

Relatório Final

Título do projeto de pesquisa: Construção de um material didático para o Modelamento e Simulação 6DOF de um Míssil Terra Ar

Bolsista: Heitor Rodrigues Leão Faria

Orientador(a): Filipe Rodrigues de Souza Moreira

Período a que se refere o relatório: Fevereiro de 2016 a Agosto de 2016

Resumo

O trabalho desenvolvido trata-se da criação de uma simulação em 6DOF de um míssil terra ar, através de cálculos dinâmicos que descrevem as forças e momentos atuantes no artefato e que simulam o seu guiamento colocando-o em direção ao seu alvo. A simulação envolve o movimento do alvo, do míssil em si, a sua aerodinâmica, seu sistema de propulsão, a ação da gravidade e sua estratégia para o guiamento. Desenvolvida em ambiente Matlab, a simulação inicia-se com as declarações das principais variáveis e das condições iniciais, em seguida o loop calcula o estado dos movimentos ponto a ponto e, por fim, a plotagem dos resultados.

1. Introdução

Foi criada uma rotina que confere a possibilidade de prever comportamentos e obter estimativas de variáveis essenciais a um projeto de míssil.

Ao rodar a simulação, espera-se que o míssil realize o comportamento dinâmico de perseguição correta em direção ao alvo, a eliminação de “bugs” e descrição de comportamentos anormais. Os gráficos devem ser de fácil visualização e interpretação, trazendo informações objetivas e claras sobre todo o voo.

2. Materiais e métodos

O desenvolvimento do equacionamento e da metodologia para desenvolvimento da rotina de simulação são apresentados na literatura “Missile Flight Dynamics”.

Toda metodologia teórica e a fundamentação de programação utilizada foram descritas no relatório parcial, a referência [1]. A partir deste ponto, os trabalhos foram realizados com o intuito de descrever e corrigir os erros de programação.

É importante acrescentar que foi incluído na dinâmica do míssil o efeito do vento. A abordagem utilizada foi conforme descrita em [3].

$$v_w = v_{wref} \left(\frac{h}{h_{ref}} \right)^{\frac{1}{7}} \quad (1)$$

A direção do vetor velocidade do vento foi considerada ser aleatória, com variações na direção Ψ entre $[-\pi, \pi]$ e na direção θ entre $[0, \pi]$.

2.1 Desenvolvimento da Rotina de Simulação

A partir dos desenvolvimentos teóricos indicados em [1], foi seguido o projeto da simulação. O ambiente Matlab permite ao usuário dividir seu programa em várias funções individuais que, após serem programadas corretamente, pode trabalhar em conjunto e trazer os resultados finais. Essa metodologia adotada permite que cada função seja avaliada independentemente, facilitando a localização de “bugs” permitindo suas correções.

A simulação total conta com um conjunto de 28 funções, tendo uma função principal que irá fazer uso de todas as outras e criar o loop necessário para a simulação. As funções serão aqui descritas em comentários para nortear o leitor dos detalhes:

1 – Função Míssil_Perseguidor.m: Essa é a principal função do programa, ela irá chamar todas as outras na ordem determinada e gerar o loop da simulação. Para entender melhor como ela trabalha, as demais funções do programa serão apresentadas em ordem de chamada. Para exemplificar como é esse processo, têm-se o seguinte comando:

```
[g, Fg, altitude] = GRAVIDADE(m, Lnb, PM);
```

Na Missil_Perseguidor.m as funções são chamadas de forma semelhante ao modelo acima. Para a função GRAVIDADE, são entrados os valores de m (massa), Lnb (matriz de conversão de referenciais) e PM (vetor de posição do míssil). Os valores retornados são g (aceleração gravitacional), Fg (vetor de forças gravitacionais que atua no míssil no referencial do Corpo) e altitude (valor da altitude que o míssil se encontra em um dado instante). Caso

alguma variável declarada dentro dessa função porém não foi solicitada no vetor ao lado esquerdo da linha de código, então essa variável não será salva no Workspace do Matlab.

