

GERAÇÃO E PRÉ-PROCESSAMENTO DE MODELO PARA ANÁLISE ESTRUTURAL DE SISTEMAS DE CABEÇA DE POÇO SUBMARINO

Carmen N. M. Paz¹ (PEC/LAMCE/COPPE/UFRJ), José L. D. Alves² (PEC/LAMCE/COPPE/UFRJ)

^{1,2} Programa de Engenharia Civil, Universidade Federal do Rio de Janeiro, LAMCE/COPPE CP 68552, CEP: 21944-900, {mena, jalves} @lamce.coppe.ufrj.br

ABSTRACT

Current practice on offshore oil industry, strongly relies in subsea wellhead systems which are the ultimate link between reservoir and floating facilities in deep and ultra-deep water hydrocarbon production scenario. The wellhead system installed at seafloor level provides the structural support for several devices such as the BOB (blow out preventor – a mandatory safety device), the flow-lines and risers that ultimately convey the hydrocarbons to surface. Loading conditions usually consider motions of the floating unit at sea surface. Loading conditions usually consider motions of the floating unit at sea surface, waves and subsurface currents acting on the risers, with in turn transfer a considerable share of these loadings to the connection on top of the wellhead system.

In the present work, we address problems related to model generation and pre-processing concerning a discrete model for this important device, a subsea wellhead system, including discrete cohesive crack interface element for the cement grouting, for the friction connector and the interface of the structural system with the surrounding soil.

From the computational point of view, a special generator is described which departing from a standard FE model, introduces the interface elements where required, changing original topology accordingly. Preprocessing of the generated model also includes mesh partition (METIS) for computation in a message passing (MPI) computational environment. Non-homogeneous, unstructured mesh, including different elements types, and node-equation ordering in each processor are considered, bearing in mind memory the hierarchy of each processor.

1, Elementos Finitos – 2, Geração e particionamento de malhas – 3, Computação paralela de alto desempenho

1. INTRODUÇÃO

Neste trabalho, apresentamos as estratégias de geração e pré-processamento do modelo, implementadas, objetivando numa fase posterior a análise estrutural de um sistema de cabeça de poço submarino realista, em sintonia com os modernos recursos de modelagem e análise computacional.

A evolução desta linha de pesquisa prevê o estudo e simulação do comportamento estrutural de sistemas de cabeça de poço submarino, com vistas à estimativa do nível de tensões no sistema de cabeça de poço e bainha de cimento, contemplando os casos de poços situados em regiões de lâmina d'água profunda e ultraprofunda, em áreas com sedimentos de predominância argilosa, em meios heterogêneos e semi-infinitos.

Os objetivos da pesquisa visam a previsão dos níveis de tensões e esforços nos revestimentos estruturais e bainhas, bem como os deslocamentos do conjunto do sistema de cabeça de poço devido às solicitações em serviço. Pretende-se ao final dessa pesquisa, reunir informações suficientes a respeito do comportamento do sistema estrutural e das diversas interações entre seus componentes, de forma a subsidiar a elaboração de uma ferramenta de cálculo expedito, orientada ao projeto de cabeças de poço submarino.

Os custos envolvidos na operação de perfuração de um poço em águas profundas ultrapassam a cifra de centenas de milhares de dólares por dia, sem falar nos possíveis impactos ambientais advindos de alguma falha acidental nos sistemas. A correta e segura operação de todo o sistema depende sobremaneira das condições de suporte estrutural conferido pelo sistema de cabeça de poço e suas fundações, papel este desempenhado pelos revestimentos – usualmente tubos de 30 a 20 polegadas de diâmetro. Diversas são as configurações e técnicas de instalação desses revestimentos, quer por perfuração ou jateamento, estendendo-se do leito submarino até profundidades da ordem de 60 metros “Mahaidi *et al.* (2004)”. Pode ser empregada cimentação do anular formado entre dois ou mais revestimentos e ainda ao solo adjacente, de forma a assegurar a transferência de solicitações e suporte. O seu comportamento, à primeira vista, assemelha-se ao de uma fundação profunda (uma estaca) em meio heterogêneo.

