

DESENVOLVIMENTO DE UMA INTERFACE GRÁFICA PARA DINÂMICA DOS FLUIDOS COMPUTACIONAL, UTILIZANDO A METODOLOGIA DE FRONTEIRA IMERSA

Gustavo Prado de Oliveira

UFU / FEMEC / LTCM
gpoliveira@mecanica.ufu.br

João Marcelo Vedovoto

vedovotojm@yahoo.com
UFU / FEMEC / LTCM

Ana Paula Fernandes

UFU / FEMEC / LTCM
apfernandes@mecanica.ufu.br

Aristeu da Silveira Neto

UFU / FEMEC / LTCM
aristeus@mecanica.ufu.br

Universidade Federal de Uberlândia – Faculdade de Engenharia Mecânica – Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG - CEP: 38400-902

Resumo: Atualmente, o Laboratório de Transferência de Calor e Massa e Dinâmica dos Fluidos - LTMC/FEMEC/UFU possui diversos trabalhos para simulações e análise computacional da dinâmica dos fluidos, porém com etapas providas por softwares proprietários, tornando onerosa a utilização de tais soluções. Além disso, a ferramenta atualmente funcional existente, que efetua tais cálculos, foi desenvolvida na linguagem Fortran, desprovida de uma interface gráfica, além de não permitir uma visualização prévia das geometrias a serem trabalhadas, o que dificulta o manuseio da mesma. Dessa forma, no presente trabalho pretende-se desenvolver uma ferramenta para a construção de geometrias e a geração de suas malhas de elementos finitos triangulares superficiais, utilizando-se de softwares livres, para posterior importação em códigos computacionais dedicados a análise computacional da dinâmica dos fluidos, através do método de fronteira imersa. Dentre as ferramentas analisadas para a modelagem da geometria e geração da malha, optou-se pela utilização do Gmsh, por ser esse um software livre, portátil para as plataformas Linux e Windows, intuitivo e de fácil aprendizado. Já dentre as ferramentas analisadas para o desenvolvimento da Interface gráfica, optou-se pela QT, por ser uma aplicação compreensiva para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataforma, além de ser também portátil para Windows e Linux. O trabalho apresentará algumas soluções de geometrias / malhas geradas no Gmsh, e a preparação dessas soluções para a utilização do método de fronteira imersa na análise computacional da dinâmica dos fluidos em geometrias como esferas, cilindros, aerofolios, buscando-se utilizar os softwares para trabalhar com formas mais complexas como aeronaves e automóveis.

Palavras chave: fronteira imersa, geração de malhas, software livre

1. INTRODUÇÃO

Atualmente, o Laboratório de Transferência de Calor e Massa e Dinâmica dos Fluidos - LTMC/FEMEC/UFU possui diversas ferramentas para simulações e análise computacional da dinâmica dos fluidos. No entanto, para efetuar tais simulações são necessários softwares proprietários (requerem registro) o que dificulta o desenvolvimento de aplicações acadêmicas devido ao custo dos mesmos.

Além disto, a ferramenta de mecânica dos fluidos encontrada no LTCM, não oferece uma interface amigável de forma a facilitar o manuseio da aplicação. Para tal, será utilizadas técnicas de programação visando criar uma **interface gráfica (GUI, Graphical User Interface)**, que é uma ferramenta de interação entre usuário e sistema computacional baseada em símbolos visuais, como ícones, menus e janelas. Seguindo esta linha de desenvolvimento, pode-se oferecer uma ferramenta simples para resolver problemas acadêmicos com entradas paramétricas e capacidade avançada de visualização. Com isso pode-se delimitar visualmente as etapas de pré e pós-processamento, gerando arquivos de saída que possam ser usados com os códigos existentes no LTCM, possibilitando maior facilidade de uso.

Juntamente com o problema de interação computacional, outro grande desafio em engenharia é a solução numérica da ação de forças sobre corpos imersos em escoamentos, com suas possíveis deformações e deslocamentos, para tal problema uma solução viável pode ser obtida através da modelagem matemática.

Evidentemente, é notório o crescimento dos recursos computacionais, viabilizando a realização de simulações numéricas cada vez mais aperfeiçoadas de escoamentos em domínios de topologia diversas. Em geometrias complexas, a escolha apropriada da malha computacional depende da metodologia de discretização do domínio contínuo. Desta forma, um algoritmo desenvolvido, por exemplo, para lidar com malhas curvilíneas ortogonais, não é capaz de lidar com malhas não-ortogonais. Assim, procura-se lidar com o problema, empregando malhas que se ajustam à sua topologia, quer seja acompanhando a superfície do corpo a ser estudado, quer seja se adaptando melhor ao escoamento, melhorando a qualidade da solução.

Escoamentos ao redor de geometrias complexas ainda apresentam sérias dificuldades em serem resolvidos, pois as geometrias apresentam problemas na geração da malha computacional, que nem sempre são triviais, e são propensa a inserir severos erros nos balanços entre seus nós elementares. Isto acontece quando se procura ajustar a malha computacional ao objeto de estudo e ao domínio, nas discretizações.

Nos casos de discretizações em blocos, o acoplamento entre os diversos domínios, se mal construído, pode implicar em sérias inconsistências físicas. A discretização de domínios tridimensionais exige grande capacidade de armazenamento e processamento de dados que, se somados aos freqüentes refinamentos locais das malhas, torna os cálculos muito caros.