Assim, ao iniciar a função Míssil perseguidor, ela irá trabalhar com as funções na mesma ordem que serão apresentadas.

2 – Função Inicializa_MASSA.m: Declara variáveis iniciais e finais da massa, momento de inércia e centro de massa do míssil. Esses valores se alteram devido a variação de massa gerada pela queima de combustível.

3 – Função Inicializa_TARGET.m: Cria as variáveis das condições iniciais do alvo, como sua posição e velocidade inicial em relação ao míssil.

4 – Função Inicializa_POSICIONAMENTO.m: Essa função recebe o vetor de posição inicial do alvo, declara as coordenadas iniciais do míssil, calcula os vetores do alcance, da velocidade relativa, da velocidade do míssil, cria a matriz de conversão de referenciais da Terra para o corpo, calcula os ângulos de Euler, a velocidade translacional do míssil em relação ao corpo (realiza a conversão de referenciais) e declara os ângulos iniciais de rolamento, arfagem, guinada, ataque e derrapagem.

4 – Função ATMOSFERA.m: Realiza os cálculos de valores da atmosfera em função da altura que o míssil se encontra, ela retorna os valores de Mach, pressão atmosférica e pressão dinâmica.

5 – Função Inicializa_Prop.m: Recebe o valor da pressão atmosférica e retorna o vetor das forças geradas pelo sistema de propulsão em relação ao eixo da Terra, essa força é somente na direção z, mas ao ser transposta para o referencial do corpo essa distribuição muda.

6 – Função Inicializa_SEEKER.m: Responsável por gerar as variáveis do rastreador e calcular os versores iniciais que apontam o míssil em direção ao alvo. Recebe os valores de alcance, velocidade do míssil, velocidade do alvo, e vetor unitário da linha de centro do míssil. Calcula o valor do ângulo Gimball, que é o ângulo gerado entre a linha de centro do míssil e a linha de visão do rastreador. Dentro da função há condições de interrupção da simulação caso esse ângulo ultrapasse seu limite e se perca do alvo.

7 – Função Inicializa_AUTOPILOTO: Calcula o vetor das acelerações laterais aplicadas ao míssil devido as suas interações aerodinâmicas das suas superfícies de controle. Recebe o valor do tempo, do módulo da velocidade do míssil, a matriz de conversão de referencial e a taxa de variação do ângulo de rolamento. Retorna os valores das deflexões dos ângulos das superfícies de controle.

8 – Função INICIALIZA_AED: Declara os valores iniciais aerodinâmicos do míssil, recebe os valores das deflexões das superfícies de controle, os componentes de velocidade do míssil no referencial do Corpo, o valor das velocidades angulares, pressão dinâmica, ângulos de ataque e derrapagem, módulo da velocidade e a posição do centro de massa inicial do míssil. Ela retorna os valores das forças aerodinâmicas no referencial do corpo que agem no míssil e dos momentos aerodinâmicos, exceto para o ângulo de rolamento.

9 – Função X_PONTO: Calcula os valores das derivadas de estado para a simulação, valores que serão somados aos do passo anterior durante o loop. Recebe as variáveis de forças e momentos (aerodinâmico, gravitacional e propulsivo), a velocidade do míssil no ponto, momento de inércia e ângulos de Euler. Retorna os valores das derivadas de espaço da velocidade do míssil, da variação angular e dos ângulos de Euler. O método numérico utilizado foi mudado, sendo o método de Euler trabalhado. Apesar de a bibliografia referenciar o método de Runge Kutta de quarta ordem, optou-se, por simplicidade, pelo uso do método de Euler,.

10 – Função INTEGRA.m: Adiciona aos valores dinâmicos dos movimentos do míssil a parcela diferencial, ou seja, as variações de cada valor gerada pelo método numérico utilizado. Recebe os valores retornados da função X_PONTO.m, e retorna os valores integrados para aquele ponto.