Os sistemas de cabeça de poço estão sujeitos a diversos tipos de solicitações, tais como, os movimentos da conexão com os *risers* sob a ação das correntes submarinas e os movimentos induzidos pelo comportamento das estruturas flutuantes na superfície, sujeitas a diversas ações ambientais quando em operação “Finnie *et al.* (2004)”. Problemas de desvio da condição (ideal) de verticalidade do sistema por ocasião da perfuração e

instalação dos revestimentos podem amplificar os deslocamentos (devido a efeitos não-lineares geométricos) e eventualmente inviabilizar sua operação. Tal comportamento exige a correta consideração do problema através de modelos tridimensionais adequados ao tratamento, tanto da geometria, quanto da arbitrariedade na direção e orientação dos carregamentos.

Estudos para a modelagem desses sistemas, com vistas ao seu comportamento estrutural, são de grande relevância e vitais para a indústria petrolífera nas atividades de exploração e produção em águas profundas e ultra-profundas.

2. METODOLOGIA

O trabalho está sendo desenvolvido empregando-se como ferramenta principal de análise o Método dos Elementos Finitos para a modelagem do sistema de cabeça de poço em seus detalhes estruturais, conexões e interação com o solo adjacente tomado como um meio heterogêneo, semi-infinito e de resposta não-linear das solicitações.

Os diversos módulos de pré-processamento empregam técnicas especiais de geração do modelo de cada componente. Além do emprego de pacotes comerciais de geração, um programa de geração de malha foi especialmente desenvolvido para a inserção dos elementos de interface onde necessário. Considerando-se o porte do modelo e em função de seu alto detalhamento, os códigos empregam técnicas de processamento de alto desempenho. Nesta fase de pré-processamento considerou também um eficiente balanceamento da carga computacional entre os processadores. Tornou-se necessário um passo adicional para viabilizar a análise do modelo em ambientes de computação paralela, para atingir o objetivo desenvolvemos um código e para o particionamento de malhas de elementos finitos.

Todas as análises previstas nesta linha de pesquisa se darão com o emprego de códigos computacionais desenvolvidas na COPPE, “Alves *et al.* (2006/2007)”, “Coelho *et al.* (2006)”, “Coutinho (2003)”, “Mendes *et al.* (2006)”, “Paz *et al.* (2005)”, considerando as não-linearidades física e geométricas do problema, bem como a interação solo-estrutura e o caráter dinâmico das solicitações.

As solicitações sobre o sistema de cabeça de poço serão determinadas com o emprego de simuladores específicos da resposta das estruturas flutuantes e *risers* às ações ambientais em operação.

Os trabalhos estão sendo desenvolvidos no Laboratório de Métodos Computacionais em Engenharia do Programa de Engenharia Civil da COPPE/UFRJ, utilizando recursos computacionais para o desenvolvimento dos códigos, pré e pós-processamento e visualização científica das análises. Para as análises de grande porte, considerando modelos tridimensionais de alto detalhamento, serão utilizados os recursos computacionais de alto desempenho atualmente disponíveis no Núcleo de Atendimento em Computação de Alto Desempenho (NACAD) da COPPE/UFRJ.

3. DETERMINAÇÃO DAS CARACTERÍSTICAS GEOMÉTRICAS

Foram levantadas as características geométricas e mecânicas de inúmeras fontes, com vistas à elaboração dos modelos discretos de análise. Pesquisou-se tanto a nível bibliográfico, como a nível industrial e comercial, verificou-se toda a gama do produto disponível no mercado, com toda a especificação dos diversos componentes que se constituem nos sistemas de cabeça de poço submarino. Foi adotado um modelo geométrico detalhado na Figura 1.

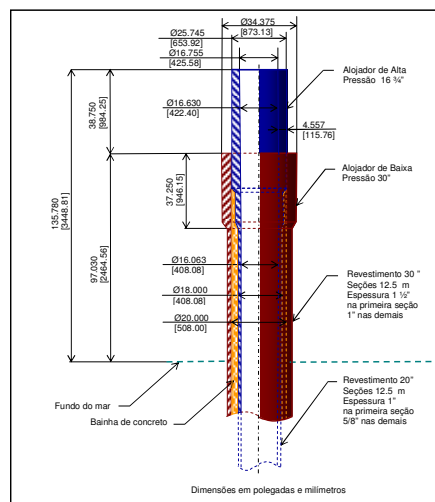


Figura 1 – Sistema de cabeça de poço submarino.