Uma metodologia alternativa para se resolver escoamentos sobre geometrias complexas e móveis vem sendo desenvolvida no LTCM. Trata-se da metodologia de Fronteira Imersa (*Imersed Boundary - IB*). Neste método, a presença de uma interface fluido-sólido, ou mesmo uma interface fluido-fluido, pode ser simulada pela adição de um termo de força (que alguns autores denominam como termo forçante) às equações do movimento do fluido. Dessa forma, o escoamento reconhece a presença de um objeto imerso, por exemplo, apenas pela ação de um "campo de força". A forma como este campo de força é calculado diferencia a metodologia empregada atualmente. Cabe salientar que tais forças são formadas a partir de efeitos diversos, agindo ao mesmo tempo no fluido, em decorrência de cisalhamentos e deformações, variações na quantidade de movimento e gradientes de pressão. Silva(2002), Campregher(2005)

O uso da metodologia *IB* requer uma malha cartesiana para a solução do escoamento, independentemente da complexidade e da mobilidade da fronteira imersa. No entanto, requer-se a construção de uma malha superficial para a interface fluido-sólido. Empregando-se uma malha triangular para a discretização desta interface, torna-se possível aproveitar as facilidades associadas a este tipo de malha, para a geração de geometrias complexas.

Para tanto, no presente trabalho optou-se por utilizar o software livre Gmsh como ferramenta geradora desta malha, devido a facilidade de criar geometrias complexas em formatos padrões como o STL. Além disso mostrar-se-á que geometrias geradas com essa ferramenta apresentam resultados compatíveis com outras geometrias geradas por softwares proprietários de renome, apresentando algumas soluções de geometrias/malhas geradas no Gmsh.

Dentre as ferramentas analisadas para o desenvolvimento da Interface gráfica, optou-se pelo QT, por ser uma aplicação compatível para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataforma, além de ser também portátil para Windows e Linux.

2. GMSH

Gmsh é um gerador automático de malhas (primariamente de Delaunay) tridimensionais em elementos finitos com facilidades de pré e pós-processamento. Ele foi desenvolvido para oferecer uma ferramenta simples para resolver problemas acadêmicos com entradas paramétricas e capacidade avançada de visualização. Uma de suas principais características é a facilidade em respeitar a regularidade na geração de malhas adaptativas em linhas, superfícies e volumes, agrupando estas estruturas uma malha única.

Gmsh está construído com base em quatro módulos, dentre os quais, os de interesse para este projeto são: geometria e malha (módulos utilizados na construção de malhas de geometrias). Todas essas instruções são prescritas utilizando-se de interface gráfica de usuário (GUI) ou em arquivos de dados ASCII usando a própria linguagem script do Gmsh. Ações interativas geram bits de linguagens em arquivos de saída e vice-versa, o que torna possível automatizar todo tratamento, utilizando *loops*, quanto as chamadas de sistemas externos e condicionais.

Geometrias são criadas pela definição sucessiva de pontos, linhas orientadas, superfícies orientadas e volumes, compondo grupos de entidades geométricas que podem ser definidas, baseando-se nestas entidades geométricas. A linguagem *script* do Gmsh permite que toda entidade geométrica seja totalmente parametrizada.

Malha de elementos finitos viabiliza a construção de um dado subconjunto de espaços tridimensionais por elementos de várias formas geométricas elementares (linhas, triângulos, quadrados, tetraedros, prismas, hexaedros, e pirâmides), arranjos de tal forma que se duas destas formas se interceptam, eles formam uma face, uma borda ou um nó. A geração de malha segue a mesma performance da criação de geometria: define-se as coordenadas de cada ponto

no eixo cartesiano; os pontos são usados para definir linhas existente entre eles; limitada pelas linhas, define-se uma superfícies, então, através das superfícies pode-se definir uma geometria complexa para se gerar a malha e também os volumes.

Solver externos podem ser inter-relacionados com Gmsh, através do Unix ou sockets TCP/IP, o que permite encaminhar facilmente cálculos externos para coletar e aproveitar resultados de simulação dentro do módulo de pós-processamento do Gmsh.

3. Fatores de qualificação do Gmsh como gerador de malha:

- Descrição simples e rápida de geometrias, graças a funções de usuário definidas: *loops*, funções condicionais e bibliotecas;
- Parametrização de geometrias. A linguagem *script* do Gmsh faz com que todos os comandos e seus argumentos dependam de cálculos prévios;
- Gera com simplicidade 1D, 2D e 3D;
- Gmsh oferece vários mecanismos para controlar o tamanho dos elementos na malha final: através de interpolação de comprimento característico de pontos geométricos;
- Cria exportações simples de geometrias e malhas;
- Interação com *solver* externo. Gmsh oferece interfaces C, C++, Perl, Python, e outras linguagens que podem ser facilmente adicionadas;
- Visualização de resultados computacionais de grande variedade;
- Exporta plotagem em diferentes formatos: Látex, PNG, JPEG, STL;
- Gera animações complexas;
- Executável em várias plataformas (Windows, Mac e Unix);
- É gratuito.

4. Linguagens de programação

Como a principal preocupação, em achar um gerador de malha que acatasse as características necessárias para resolver os problemas físicos, foi resolvida, a primitiva agora é valer-se de uma ferramenta de programação capaz de dar suporte a ferramenta geradora de malha escolhida, para que se possa desenvolver uma ferramenta capaz de efetuar os cálculos necessários para problemas de interação fluido-estrutura.

Para tanto, é necessário ressaltar que, para desenvolver uma interface de uso amigável e de boa aparência, os melhores tipos de linguagem de programação, são aquelas que trabalham com desenvolvimento de aplicações visuais de interface com o usuário. Para isso, tal ferramenta deve se valer de uma Biblioteca de componentes visuais (VCL – *Visual Component Library*) consistindo de objetos reutilizáveis, incluindo objetos padrões de interface com o usuário, gerenciamento de dados, gráficos e multimídia, gerenciamento de arquivos e quadros de dialogo padrão.