11 – Função T_earth_body.m: Recebe os valores dos ângulos de Euler e retorna a matriz de conversão de referencial, responsável por atualizar a matriz a cada passo do loop.

12 – Função POSICIONAMENTO.m: Realiza o cálculo da posição do míssil no segundo passo da integração do movimento, que é feito sobre a velocidade de cada passo do loop calculada para o míssil, sendo em seguida convertida para o referencial da Terra. Nessa função é acrescido a cada componente do vetor velocidade (no sistema do corpo do artefato) a respectiva componente da velocidade do vento, conforme equação (1). A velocidade que é convertida para o referencial da Terra é, na verdade, o vetor velocidade relativa entre o artefato e o vento. A função POSICIONAMENTO.m realiza seus cálculos através dos seguintes parâmetros: velocidade do míssil, da posição do míssil, posição do alvo, matriz de conversão de referencial, variação infinitesimal do tempo (0.001s), velocidade do alvo e vetor unitário da linha de centro do míssil.

13 – Função Inicializa_RESULTADOS.m: Recebe todas as variáveis que ao serem salvas por todo o loop, cria um vetor com as condições iniciais da simulação. A função retorna o vetor coluna com todas as condições iniciais.

14 – Função ATUALIZA_TEMPO.m: Responsável por atualizar o tempo a cada loop realizado.

15 – Função GRAVIDADE.m: Calcula o valor das forças impostas ao míssil sob influência do campo gravitacional terrestre, além disso já faz a conversão de referencial da Terra para o corpo que serão os valores de forças utilizados na equação diferencial. Retorna o valor da aceleração gravitacional, o vetor de forças gravitacionais no míssil com referencial do corpo e a altitude. É utilizada dentro do loop.

16 – Função ATMOSFERA.m: Recebe o valor da aceleração da gravidade, o módulo da velocidade e altitude. Retorna o valor da pressão atmosférica, pressão dinâmica e número de Mach. É utilizada dentro do loop.

17 – Função PROP.m: Realiza a interpolação da tabelas de valores da força de propulsão em função do tempo. Então, o vetor de forças geradas pela propulsão é salvo para ser trabalhado na resolução da equação diferencial numericamente. É utilizada dentro do loop.

18 – Função MASSA.m: Atualiza os valores do centro de massa do míssil e momentos de inércia. Recebe os valores do centro de massa, momento de inércia, tempo, massa, momentos iniciais e força de propulsão. Retorna os valores atualizados para o passo do loop da massa, momento de inércia e centro de massa. É utilizada dentro do loop.

19 – Função SEEKER.m: Essa função recebe os valores do alcance, ângulos gimball, tempo, ângulos de gimball máximo, variação angular da linha da linha de centro do míssil do passo anterior, tempos de processamento do rastreador. Retorna os valores atualizados das variações angulares da linha de centro. É utilizada dentro do loop.

20 – Função AUTOPILOT.m: Recebe os valores das variações angulares da linha de centro, do tempo, módulo da velocidade, vetor dos ângulos Gimball, matriz de conversão de referenciais, vetor unitário do alcance, variação angular de guinada, arfagem e rolamento. Retorna os valores das deflexões das superfícies de controle e da aceleração lateral gerada. É utilizada dentro do loop.

21 – Função AERODINAMICA.m: Recebe os valores da área molhada das superfícies de controle, das deflexões das superfícies de controle, número de Mach, pressão dinâmica,

módulo da velocidade, ângulos de ataque e variações angulares. Essa função retorna as forças e momentos aerodinâmicos no referencial do Corpo do míssil. É utilizada dentro do loop.

As funções citadas a seguir são inicializadas dentro do loop e já foram descritas anteriormente, na qual são usadas para recalculer os valores apresentados. A ordem das funções são:

22 – X_PONTO.m

23 – INTEGRA.m

24 – POSICIONAMENTOS.m

Depois de utilizar essas três funções, a Missil_Perseguidor.m continua com as seguintes funções:

25 – Função RESULTADOS.m: Faz-se o uso de todas as variáveis que se alteram durante a simulação e as salva em um novo vetor coluna. A cada loop, um novo vetor coluna é criado, gerando assim uma matriz com todos os resultados que serão plotados em seguida.

