A segunda seção descreve, de modo sucinto, considerações sobre o ambiente de tráfego aéreo, especificamente os espaços aéreos (atual e futuro) do Brasil, levando-se em conta a implantação do sistema CNS/ATM suportado por tecnologias existentes e em desenvolvimento.

Cita também resumidamente observações encontradas na coletânea de métodos de resolução de conflitos, resultantes de uma pesquisa realizada pelo projeto *iFly* da CE. Termina por revelar a motivação para este artigo baseada na descoberta de similaridades entre os comportamentos apresentados por agentes que se movimentam (presentes nos jogos computadorizados) e aqueles esperados das aeronaves que precisam realizar manobras evasivas para solucionar conflitos de tráfego de curto prazo. A terceira seção aborda inicialmente o modelo de um veículo simples. Cita alguns tipos principais de comportamentos que são criados a partir do modelo conceitual do veículo para os quais existe suporte nos algoritmos implementados na biblioteca *OpenSteer*. Distingue então aqueles comportamentos que serão inseridos no Simulador de Visualização de Controle de Tráfego – *VisuATCSim*, seguindo-se uma breve apresentação do ambiente de simulação já implementado no qual espera-se gerar os efeitos desejados e obter resultados que propiciem análises. A quarta e última seção oferece uma síntese das considerações apresentadas.

## II. O CENÁRIO

Atualmente, no Brasil, a separação das aeronaves, quer voando sob as regras de voo visual (VFR- *Visual Flight Rules*), quer por instrumentos (IFR – *Instrument Flight Rules*), são definidas pelo Comando da Aeronáutica em consonância com as orientações da OACI.

Os limites de separação horizontal e vertical das aeronaves entre si e destas em relação a obstáculos, a serem observados pelos pilotos e controladores, são definidos de acordo com as classes do espaço aéreo, posição relativa aos auxílios rádio à navegação aérea e regras de voo. [9]

De maneira bastante sucinta, pode-se dividir o processo de separação de aeronaves de duas formas distintas:

a) aeronaves voando em espaços aéreos controlados são separadas por profissionais de controle do espaço aéreo que estabelecem comunicação bilateral com os pilotos e os orientam sobre manobras a serem executadas tomando por base as informações disponibilizadas em sistemas de controle (radares e sistemas de tratamento e visualização de dados) e normas em vigor.

b) aeronaves voando em espaços aéreos não controlados são separadas pelos próprios pilotos baseando-se nas normas em vigor e nas observações contínuas do campo visual exterior à cabine e/ou através de comunicação bilateral oral entre aeronaves no mesmo setor.

Para propiciar maior segurança nas operações, além dos sistemas de controle baseados em terra (radares, auxílios rádio à navegação e estações de controle), existem atualmente sistemas embarcados em aeronaves, independentes de meios terrestres e conhecidos como ACAS – *Airborn Collision Avoidance Systems* – que fornecem alarmes sonoros e visuais ao piloto nas situações que apresentem potencial de risco de colisão aeronave/aeronave e aeronave/solo. [10]

Além da segurança, crescem as necessidades de movimentos de aeronaves nos aeroportos (pousos e decolagens) e de otimização do uso do espaço aéreo (aumento da capacidade) para atender à demanda imposta pelo transporte aéreo mundial, com aumentos da ordem de 5% ao ano.

Para enfrentar esses desafios, a ICAO e alguns países membros realizaram acordos visando o desenvolvimento e

implantação do CNS/ATM. No Brasil, ações estão sendo desenvolvidas de acordo com um programa de transição, desde 2005, com prazo estimado de conclusão para 2020 [3, 11].

O sistema supracitado envolve o emprego integrado de várias tecnologias tais como a de posicionamento global por satélite (GPS), a de vigilância dependente automática (ADS) e a de comunicações via enlace de dados (CPDLC – *Controller Pilot Data Link Communications*). As últimas se utilizam dos meios existentes de comunicação por alta frequência baseado ondas de rádio (VHF, por exemplo) e por meio de satélites, para substituir/complementar o contato bilateral oral entre controlador e piloto. [12]

Uma das vantagens pretendidas pelo novo sistema é o aumento da cobertura dos espaços aéreos não controlados em função dos altos custos e das características intrínsecas dos radares de rota.