Como a interface gráfica terá que ser baseada na plataforma Linux(plataforma base do cluster utilizado no LTCM), pode-se descartar facilmente a linguagem Delphi, que apesar de se enquadrar perfeitamente nas características acima citadas, esta ferramenta está disponível somente para Windows, além de não ser uma ferramenta gratuita. Com isso, uma opção que seria plausível é o Kylix, que é um Delphi criado para plataforma Linux, porém, seus direitos autorais são de poderio da Borland, o mesmo proprietário do Delphi, sendo assim, Kylix é ferramenta que onera custos.

Na busca por ferramentas visuais que tenham como plataforma o sistema operacional Linux, encontrou-se uma ferramentas que apresenta uma relevância muito importante, tanto quanto o próprio Kylix. O Qt foi escolhido para ser utilizado, por ser uma aplicação compreensiva para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataformas, além de possuir uma ferramenta gráfica chamada QtDesigner que é uma poderosa construtora de *layouts* e *forms* com características GUI, a qual está hábil ao desenvolvimento rápido de interfaces de usuários com alta performance com visual nativo e suportado em várias plataformas(Windows, Linux/Unix, Mac OS X).



Fig. 4.1- Escalabilidade de plataformas para QT

5. Resultados

5.1 Geometrias geradas: esfera, cilindro, trem de pouso, aeronave

Com o Gmsh pôde-se obter geometrias simples como mostrado na Fig. 5.1, geradas com bastante agilidade se comparado ao uso de pacotes comerciais. Levando-se em conta que para atividades acadêmicas esses tipos de geometrias podem ser utilizadas facilmente para realizar simulações.

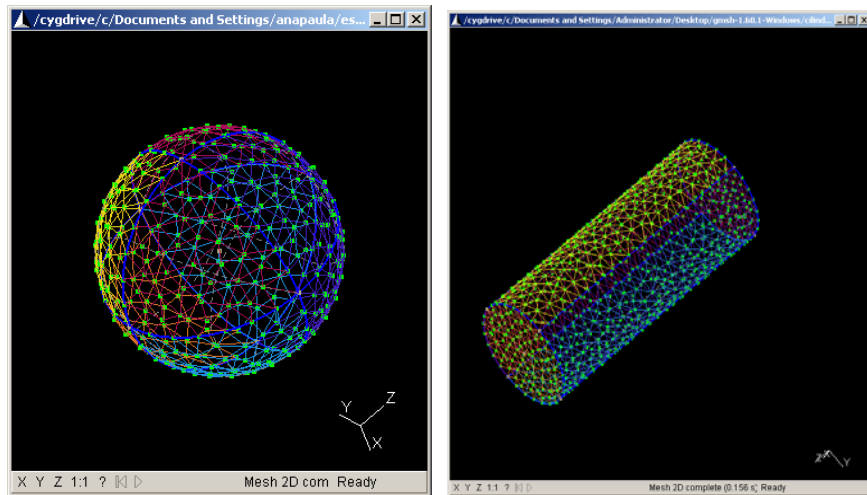


Fig 5.1 – Geometrias simples geradas no Gmsh

Em se tratando de geometrias mais complexas como mostrado na Fig. 5.2 e Fig. 5.3, a tarefa de gerá-las no Gmsh torna-se um pouco mais complicada, por exigir maior destreza e habilidade em criação de figuras 3D. Apesar de ser uma tarefa mais difícil, não é impossível de ser efetuada, porém necessita-se de uma quantidade maior de tempo para obter sucesso.