26 – Função STOPSIM.m: Utilizada dentro do loop, é a responsável por verificar todas as condicionais que podem interromper o loop. As condições que podem gerar esse tipo de situação são:

- Se o loop ultrapassar o número de 100000 interações, nessa situação pode-se interpretar que o míssil não alcança o alvo.
- A simulação ultrapassar o tempo máximo determinado inicialmente, no qual pode ter a interpretação que o míssil não alcança o alvo.
- A distância mínima de aproximação é igual ao módulo do alcance, nessa situação o míssil atingiu o alvo.
- A distância mínima de aproximação é menor do que 3 metros, nesse caso também interpretado que o míssil atingiu o alvo.
- Se a velocidade relativa entre o míssil e o alvo tornou-se um valor negativo, o que seria interpretado que o míssil atingiu o alvo ou passou por ele (nesse caso também seria necessário averiguar o gráfico 3D gerado).
- Se a altura que o míssil se encontra tornar um número negativo, nesse caso interpreta-se que o míssil colidiu com o solo.
- Se os ângulos de deflexões do míssil atingiram seus valores máximos, desta forma poderia ser interpretado que eles perderam o rastreamento do alvo.

27 – Função ATUALIZA_TEMPO.m: Novamente ela se repete dentro do loop para atualizar o tempo de cada passo de simulação. Nesse momento, é utilizada dentro do loop.

Ao finalizar todo o loop, a função Missil_Perseguidor.m cria uma nova matriz onde salva todos os resultados da simulação, por fim essa matriz é inserida na última função.

28 – Função Plotagem.m: Recebe a matriz de resultados, a distância de aproximação do míssil, do alvo e uma variável que representa o motivo do fim da simulação. Com esses valores a função plota todos os gráficos gerados, o gráfico 3D do deslocamento do míssil e do alvo, o gráfico do módulo da velocidade do míssil, e os gráficos das taxas de variações angulares do míssil durante sua trajetória.

3. Resultados

Os gráficos gerados ao final da simulação são apresentados a seguir:

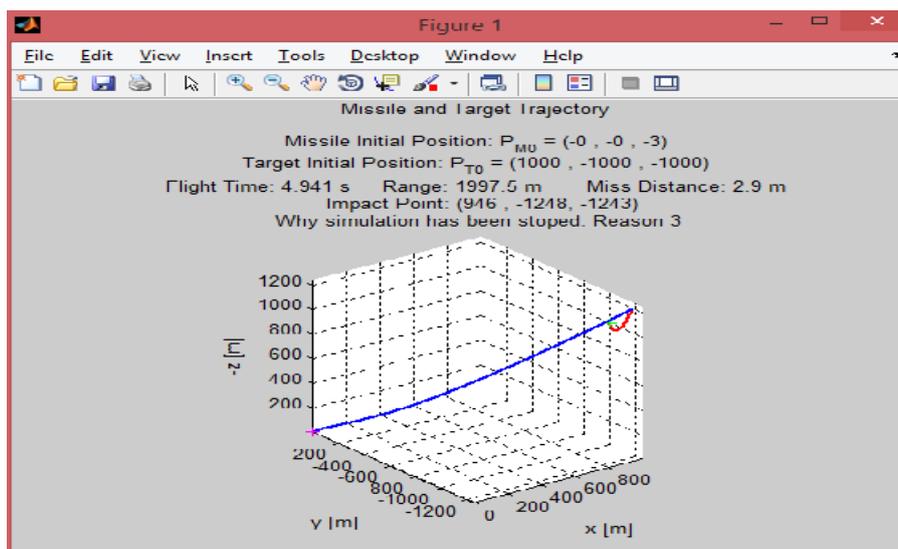


Figura 1 – Figura 3D representando o deslocamento do míssil (linha azul) e do alvo (linha vermelha)