As tecnologias citadas (GPS e CPDLC) propiciarão, no futuro, o desenvolvimento de soluções automatizadas para “a melhor utilização do espaço aéreo” que, “por sua vez, tem como um de seus princípios a flexibilização do uso do espaço aéreo, onde uma estrutura de rotas fixas seria exceção, em vez de regra. A intervenção tática do controlador de tráfego aéreo para fins de manter a separação entre aeronaves deveria ser desnecessária, pois qualquer intervenção dessa natureza implica em alterar o perfil ótimo de voo, com seus custos correspondentes.” (*apud* PCA 351-3, 2006, p.33).

O grupo de pesquisa *iFly* da Comissão Européia, que trabalha no projeto de segurança, complexidade e responsabilidade baseada em design e validação de gerenciamento de tráfego aéreo automatizado em larga escala, vem estudando problemas e soluções que forneçam desempenho comprovado sobre métodos que propiciem o voo livre e autônomo de aeronaves, ou seja, aqueles que requeiram o mínimo de interferência de um controlador de tráfego aéreo. Neste sentido, confeccionou um relatório[6] no qual apresentou os resultados de uma pesquisa conduzida para identificar os diferentes métodos de resolução automática de conflitos. A coletânea reuniu os métodos de abordagem centralizada e descentralizada que foram analisados e comparados em termos de suas capacidades, limitações e complementaridades, sob a perspectiva da resolução de conflitos para aeronaves autônomas.

O trabalho supracitado classificou as possibilidades de conflitos em três categorias:

Longo Prazo – aqueles associados ao gerenciamento de fluxo ou de trajetória de tráfego aéreo, ou seja, os que consideram informações contidas nos planos de voo, capacidades de aeroportos, efeitos meteorológicos entre outros.

Médio Prazo – os que consideram as informações atualizadas sobre a trajetória de voo pretendida e que se encontram num “horizonte temporal” de 15-20 minutos.

Curto Prazo – baseados na informação de posição e sua extrapolação no futuro próximo (horizonte temporal de 6-5 minutos para garantir a utilidade da informação e reduzir a quantidade de falsos alarmes).

Sobre esta última categoria, apontou-se a dificuldade de se buscar formas de otimização devido ao esforço computacional requerido e à complexidade envolvida na combinação de mais de duas aeronaves.

Ainda sobre os conflitos de curto prazo, o relatório[6] descreveu duas técnicas empregadas em Robótica

Móvel aplicadas ao contexto: Campos Potenciais Artificiais e Funções de Navegação.

Observando-se a natureza das particularidades envolvidas na resolução de conflitos desta última categoria e buscando-se alternativas de soluções em outras áreas de conhecimento, vislumbrou-se a possibilidade de utilização de conceitos empregados na indústria de jogos computadorizados (baseados em plataformas dedicadas) e na indústria do cinema (filmes e desenhos com elevado grau de aplicação da computação gráfica).

Mais especificamente, adotou-se como fonte inicial da modelagem alguns algoritmos da biblioteca *OpenSteer*, oferecidos como *open source* de acordo com a licença do MIT (*Massachusetts Institute of Technology*).

Os conceitos utilizados para implementação desta biblioteca foram apresentados em [8].

### III. A APLICAÇÃO

Para Reynolds[8], no universo da animação por computador (cinema) e da mídia interativa (jogos), personagens autônomos são agentes que possuem habilidade para navegar pelo mundo virtual de modo semelhante à vida real.

Estes comportamentos (habilidades) direcionais (*steering behaviors*) são independentes e, quando combinados, podem ser utilizados para se alcançar objetivos mais complexos tais como seguir um determinado caminho desviando-se de obstáculos fixos. Os agentes autônomos podem representar pessoas, animais, microorganismos e veículos (carros, motocicletas e aviões, por exemplo) entre outros tipos de seres que se locomovem para atingir um objetivo, dado um conjunto de ações que provoquem reações.

No contexto da simulação de controle de tráfego aéreo (Air Traffic Controller – ATC) a modelagem de um avião, sem perda de generalidade, seria equivalente a de um simples veículo dotado de massa pontual, que favorece a uma simplicidade computacional bastante conveniente para os objetivos deste trabalho (massa pontual tem velocidade – momento linear, mas não possui momento de inércia – momento rotacional).

#### O Modelo Simplificado do Veículo

A implementação do modelo, segundo Reynolds, seria definida pelas propriedades posição, massa e velocidade, sendo a última modificada pela aplicação de forças.