Utilizando o pacotes comerciais de geração estabeleceu-se a geometria do modelo com as especificações dos diferentes componentes, representados na Figura 2, gerou-se um arquivo padrão que estabelece os detalhes das características geométricas de cada componente da cabeça de poço, sendo factível efetuar mudanças, especificando variações na geometria do modelo. A Figura 3 apresenta detalhes da geometria na região próxima aos alojadores.

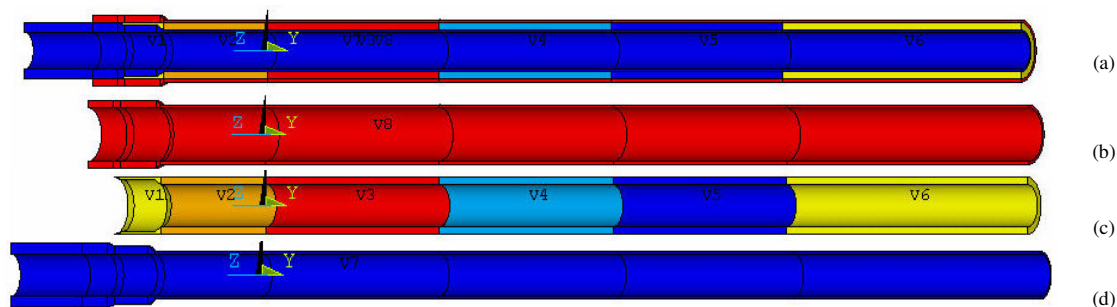


Figura 2 – Geometria do sistema de cabeça de poço submarino.

- a) Geometria do conjunto dos componentes, b) Alojador de baixa pressão e revestimento, c) Bainha de concreto, d) Alojador de alta pressão e revestimento.

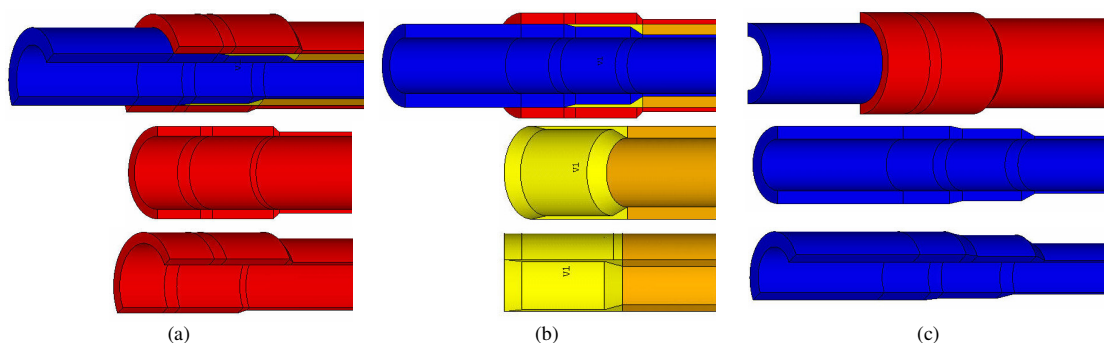


Figura 3 – Detalhes da geometria do sistema de cabeça de poço submarino da região próxima aos alojadores
a) Alojador de baixa pressão, b) Topo da bainha de concreto, c) Alojador de alta pressão.

4. GERAÇÃO DOS MODELOS DISCRETOS - PRÉ-PROCESSAMENTO

Efetuu-se a geração da malha de elementos finitos tetraédricos, como mostrados na Figura 4, especificando o refinamento necessário em cada um dos componentes da cabeça de poço. A Figura 5 apresenta detalhes da malha de tetraedros na região próxima aos alojadores, cabe ressaltar que estas representações só mostram os perímetros dos volumes da malha, devido ao elevado numero de elementos finitos.