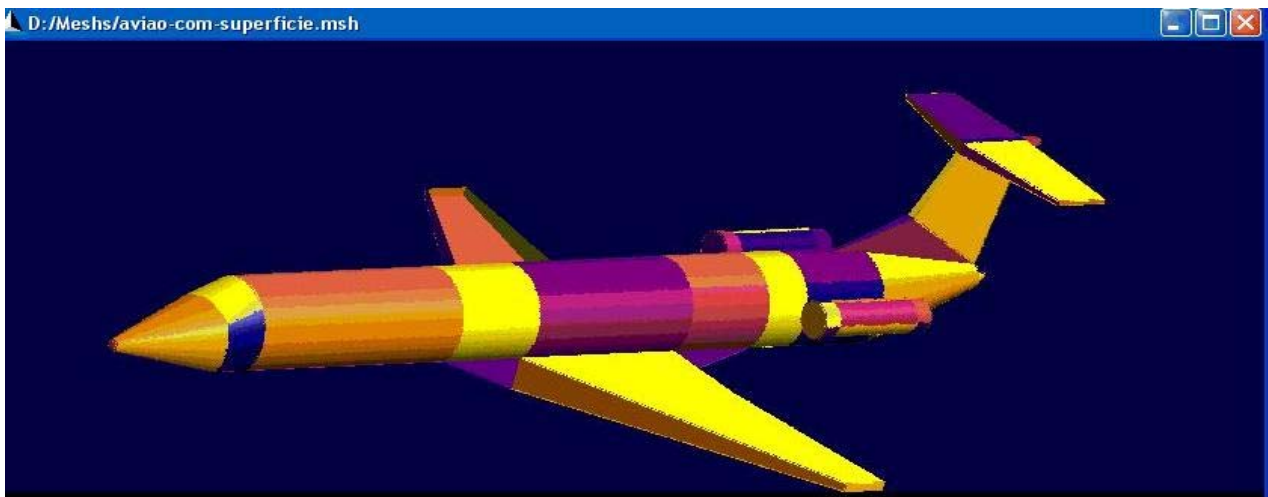


Fig. 5.2 Malha de um avião gerada com Gmsh

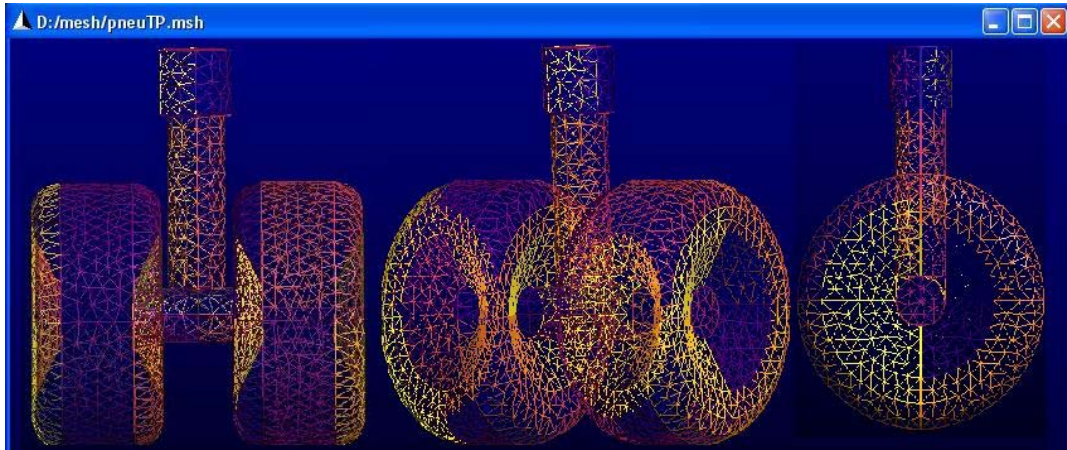


Fig. 5.3 – Malha de um trem de pouso gerada com Gmsh

6. Validando as malhas geradas pelo Gmsh

Para analisar a topologia do escoamento iremos utilizar de alguns parâmetros, efetuando comparações dos resultados ao utilizar o Gmsh para gerar uma esfera e resultados utilizando pacotes comerciais para a mesma geometria. Um importante parâmetro que pode ser empregado é o critério Q , Eq. 6.1 (Jeong e Hussain, 1995), definido da forma:

$$Q = \frac{1}{2} [|\Omega|^2 - |S|^2] > 0 \quad (6.1)$$

, onde S e Ω , são as componentes simétrica e assimétrica do gradiente de u , portanto Q representa o balanço local entre a taxa de deformação e a vorticidade, ou seja, a norma Euclidiana para a qual o tensor vorticidade supera o tensor deformação (Haller, 2005), definidos respectivamente como:

$$\Omega = \frac{1}{2} [\nabla v - (\nabla v)^T] \quad (6.2)$$

$$S = \frac{1}{2} [\nabla v + (\nabla v)^T] \quad (6.3)$$

Portanto o critério Q é uma das várias ferramentas usadas para analisar as propriedades das estruturas turbilhonares. Sua principal vantagem é mostrar as estruturas que adquirem mais peso devido ao efeito rotacional, permitindo visualizar as regiões do escoamento com fortes deformações do fluido.

O parâmetro adimensional número de Reynolds (Re), Eq. 6.4, foi empregado para caracterizar o regime de escoamento do fluido sobre a esfera. O Re foi definido baseado no diâmetro D da esfera na forma:

$$Re_D = \frac{\rho U_\infty D}{\mu} \quad (6.4)$$

onde U_∞ é a velocidade da corrente livre, ρ e μ são a massa específica e a viscosidade dinâmica do fluido, respectivamente.

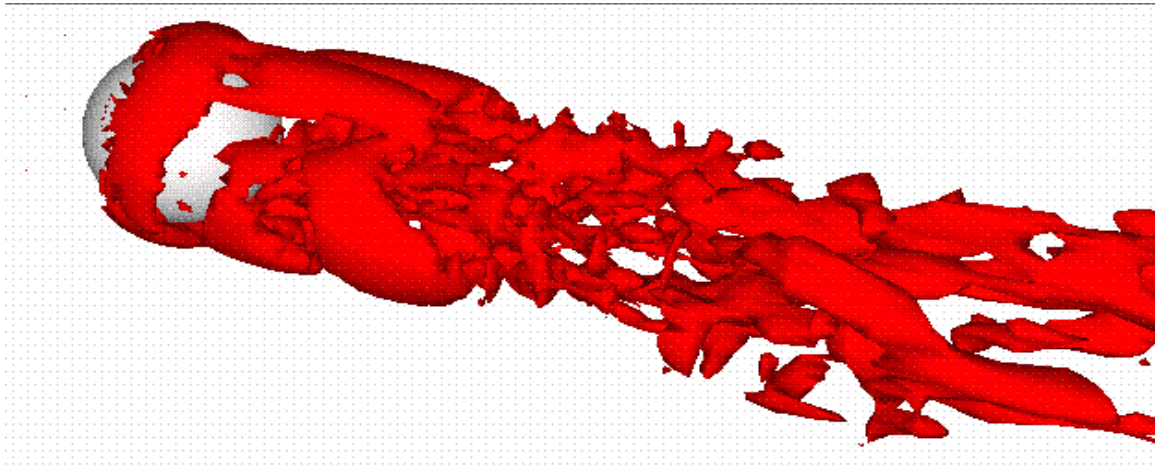


Fig. 6.1 - Detalhe dos vórtices sobre a esfera a $Re = 1.000$ evidenciado pelo critério Q.

A visualização da Fig. 6.1 a cima foi realizada com o isovalor do critério $Q = 10$ para uma esfera gerada pelo Gmsh, sendo esta com elementos da malha quase todos de mesmo tamanho, já a Fig. 6.2 abaixo, foi retirada de Campregher(2005) e gerada com uma malha de elementos triangulares isósceles.