Na visão das linhas de deslocamento nota-se que o míssil alcançou o alvo. Outras informações importantes são fornecidas como o ponto inicial do míssil (0,0,-3) (referencial no eixo z é adotado como negativo), ponto inicial do alvo (1000,-1000,-1000), tempo de voo do míssil (4,941s), alcance (1997.5 m), distância mínima de aproximação do míssil e do alvo (2.9m), ponto de impacto (946,-1248,-1243) e a razão porque a simulação parou (caso 3), isso

ocorreu devido a distância mínima de aproximação do míssil ter alcançado um valor inferior a 3 (função STOPSIM.m).

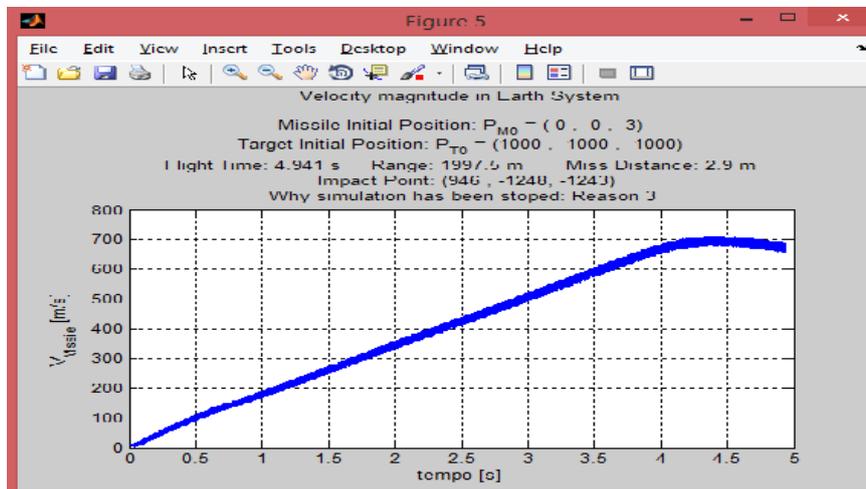


Figura 2 – Gráfico do módulo da velocidade do míssil em função do tempo de simulação

Pela análise do gráfico nota-se que a variação do módulo da velocidade é praticamente linear durante quase toda a simulação, cerca de 4 segundos após o voo, a velocidade começa a se tornar constante e em seguida ter uma leve queda ao alcançar o alvo. Isso ocorre devido as forças do sistema de propulsão que começa a diminuir após 4 segundos e cessarem (condição em que todo combustível foi consumido) aos 5.6 segundos. Observa-se um caráter levemente oscilatório, em todo o voo, para o módulo do vetor velocidade: isso se deu pela introdução do modelo do vento. Como em cada passo de integração o módulo e a direção do vento são diferentes, é natural que haja uma pequena variação dessa variável em torno do valor que ela teria caso o vento não estivesse sendo considerado.

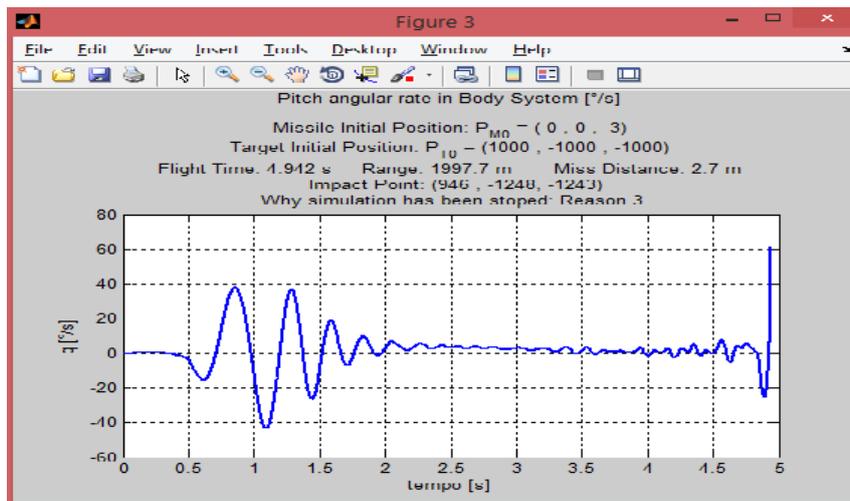


Figura 3 – Gráfico da taxa de variação do ângulo de guinada do míssil em função do tempo de simulação