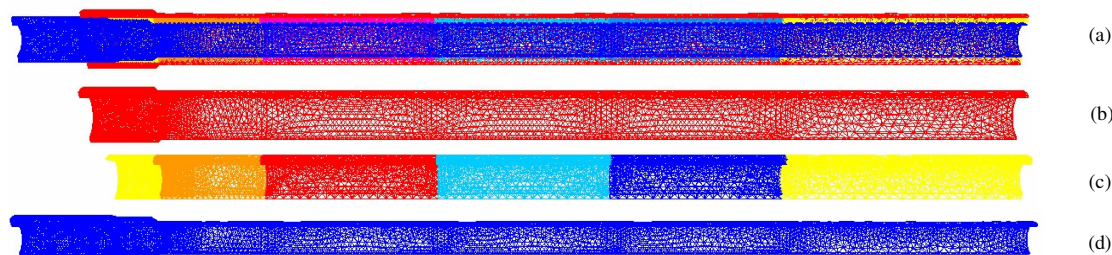


Figura 4 – Malha elementos finitos tetraedros do sistema de cabeça de poço submarino.

- a) Conjunto dos componentes, b) Alojador de baixa pressão e revestimento, c) Bainha de concreto, d) Alojador de alta pressão e revestimento.

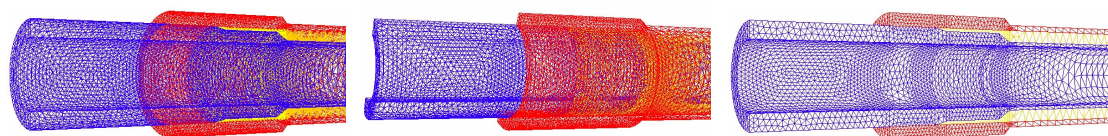


Figura 5 - Detalhes dos perímetros dos volumes da malha de tetraedros na região próxima aos alojadores

4.1 Implementação do gerador de malha de elementos de interface

Entre os geradores de malha comerciais não dispomos de um que tenha a capacidade de introduzir o elemento de interface adotado. Devido ao modelo de análise da pesquisa, surgiu a necessidade de desenvolver um código computacional em linguagem *FORTRAN* para a geração de malha, com o objetivo de introduzir elementos de interface de seis nós de faces triangulares e espessura nula, (Figura 6) numa malha de tetraedros.

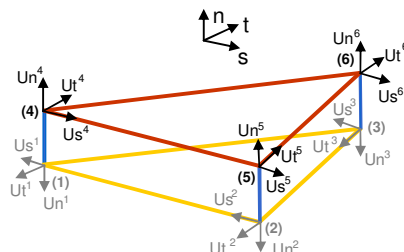


Figura 6 - Elemento de Interface

Em uma primeira instancia os elementos de interface são introduzidos entre as superfícies dos tetraedros, que constituem as fronteiras de contato entre regiões (Figura 7 a - esquerda) que representam os distintos componentes do sistema da cabeça de poço (alojador de alta pressão, bainha de cimento e alojador de baixa pressão, Figura 7b). Em uma segunda instancia os elementos de interface são introduzidos em todas as faces dos tetraedros na região da bainha de concreto ao longo da extensão cimentada (Figura 7c).

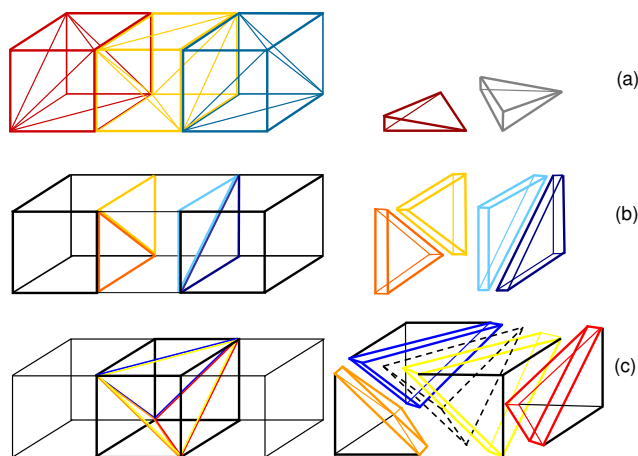


Figura 7 - Gerador de Malha de Elementos de Interface

a) Delimitação de regiões da malhas de tetraedros (esquerda), e elementos de interface com nós colapsados (direita), b) Geração de interface nas superfícies das fronteiras, c) Geração de interface em todas as faces dos tetraedros na bainha.