O coeficiente de arrasto é definido abaixo, Eq. 6.5:

$$C_D = \frac{F_x}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.5)$$

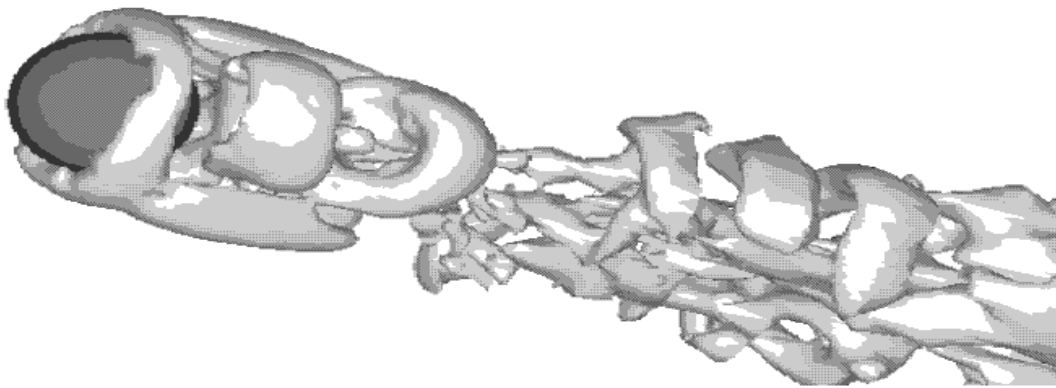


Fig. 6.2 - Detalhe dos vórtices sobre a esfera a $Re = 1.000$ evidenciado pelo critério Q.

sendo F_x a força de arrasto, agindo sobre o corpo imerso (no caso, a esfera), obtida no presente trabalho a partir do Modelo Físico Virtual. É interessante salientar que seu valor é uma resposta direta da metodologia de fronteira imersa, não sendo necessário nenhum procedimento adicional para a avaliação da força de arrasto. Esta é uma forma totalmente diferente de se avaliar a força de arrasto (ou de sustentação) em um corpo, pois, nas metodologias tradicionais, é necessário obtê-la de forma indireta como, por exemplo, a partir da distribuição de pressão e de tensões cisalhantes na superfície do corpo.

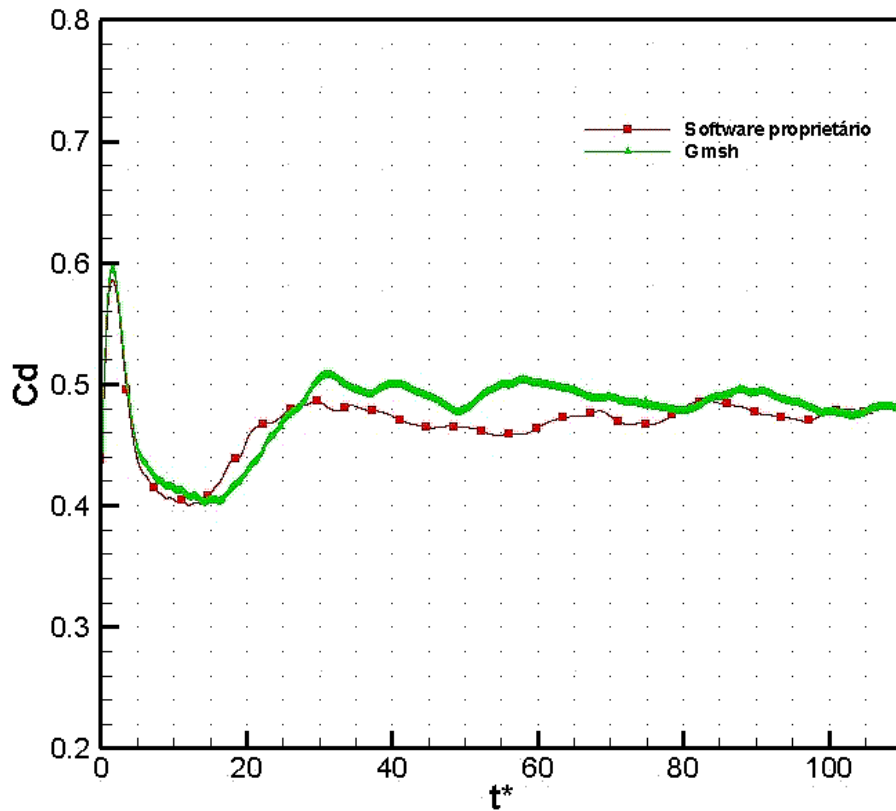


Fig. 6.3- Resultados para o coeficiente de arrasto a Re = 1000

A partir da Fig.6.3 acima, pôde-se perceber que os resultados para o coeficiente de arrasto são bastante satisfatórios para o número de Reynolds simulado, tanto para o pacote comercial quanto para o Gmsh.

Outros parâmetros igualmente importantes, obtidos de maneira semelhante ao coeficiente de arrasto, são o coeficiente de sustentação (lift coefficient – CL) e o coeficiente lateral (side coefficient – CS) avaliados, respectivamente, como Eq. 6.6 e Eq. 6.7:

$$C_L = \frac{F_z}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.6)$$

$$C_S = \frac{F_y}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.7)$$

onde F_y e F_z são a somatório das forças que atuam sobre a esfera nas direções y e z, respectivamente.

Nas Figuras 6.4 e 6.5, respectivamente, pode-se visualizar os resultados de CL e CS para Re = 1.000, indicando o comportamento aleatório da emissão de estruturas turbilhonares à jusante da esfera.

Ele permite visualizar as regiões do escoamento onde se encontram fortes tadas de deformação do fluido.