O pseudocódigo da figura 1 apresenta as propriedades criadas para adequar características do mundo real ao virtual, restrições, tais como a potência do motor, seriam obtidas via parâmetro *max\_force* (força máxima) enquanto que as devidas ao arrasto obtidas por outro parâmetro *max\_speed* (velocidade máxima).

Para completar a modelagem define-se uma orientação, que em conjunto à posição do veículo, formaria um **espaço local de coordenadas alinhado à velocidade** onde o modelo geométrico (do veículo) pudesse ser situado.

```
Simple Vehicle Model:
mass          scalar;
position      vector;
velocity      vector;
max_force     scalar;
max_speed     scalar;
```

Fig. 1. Propriedades do Modelo Simplificado do Veículo.

Para o modelo de um veículo em 3D, os vetores posição e velocidade teriam três componentes, e a orientação um conjunto de três vetores.

No caso da simulação ATC considerando-se apenas manobras na horizontal (2D), os vetores teriam duas componentes e a orientação dois vetores (base 2D) podendo ser representada por um único escalar correspondente ao ângulo da proa.

A evolução física do modelo baseia-se no princípio de integração de Euler. A figura 2 exprime o pseudocódigo das ações ocorridas a cada passo da simulação. A resultante das forças de direcionamento (*steering\_force*), truncada pela força máxima e determinada pelos comportamentos (ações e reações), produziria aceleração (resultante da divisão pela massa, no caso não pontual) que somada à velocidade antiga e depois truncada pela velocidade máxima, produziria nova velocidade, esta responsável por alterar a posição do veículo.

```
steering_force =
truncate(steering_direction, max_force);
acceleration = steering_force / mass;
velocity = truncate(velocity +
acceleration, max_speed);
position = position + velocity;
```

Fig. 2. Procedimentos em cada passo de simulação.

O modelo do veículo mantém seu espaço local de coordenadas alinhado à velocidade através de ajustes incrementais aplicados às propriedades do passo anterior da simulação.

Este espaço local é definido em termo de quatro vetores conforme ilustrado na figura 4: um vetor posição que especifica a origem local e três vetores que formam a base ortogonal (vetores unitários mutuamente ortogonais) relativa ao veículo.

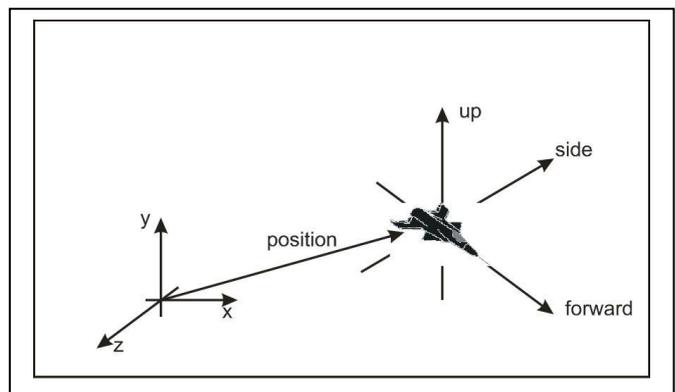


Fig. 4. Espaço Vetorial Local.

Objetivando-se a simplificação da compreensão, os vetores desta base foram nomeados de *forward*, *up* e *side*.

Para que a base se mantivesse alinhada à velocidade do veículo a cada passo da simulação, os vetores da base deveriam passar por um processo de rotação. No entanto, em [8] opta-se por reconstruir o espaço local pela combinação de substituições, aproximações (normalizações) e reortogonalizações (produtos vetoriais – *cross product*) tais quais explicitadas no trecho de pseudocódigo da figura 3.

```
new_forward = normalize (velocity);
approximate_up = normalize (up);
new_side =
cross(new_forward,approximate_up);
new_up = cross(new_forward, new_side);
```

Fig. 3. Procedimentos para reortogonalização da base.

A validade da idéia baseia-se no fato de que a cada passo da simulação (*frame-to-frame*) as mudanças de orientação são pequenas. Os produtos vetoriais aplicados convenientemente garantem a reortogonalização dos vetores da base.

### Comportamentos Direcionais

Os comportamentos direcionais, desenvolvidos a partir do que foi discorrido anteriormente sobre o modelo simplificado do veículo, promovem o deslocamento deste no ambiente virtual.