O gerador de malha elaborado supre satisfatoriamente as nossas necessidades e funciona da seguinte maneira (Quadro1):

Programa principal	• Efetua a leitura de dados da malha de tetraedros. • Estabelece a estrutura de vizinhanças dos elementos tetraedros. • Gera elementos de interface nas fronteiras de contato entre regiões. • Gera elementos de interface em todas as faces do tetraedro na região da bainha de concreto. • Restabelece a estrutura de vizinhanças incluindo os elementos de interface. • Reordena a numeração dos nós da malha de elementos tetraedros e interface. • Gera arquivos de saída da nova malha, para processamento e visualização no <i>Ensignt</i>. • Gera arquivo de saída da nova estrutura de vizinhanças dos elementos tetraedros e interfaces.
Fim do programa	

Quadro 1 - Gerador de malha de elementos de interface para cabeça de poço submarino

Inicialmente gera-se a malha de elementos de interface nas superfícies de fronteira entre regiões, de uma malha de tetraedros (Figuras: 7b; 8a-b, 9c.). O algoritmo (Quadro 2) efetua uma busca para identificar os nós da

face que precisa de elementos interface, percorrendo as incidências dos elementos, e o nó que esta sendo pesquisado e percorrido checando as vizinhanças, para identificar os elementos da fronteira. Esta identificação é efetuada pela diferença no material de cada um dos componentes do sistema da cabeça de poço, predefinida ao criar a malha de tetraedros. Contabiliza-se os elementos que compartilham o nó, mapea-se a posição do nó nas conectividades, mapea-se os elementos aos que pertence o nó. Após estas verificações cria-se o novo nó, atualiza-se a numeração da conectividade no tetraedro, pesquisa-se a estrutura de conectividade das faces do elemento e do vizinho, e finalmente cria-se o elemento de interface, estabelecem-se as conectividades do novo elemento de interface e atualizam-se as vizinhanças.

```

para nó = 1, ate Nnós faça
  se (nó é do domínio) então
    para elem = 1, ate N elem faça
      para incid = 1, ate Nincid faça
        se (nó . = . nó pesquisado) então
          para face = 1, ate Nvizinhos faça
            se (face é da fronteira) então
              se (nó é da fronteira ) então
                Mapea-se os nós que precisam ser duplicados
              fim se
            fim se
          fim para
        fim se
      fim para
    fim se
  fim para
para nó = 1, ate Nnós faça
  se (nó é domínio) então
    para elem = 1, ate Nelem faça
      para incid = 1, ate N incid faça
        se (nó vai ser duplicado) então
          Mapea-se a posição do nó na conectividade
          Mapea-se e contabilizam-se os elementos que concorrem ao nó
        fim se
      fim para
    fim para
    se (nó vai duplicar )então
      para Nónovo = 1, ate Nelem concorrem ao nó faça
        se(nó ainda não foi duplicado) então
          Cria-se nó novo
          Atualiza-se a numeração da conectividade no tetraedro
        senão
          Atualiza-se a numeração nas conectividades na malha de tetraedros
        fim se
      fim para
    fim se
  fim se
para nó = 1, ate Nnós faça
  para face = 1, ate Nvizinhos faça
    se (vizinho não é interface) então
      se ( se pelo menos um nó da face é do domínio) então
        para face vizinha = 1, ate Nvizinhos faça
          Identifica as faces de vizinhança
          Pesquisa a estrutura de conectividade das faces, do elemento e do vizinho
          Cria-se o elemento de interface
          Estabelecem-se as conectividades do novo elemento de interface
        fim para
        Atualizam-se as vizinhanças
      fim se
    fim se
  fim para
fim para

```

Quadro 2 - Algoritmo de geração de elementos de interface nas fronteiras entre regiões