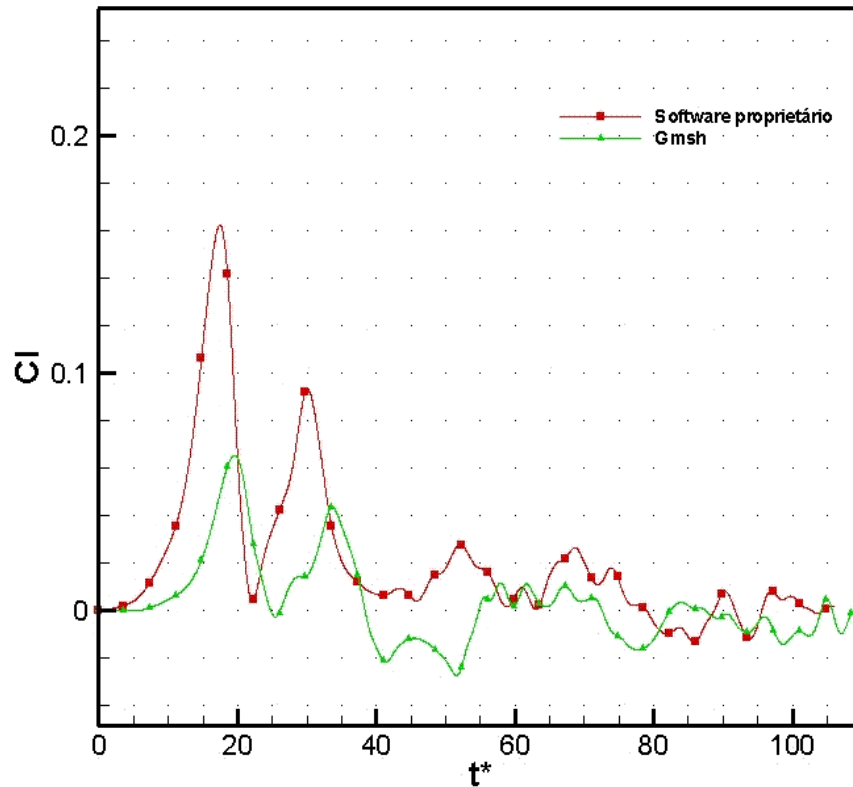


Fig. 6.4 - Coeficiente de sustentação (CL) para Re = 1000

Avaliando estatisticamente, os valores para o coeficiente de sustentação gerada pelos dois softwares geradores de malha são em média compatíveis, porém para o coeficiente lateral podemos notar maiores variações devido ao mesmo ser mais sensível ao tipo de malha gerada, uma vez que o Gmsh não consegue gerar uma malha feita com elementos completamente iguais (triângulos isósceles), havendo nesta malha algumas pequenas variações de tamanho dos triângulos. Contudo podemos verificar que nos cálculos realizados, os valores médios tendem a convergir para um valor semelhante, validando o resultado para esfera gerada pelo Gmsh.

Um outro parâmetro importante a ser calculado sobre a geometria é o L2, que é definida aqui como a raiz quadrada da norma da diferença entre a velocidade do fluido sobre a interface fluido/sólido e da velocidade desta interface (que no caso da esfera estacionária tem valor nulo). Este parâmetro tem por finalidade garantir a condição de não deslizamento sobre a interface, através do campo de força calculado. Como esperado, L2 deve tender a zero à medida que o tempo evoluir, visto que a velocidade do fluido sobre a interface tende ao valor da velocidade da interface. Na Fig. 6.6 mostra-se a evolução temporal do parâmetro L2, para o escoamento sobre a esfera onde o número de Reynolds é igual a 1000.

A pequena diferença existente entre o L2 produzido pela esfera feita com o pacote comercial, e o L2 produzido pela esfera feita no Gmsh, é devido ao fato de que esta última esfera possui aproximadamente duzentos pontos a mais que a do pacote comercial, representando assim uma área superficial um pouco maior que a outra esfera, portanto a velocidade do fluido sobre a interface fluido/sólido é maior na esfera do Gmsh do que na do software proprietário, o que justifica tal diferença.