Reynolds menciona em [8] vários tipos básicos de comportamentos direcionais que podem ser aplicados aos veículos, entre os quais se destacam:

- **Seek** (perseguição a um alvo estático) – figura 5;
- **Flee** (afastamento em relação a um alvo estático) – figura 5;

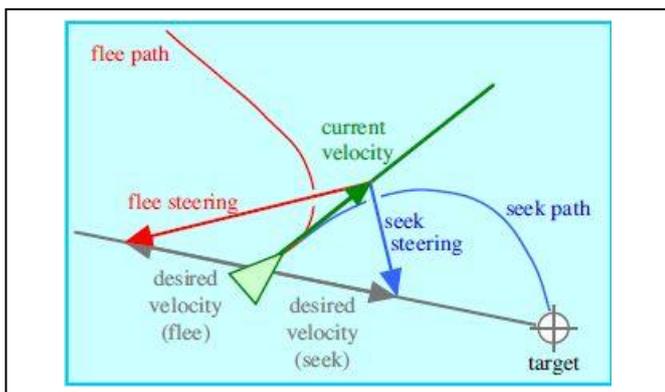


Fig. 5. Seek e Flee.

- **Pursuit** (perseguição a um alvo móvel, normalmente outro veículo, até o “toque”) – figura 6;

- **Evasion** (afastamento em relação a um alvo móvel, ou seja, manobras de fuga em relação a outro veículo) – figura 6;

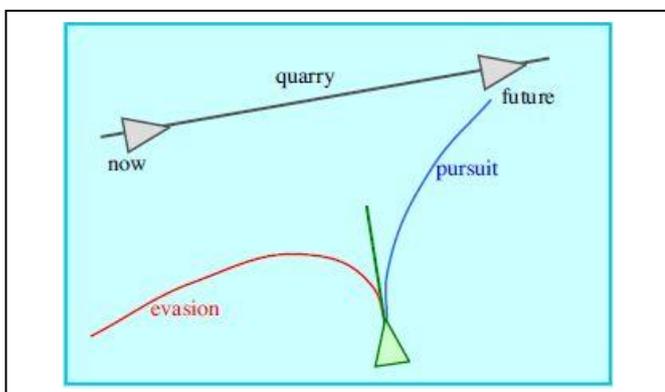


Fig. 6. Pursuit e Evasion.

- **Arrival** (chegada a um determinado alvo seguida da desaceleração necessária para não ultrapassá-lo e lá permanecer) – figura 7;

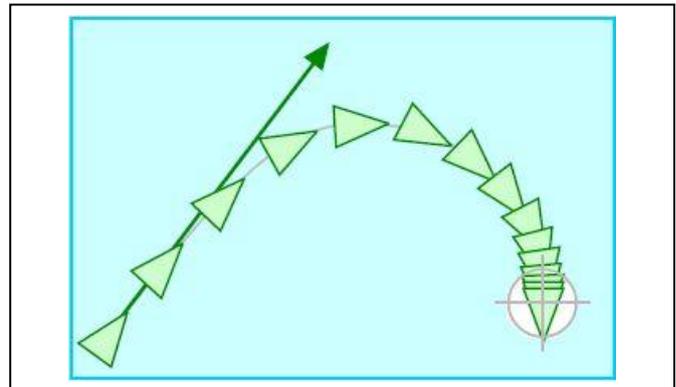


Fig. 7. Arrival.

- **Offset Pursuit** (análogo ao *Pursuit*, só que neste caso permanece a uma distância determinada similar a um acompanhamento) – figura 8;

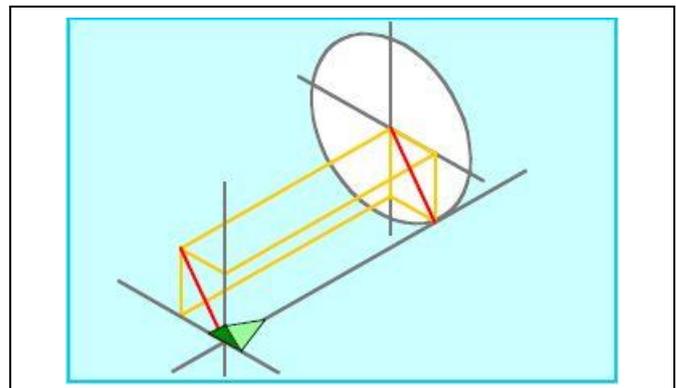


Fig. 8. Offset Pursuit.