Posteriormente numa segunda instancia gera-se elementos de interface em todas as faces dos elementos tetraedros, na região da bainha de concreto ao longo de toda a extensão cimentada (Figuras: 7c; 8a,c, 9d,). Através do algoritmo de geração (Quadro 3) percorre-se os nós objetivando: mapear os nós da região de interesse; contabilizar os elementos que compartilham o nó; quantificar a necessidade de nós novos; mapear a posição do nó na conectividade; mapear o elemento ao que pertence o nó. Após estas verificações procede-se a criar novos nós, atualizar a numeração nas conectividades dos tetraedros, para finalmente criar o elemento de interface, pesquisando a estrutura de conectividade das faces do elemento e do vizinho, para estabelecer as conectividades do novo elemento de interface e atualizar as vizinhanças.

```

para nó = 1, ate Nnós faça
  se (nó é do domínio) então
    para elem = 1, ate Nelem faça
      para incid = 1, ate Nincid faça
        se (nó =, nó pesquisado) então
          para face = 1, ate Nvizinhos faça
            se (elemento é da baina) então
              se (nó faz vizinhança) então
                Mapea-se os nós da baina
                Mapea-se e contabilizam-se os elementos que concorrem ao nó
                Mapea-se a posição do nó na conectividade.
                Contabiliza-se a necessidade de nós novos.
              fim se
            fim se
          fim para
        fim se
      fim para
    fim para
  fim para
para nó = 1, ate Nnós faça
  se (nó é do domínio) então
    se (nó é da baina) então
      para Nónovo = 1, ate Nnósnovos faça
        Cria-se nós novos
        Atualiza-se a numeração nas conectividades dos tetraedros
      fim para
    fim se
  fim se
fim para
para elem = 1, ate N elem faça
  para face = 1, ate Nvizinhos faça
    se (vizinho não é interface) então
      se (nó é da baina) então
        se (se pelo menos um nó da face é do domínio) então
          para face vizinha = 1, ate Nvizinhos faça
            Identificam-se as faces que fazem vizinhança
            Pesquisa a estrutura de conectividade das faces, do elemento e do vizinho
            Cria-se o elemento de interface
            Estabelecem-se as conectividades do novo elemento de interface
          fim para
          Atualiza as vizinhanças
        fim se
      fim se
    fim se
  fim para
fim para

```

Quadro 3 - Algoritmo para geração de malha de elementos de interface na baina de concreto

Para a compatibilização dos nós da malha, foi necessário criar um artifício em toda a periferia do domínio, onde se inserem os elementos de interface, e onde apenas um ou dois nós da face do tetraedro estão inseridos no domínio. Nesses casos geraram-se elementos de interface com dois ou quatro nós colapsados, conforme ilustrado na Figura 7a (direita). Estes elementos são implementados através de um artifício que permite utilizar o mesmo elemento de 6 nós, duplicando se a numeração dos nós colapsados que fazem interface com a malha fora do domínio.

A geração de malha de elementos finitos de tetraedros e interfaces é efetuada em módulos diferentes, introduzindo no processo uma numeração arbitraria dos nós. Visando a otimização dos dados no acesso a memória durante o processamento, surgiu necessidade de reordenar toda numeração da malha, para tal efeito implementamos em linguagem *FORTRAN* o algoritmo Reverse Cuthill-Mckee “Cuthill (1972) “. Onde a reordenação é efetua-se a partir dos grafos nodais da malha de tetraedros e interfaces (Figura 11c). Mostrado esquematicamente a seguir (Quadro 4):

Reordenador Reverse Cuthill-McKee

1. Estabelece-se a estrutura de grafos nodais: Para a malha de elementos finitos de tetraedros e interfaces, armazenam-se os grafos nodais da malha no vetor r e atribui-se r ao vetor x_i .
2. Núcleo principal: Para $i = 1$ ate N faça. Através da estrutura de vizinhança dos grafos nodais x_i estabelece-se a estratégia de renumeração em ordem crescente.
3. Reordenação da numeração dos nós: No reverse Cuthill-McKee o reordenamento é dado no vetor $y_1, \dots, y_N$ onde o y_i é x_N .
 $i+1$ Para $i = 1$ ate N faça.