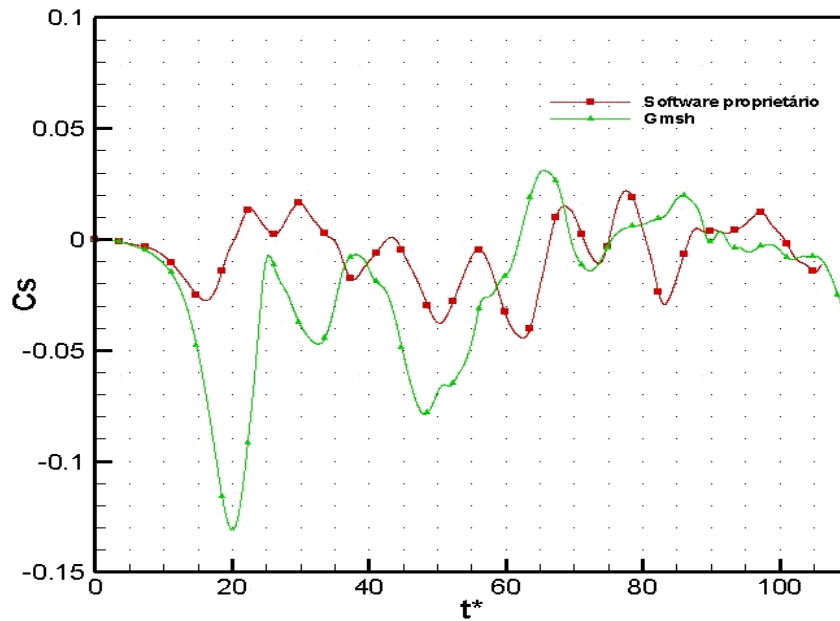


Fig. 6.5 - Coeficiente lateral (C_s) para $Re = 1000$

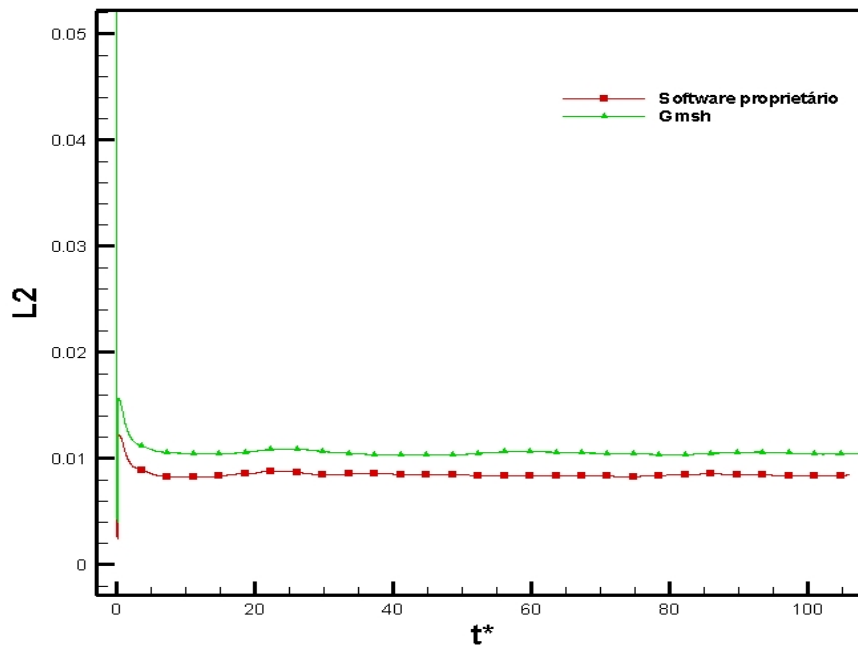


Fig. 6.6 - Evolução temporal do parâmetro L2 para $Re = 1000$

7. Pontos para futuras otimizações de uma interface para CFD

A atual ferramenta disponível no LTCM, além de ser desenvolvida em fortran, é uma ferramenta executável sobre plataforma Linux. Tal característica, faz com que o usuário final da ferramenta tenha que ser hábil com o sistema operacional em questão, para que possa simular qualquer caso com o código disponível.

O objetivo de se criar uma interface gráfica de usuário (GUI), tem como finalidade facilitar o uso da ferramenta para efetuar simulações, de forma que o usuário final possa interagir com o software intuitivamente.

Atualmente com o código do LTCM, levando-se em consideração a fase de pré-processamento, o usuário deve efetuar uma conexão com o cluster (local onde o código está armazenado e será executado em paralelo), feito isso, é

necessário conhecer a estrutura de diretórios do código para localizar os arquivos os quais devem ser editados. Para editar os arquivos deve-se fazer uso de uma ferramenta de edição de arquivo, através do comando “vi” do Linux, por exemplo.

Todas as condições de contorno são configuradas manualmente em um arquivo texto, sendo que algumas das condições não precisam ser modificadas, por tanto não existe a necessidade de que o usuário tenha permissão para isso, muito menos para visualizar o arquivo inteiro.

Além de visualizar um arquivo ao qual não se tem a necessidade de alterar todos os seus parâmetros, como feito com o arquivo de condições de contorno, no atual código, o usuário faz alterações no arquivo do código principal do programa. Tal tipo de operação, é de grande periculosidade, pois o usuário, caso não conheça bem o código do programa, pode fazer modificações de forma a inutilizar o código.

Algumas modificações no arquivo do código principal, têm como função, mostrar o local (pasta de arquivos) ao qual o programa irá buscar informações para poder executar seus cálculos. Uma solução viável para isso, pode ser dada ao inserir uma variável global no programa principal a ser configurada pelo usuário, de modo que o mesmo só indique os caminhos onde irá buscar e salvar informações apenas uma vez, e que essa informação seja enviada ao código principal sem que o usuário tenha contato com o código.

Outro problema a ser modificado na atual estrutura é o arquivo de configuração que faz referência aos dados iniciais a serem lidos e ao local de gravação dos resultados, além de ser um arquivo extenso, o usuário ainda tem que localizar por exemplo, o local ao qual deve acrescentar o nome da malha e o local onde irá gravar as sondas. Semelhante ao problema com o arquivo do código principal, a solução a ser tomada seria a mesma, ou seja, uma janela para indicar os arquivos em questão, porém, ao modificar tanto o arquivo de dados a serem lidos quanto o programa principal o código deve ser recompilado para que se faça valer as modificações.

Para finalizar as modificações a serem feitas a fim de se criar uma interface gráfica, existe a necessidade de criar um campo na interface para determinar quais máquinas irão executar o cálculo da simulação, e é claro um botão de comando para inicializar a execução do código, com as informações do número de máquinas e o arquivo com o nome das máquinas.

Como diferencial para o código existente, há uma necessidade de se criar um pacote na interface responsável por pós processamento, a fim de mostrar a quantificação dos resultados, de forma que a ferramenta possa se tornar o mínimo usual para fins acadêmicos.

De acordo com comparações feitas entre interfaces gráficas de softwares proprietários, optou-se por utilizar um padrão semelhante a um pacote comercial conhecido, já que o mesmo possui um formato organizacional de fácil manuseio, de forma a oferecer facilidade para que o usuário trabalhe com o mesmo, sendo assim teremos uma interface a qual se possa visualizar a malha da geometria a ser simulada, um campo para configurar as condições de contorno mostrado na Fig. 7.1 abaixo e outro para visualização de resultados como mostrado na Fig. 7.2.