- **Obstacle Avoidance** (habilidade de manobrar em ambiente restrito desviando-se de obstáculos quando estes se encontram próximos a trajetória a ser percorrida) – figura 9;

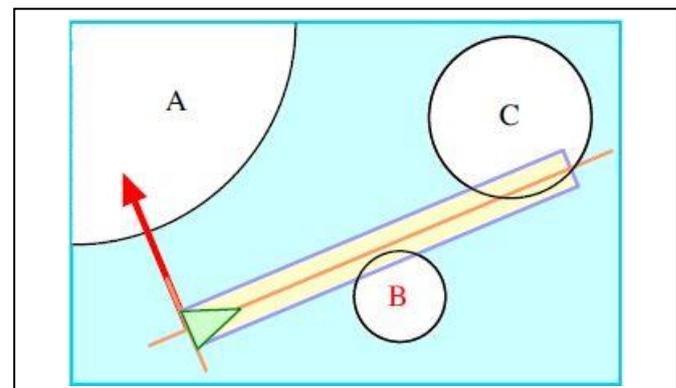


Fig. 9. Obstacle Avoidance.

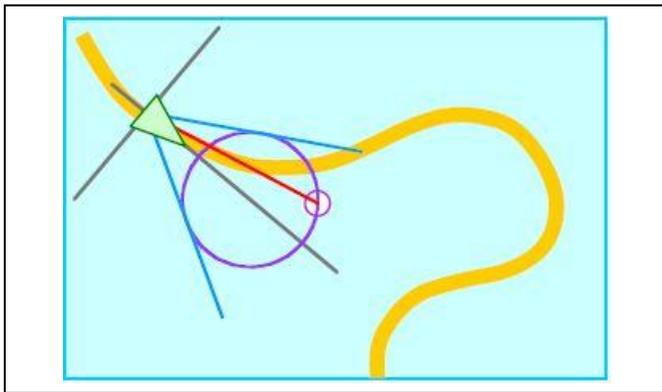


Fig. 10. Wander.

- **Wander** (tipo de direcionamento aleatório) – figura 10;
- **Path Following** (habilidade que permite ao veículo se locomover dentro dos limites de uma via pré-determinada, sem, contudo, estar rigidamente preso a ela tal qual acontece numa composição sobre os trilhos) – figura 11; e

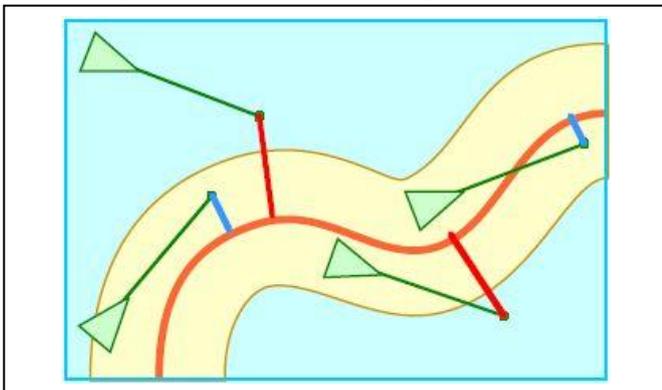


Fig. 11. Path Following.

- **Unaligned Collision Avoidance** (habilidade que tenta manter livre de colisão os veículos que se movimentam em direções arbitrárias, tal qual ocorre no movimento de pedestres em uma praça cheia de pessoas) – figura 12.

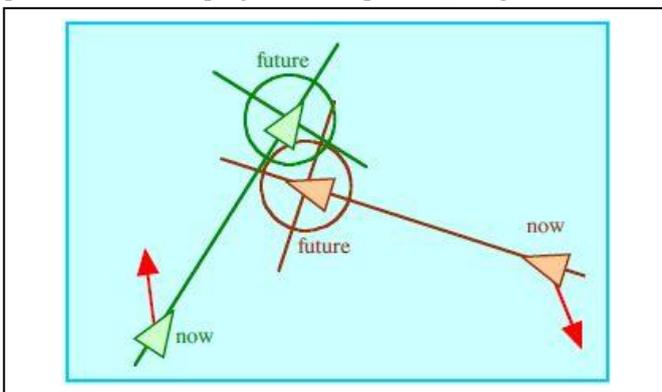


Fig. 12. Unaligned Collision Avoidance.