Nesse gráfico observa-se que a variação angular de guinada é mínima no início da simulação, após esse período sofre grandes variações entre os tempos de 0.5 a 2 segundos, que é quando há maior influência do sistema de auto piloto, que é somente ativado depois de 0.5 segundos. Somente no próximo instante do míssil encontra-se o alvo há uma grande variação novamente.

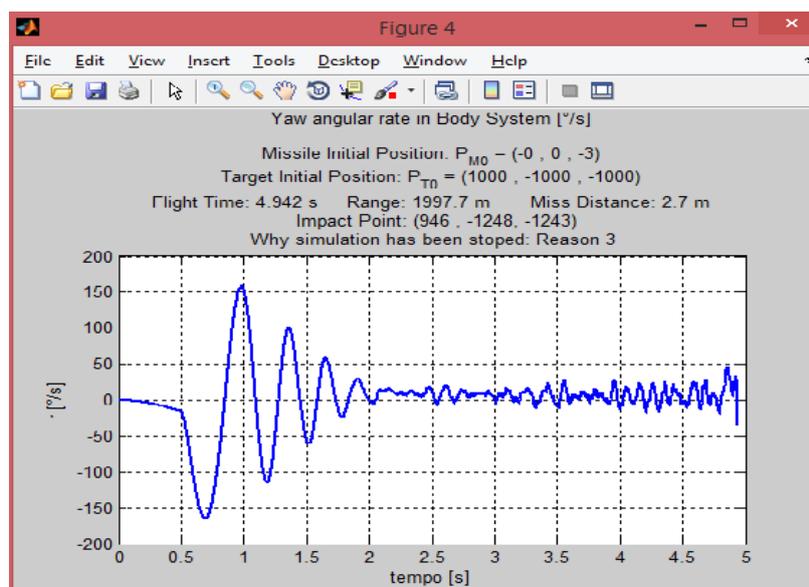


Figura 4 – Gráfico da taxa de variação do ângulo de arfagem do míssil em função do tempo de simulação

No gráfico verifica-se que também o período de maior instabilidade de movimento foi entre os tempos de 0.5 a 2 segundos, quando o auto piloto é ativado. Uma vez estabilizado, o míssil segue em direção ao alvo sem grandes variações angulares, o que representa o trabalho do sistema de rastreamento em conjunto com o sistema de auto piloto, que conduz o míssil corretamente na direção destinada à sua missão.

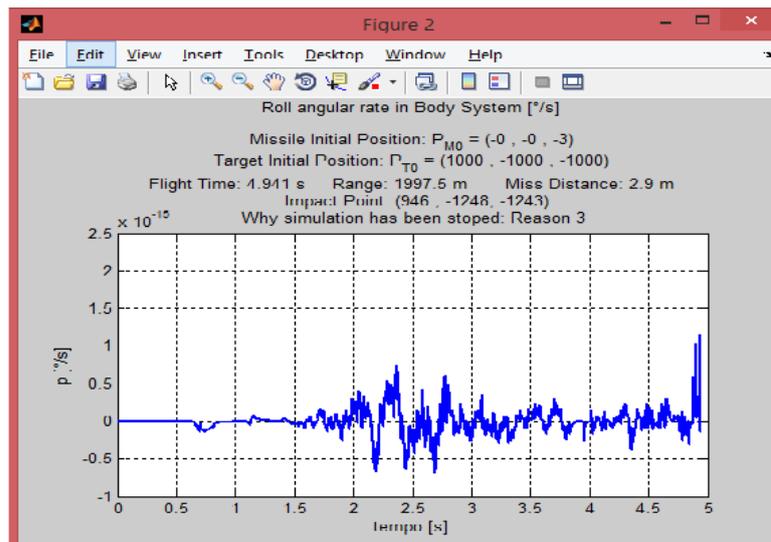
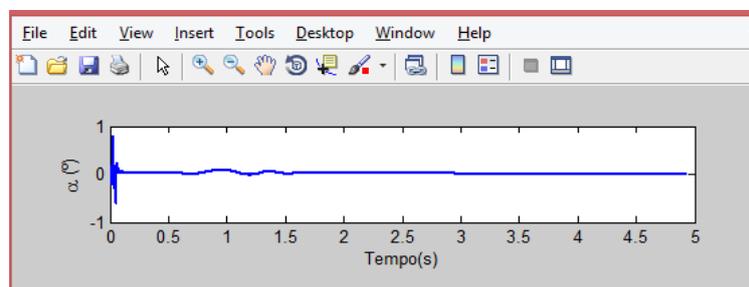