Quadro 4 – Pseudo código do Reverse Cuthill-McKee.

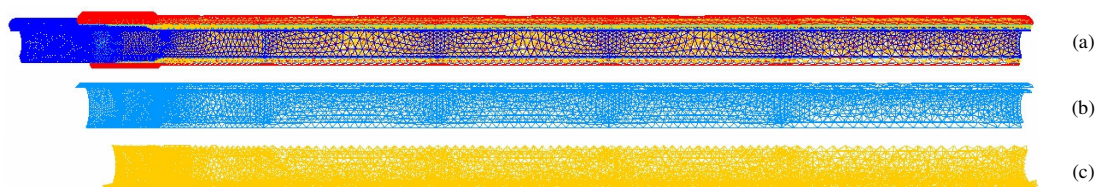


Figura 8 – Geração da malha de elementos de interface no sistema de cabeça de poço submarino
a) Malha de tetraedros e interfaces, b) Geração de malha de interface nas superfícies entre fronteiras, c) Geração de malha de interface em todas as faces dos tetraedros na região da bainha.

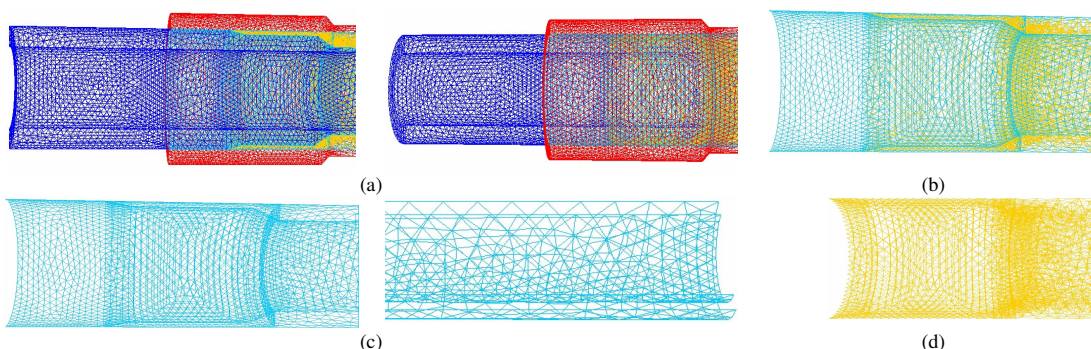


Figura 9 - Detalhes da malha de elementos finitos de tetraedros e interface.
a) Malha de tetraedros com interfaces nas superfícies entre fronteiras, e em todas as faces do tetraedro na bainha, b) Malha de interfaces entre fronteiras (azul) e na bainha (amarelo), c) Topo e base da malha de interface entre fronteiras, d) Malha de interface na bainha.

4.2 Implementação do particionamento da malha de elementos finitos

O presente trabalho está sendo desenvolvido utilizando computação de alto desempenho com processamento em paralelo, sendo assim a premissa inicial é levar em conta, o tipo da arquitetura, o tipo de memória utilizada no ambiente, e tipo de comunicação dos processadores, por serem fatores determinantes na maneira de se escrever um programa paralelo.

A COPPE/NACAD conta com equipamentos com memória distribuída com múltiplos processadores operam independentemente, sendo que cada um possui sua própria memória; Os dados são compartilhados através de uma rede de comunicação, utilizando-se "*Message-Passing*"; onde todo paralelismo é explícito e o programador é responsável em identificar e implementar: a sincronização das tarefas; a decomposição do algoritmo, ou dos dados, em "particionamentos"; a distribuição dos particionamentos por entre os diferentes processadores para que trabalhem simultaneamente; a coordenação do trabalho e a comunicação entre esses processadores.

Utilizamos o *MPI* "*Message Passing Interface*" que é uma biblioteca de "*Message-Passing*", desenvolvida para ser padrão em ambientes de memória distribuída, em computação paralela. A plataforma alvo para o *MPI*, são ambientes de memória distribuída, máquinas paralelas massivas, "*clusters*" de estações de trabalho e redes heterogêneas. O *MPI* permite implementação com: alta escalabilidade; pequena restrição de *hardware*; particionamento, balanceamento e distribuição de tarefas.