Entre os tipos citados, os dois últimos foram escolhidos para serem inseridos no ambiente de simulação, em virtude da relação entre os comportamentos modelados e a natureza do problema.

O comportamento *Path Following* possibilita a implementação do comportamento esperado de uma aeronave que se movimenta numa aerovia ou que mantém uma rota pré-estabelecida.

O comportamento *Unaligned Collision Avoidance* possibilita que, ao se provocar cruzamentos intencionais que simulem possíveis conflitos de tráfego, surjam manobras evasivas para evitar a colisão ou sobreposição de volumes de proteção ao redor de aeronaves.

Para testar as hipóteses supracitadas e estudar parâmetros da resolução de conflitos entre aeronaves simuladas, desenvolveu-se uma ferramenta de software para simulação na linguagem Java, o Simulador de Visualização de Controle de Tráfego – **VisuATCSim**.

- A interface de preparação de exercícios; e
- A interface de execução da simulação.

A primeira, de acordo com a figura 13, oferece as funcionalidades necessárias à montagem do cenário a ser estudado, tais como a disposição individual das aeronaves e dos pontos de controle da navegação (*waypoints*), o carregamento de arquivos de exercícios previamente definidos e gravação de novos arquivos, a definição das velocidades iniciais, a edição e exclusão de aeronaves e rotas.

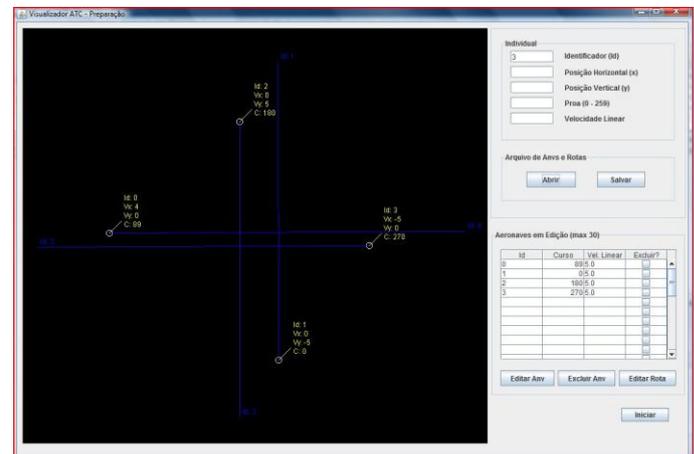


Fig. 13. Interface de Preparação - VisuATCSim.

A segunda, de acordo com a figura 14, oferece as funcionalidades necessárias ao acompanhamento da simulação e algumas interferências sobre esta, tais como a mudança da velocidade de reprodução, a inibição de informações sobre cada aeronave (etiqueta de voo), a visualização atualizada do rumo e velocidade de cada aeronave e as trajetórias percorridas e pretendidas.

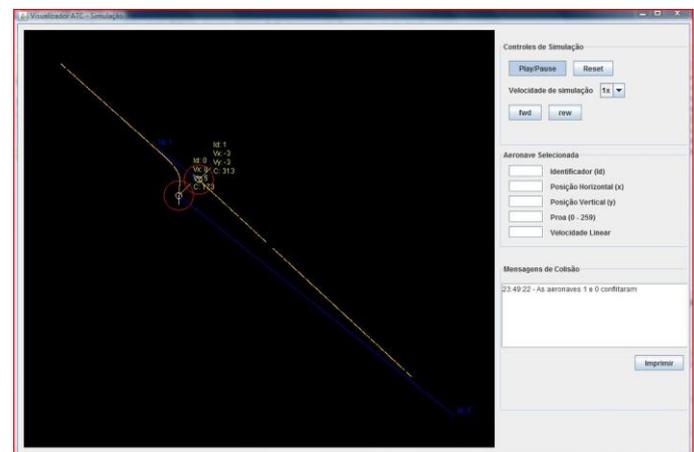


Fig. 14. Interface de Execução - VisuATCSim.

A título de trabalhos futuros poderiam ser inseridas novas funcionalidades no aplicativo, tais como:

- a escolha, em tempo de preparação de exercício, de aeronaves com diferentes performances para se analisar a eficiência da resolução de conflitos sob ambiente diversificado;
- a geração aleatória de aeronaves e de rotas associadas de modo a cobrir um espectro amplo de configurações de espaço aéreo visando criar o maior número de condições iniciais para a simulação;
- a leitura de dados de planejamentos de vôo das manobras militares, cujos mínimos de separação são bastante inferiores ao utilizado na aviação geral, a fim de se verificar a eficiência da resolução sob condições mais restritivas; e
- o registro sistematizado das manobras evasivas (aquelas que resolvem os conflitos) de acordo com a priorização de parâmetros (menor consumo, menor variação de velocidade, menor desvio da rota, entre outros).