Figura 5 – Gráfico da taxa de variação do ângulo de rolamento do míssil em função do tempo de simulação

As variações dos ângulos de rolamentos, apesar de plotadas, não relevam o sexto grau de liberdade do míssil. Devido à falta de dados referentes à aerodinâmica da influência do ângulo de rolamento, o ângulo de Euler referente ao eixo x (eixo de rolamento) foi adotado como zero durante toda a simulação. Sendo portanto uma simulação de 5GDL.



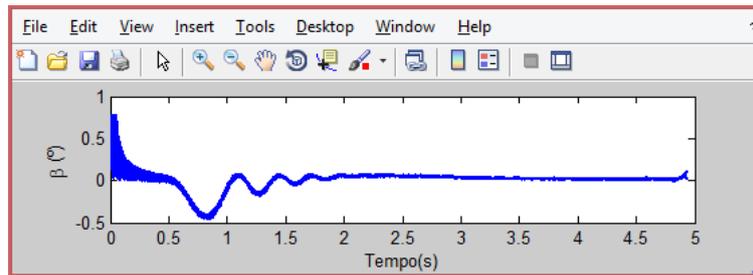


Figura 6 – Gráficos da variação do ângulo de ataque e de derrapagem em função do tempo.

O ângulo de ataque do míssil possui maiores variações somente no início do movimento, mas depois permanece constante durante o voo. O ângulo de derrapagem sofre maiores variações entre 0,5 e 2 segundos, estabilizando-se em seguida.

4. Conclusões

O trabalho realizado criou uma simulação dinâmica que pode facilitar futuros projetos de mísseis. Algumas modificações e atualizações devem ser feitas, tanto para incluir os dados aerodinâmicos do sexto grau de liberdade e adaptá-la com uma nova interface para ser utilizada com maior facilidade.

Com a simulação, foi criado um material didático para demonstrar todo conhecimento sobre mísseis e seu funcionamento geral. Os resultados mostrados nos gráficos descrevem visualmente o comportamento de trajetórias alvo e míssil, como deve ser apresentado em uma simulação. Além de permitir plotar outros parâmetros como suas variações angulares rotacionais e ângulos de ataque e derrapagem. Dessa forma, permite uma análise dinâmica completa e bem didática, com resultados e conclusões objetivos.

Agregou-se conhecimento no funcionamento geral de um míssil, na resolução numérica de equações diferenciais, na programação do ambiente Matlab e lógica de programação, na visão geral de influências aerodinâmicas para controlar um artefato em voo em conjunto com seu sistema de guiamento e controle.

Referências

- [1] FARIA, H, Relatório Parcial. PIBIC IAE (2015/2016)
- [2] STRICKLAND, J Missile Flight Simulation : ISBN-10: 1105102815
- [3] HEIER, S. Grid Integration of Wind Energy Conversion Systems. Chichester: John Wiley & Sons. p. 45. ISBN 0-470-86899-6.