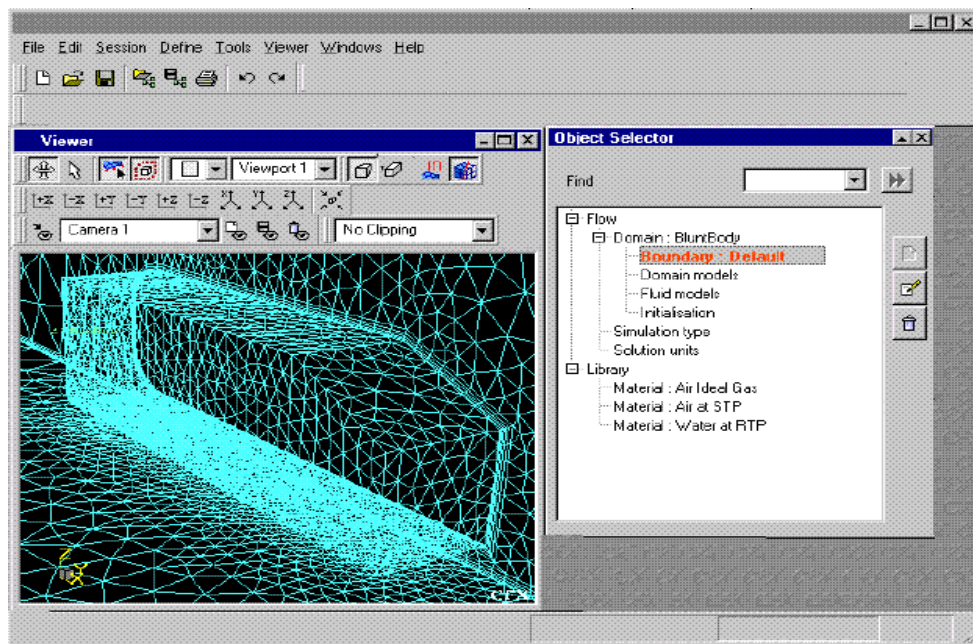


Fig. 7.1 – Interface com campo de visualização da geometria e condições de contorno

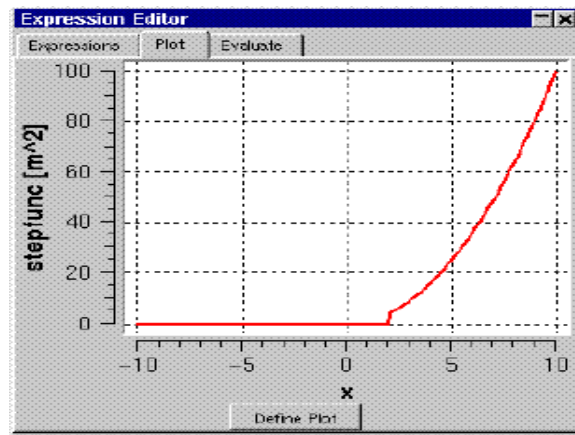


Fig. 7.2 – Exemplo de visualização de resultados

6.1 Referências

- Geuzaine, C., et al. "Gmsh Reference Manual: The documentation for Gmsh 1.60", 2005. Disponível em: <<http://www.geuz.org/gmsh/>> Acessado dia: 20/04/2005;
- Campregher, R. J. "Modelagem Matemática Tridimensional para Problemas de Interação Fluido-Estrutura". Tese de doutorado do Programa de Pós Graduação em Engenharia Mecânica da Universidade Federal de Uberlândia 2005.
- Jeong, J. e Hussain, F., 1995, "On the identification of a vortex", Journal of Fluid Mechanics 285, 69-94.
- Silva, A. L. F. L. "Metodologia Para Modelagem De Escoamentos Sobre Geometrias Complexas: Método da Fronteira Imersa com Modelo Físico Virtual". Tese de doutorado do Programa de Pós Graduação em Engenharia Mecânica da Universidade Federal de Uberlândia 2002.
- White, F., 1991, "Viscous Fluid Flow", McGraw-Hill, New York, USA.

DEVELOPMENT OF A GRAPHIC INTERFACE FOR COMPUTACIONAL FLUID DYNAMIC'S, USING THE METHODOLOGY OF BORDER IMMERSED.

Gustavo Prado de Oliveira

UFU / FEMEC / LTCM
gpoliveira@mecanica.ufu.br

João Marcelo Vedovoto

vedovotojm@yahoo.com
UFU / FEMEC / LTCM

Ana Paula Fernandes

UFU / FEMEC / LTCM
apfernandes@mecanica.ufu.br

Aristeu da Silveira Neto

UFU / FEMEC / LTCM
aristeus@mecanica.ufu.br

Universidade Federal de Uberlândia – Faculdade de Engenharia Mecânica – Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG - CEP: 38400-902

Abstract: Now, the Laboratory of Transfer of Heat and Mass and Dynamics of the Fluids - LTMC/FEMEC/UFU possesses several works for simulations and analysis computational of the dynamics of the fluids, however with stages provided by software proprietors, turning onerous the use of such solutions. Besides, the tool now functional existent, that it makes such calculations, it was developed in the language Fortran, without a graphic interface, besides not allowing a visualization foresaw of the geometries to be worked, what hinders the handling of the same. In that way, in the present work it intends to develop a tool for the construction of geometries and the generation of their meshes of triangular finite elements superficial, being used of free softwares, for subsequent import in codes dedicated the analysis computacional of the dynamics of the fluids, through the border method immersed. Among the tools analyzed for the modelling of the geometry and generation of the mesh, she opted for the use of Gmsh, for being that a free software, functional for the platforms linux and windows, intuitive and of easy learning. Already among the tools analyzed for the development of the graphic Interface, she opted for QT, for being an understanding application for framework development in C++. This includes a library of classes and tools for the development and platform migration, besides being also functional for Windows and Linux. The work in geometries as spheres, cylinders, airfoils, being looked for to use the softwares to work with more complex forms as aircrafts and automobiles.

Keywords: border immersed, generation of meshes, free software