#### IV. CONCLUSÃO

Neste artigo abordou-se a temática do controle de tráfego aéreo sob a ótica das taxas mundiais de crescimento, os índices de incidentes e a implantação de novos sistemas que irão compor o sistema CNS/ATM.

Tecnologias existentes e futuras permitirão que aeronaves voem cada vez mais próximas no espaço e no tempo, procurando atender às questões de ampliação, otimização e uso seguro do espaço aéreo controlado.

Procurou-se mostrar uma “imagem” presente e futura do cenário de tráfego no Brasil, fundamentada em diretrizes e orientações disponibilizadas pelo órgão internacional, a OACI, e pelo órgão governamental brasileiro, o DECEA, responsável pela normatização e gestão do sistema CNS/ATM.

Em função das considerações sobre o cenário, vislumbrou-se uma justificativa para a proposta de aplicação de uma biblioteca em C++ denominada OpenSteer, desenvolvida a partir de um modelo conceitual de um veículo simples que possibilitou a criação de modelos de comportamentos (habilidades) direcionais.

A adaptação para a linguagem Java de dois desses comportamentos, *Path following* e *Unligned Collision Avoidance*, e a inserção destes em ferramenta de software especialmente desenvolvida para a simulação de controle de tráfego aéreo (VisuATCSim) visou oferecer um meio para o estudo de parâmetros das resoluções de conflito entre aeronaves em ambiente simulado.

Novas idéias foram sugeridas de modo a propiciar a evolução do simulador para adequá-lo ao emprego em condições mais próximas as da realidade, tais como a diversificação de aeronaves e a utilização de dados de planejamento aéreo de manobras militares.

#### REFERÊNCIAS

- [1] <http://www1.folha.uol.com.br/folha/dinheiro/ult91u339410.shtml>. Acessado em 22 de agosto de 2008.
- [2] [http://www.defesenet.com.br/zz/br\\_aereo\\_20.htm](http://www.defesenet.com.br/zz/br_aereo_20.htm) Acessado em 22 de agosto de 2008.
- [3] [http://inovacao.scielo.br/scielo.php?script=sci\\_arttext&pid=S1808-23942007000500008&lng=es&nrm=iso](http://inovacao.scielo.br/scielo.php?script=sci_arttext&pid=S1808-23942007000500008&lng=es&nrm=iso) Acessado em 22 de agosto de 2008.
- [4] [http://www.aisweb.aer.mil.br/aisweb\\_files/indices/AIP-BRASIL/enr2\\_eng.pdf](http://www.aisweb.aer.mil.br/aisweb_files/indices/AIP-BRASIL/enr2_eng.pdf) Acessado em 22 de agosto de 2008.
- [5] <http://ifly.nlr.nl/> Acessado em 17 de março de 2009.
- [6] G. Chaloulos, J. Lygeros, I. Roussos, K. Kyriakopoulos, E. Siva, A. Lecchini-Visintini, P. Casek. “Comparative Study of Conflict Resolution Methods”. iFly. 2008.
- [7] <http://opensteer.sourceforge.net/>. Acessado em 22 de abril de 2009.
- [8] Reynolds, C. W. “Steering Behaviors For Autonomous Characters”, *Game Developers Conference 1999*, San Jose, California. Miller Freeman Game Group, San Francisco, California, pp 763-782.
- [9] BRASIL. Comando da Aeronáutica. ICA 100-12 - **Regras do Ar e Serviços de Tráfego Aéreo**. Rio de Janeiro, 2006.
- [10] ICAO. *International Civil Aviation Organization*. Doc 9863 AN/461 – *Airborn Collision Avoidance System (ACAS) Manual*. 2006.
- [11] BRASIL. Comando da Aeronáutica. PCA 351-3 - **Programa de Transição do SISCEAB utilizando o conceito de Sistema CNS/ATM**. Rio de Janeiro, 2006.
- [12] ICAO. *International Civil Aviation Organization*. Doc 9694-AN/955– *Manual of Air Traffic Services Data Link Applications*. 1999.