Para se decompor um programa em tarefas, visando a execução em paralelo, é necessário se ter a idéia da decomposição funcional e da decomposição de domínio. Na decomposição funcional, o problema é decomposto em diferentes tarefas que serão distribuídas por entre múltiplos processadores para execução simultânea onde cada tarefa será um programa diferente. Na decomposição de domínio os dados são decompostos em grupos que serão distribuídos por entre múltiplos processadores que executarão, simultaneamente, um mesmo programa como se da no caso da resolução de cálculos utilizando elementos finitos.

Um passo adicional é necessário para viabilizar a análise nesse ambiente computacional para o particionamento do modelo. Esta fase de pré-processamento é necessária para um eficiente balanceamento da carga computacional entre os processadores. Tornou-se necessário utilizar o particionador de malha de elementos finitos *METIS 4.0* que é um código que proporciona técnicas consagradas de particionamento. (de distribuição gratuita) desenvolvido em linguagem C, pela universidade de Minnesota, no departamento de ciência da computação. O *METIS* é baseado em métodos de minimização de corte de arestas e executa: particionamento de grafos e malhas (Figura 10); reduz largura de banda de matrizes esparsas; produz partições com balanceamento de carga. Possui versão em paralelo *PARMETIS*.

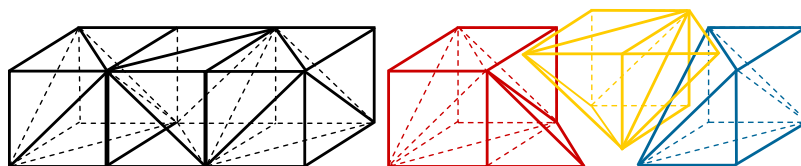


Figura 10 – Ilustração do particionamento da malha de tetraedros com *METIS*.

No presente estudo, não foi possível a utilização direta do *METIS* porque, a nossa malha é composta de elementos tetraedros e interfaces (Figura 11-a). Tornou-se necessário efetuar implementação de um código de interface em linguagem *FORTRAN*, objetivando a criação do grafo dual a nível dos elementos tetraedros e interfaces, (Figura 11-b) ponderando os vértices do grafo com pesos diferentes para elementos tetraedros e interfaces. Viabilizando assim a escalção das rotinas correspondentes da biblioteca do *METIS* (Quadro 5).

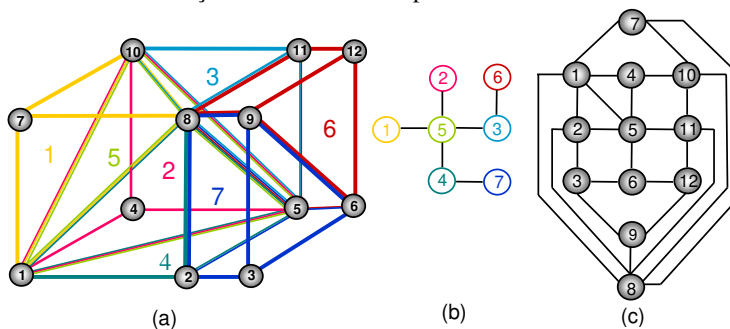


Figura 11 – Malha de elementos Finitos e grafos

a) Malha não estruturada de elementos finitos de tetraedros e interfaces, b) Grafo dual (elementos) da malha de tetraedros e interfaces, c) Grafo nodal da malha de tetraedros e interfaces.

Programa principal	• Requisita as bibliotecas do <i>METIS</i>. • Efetua a leitura de dados da malha de elementos tetraedros e interfaces. • Efetua a leitura da estrutura de vizinhanças. • Cria o grafo dual (elementos) da malha. Ponderando os vértices do grafo com pesos diferentes para elementos tetraedros e interfaces. • Escala a rotina <i>METIS_PartGraphKway</i> e particiona a malha. • Mapeia os nós que delimitam as regiões particionadas. • Gera arquivos da malha particionada, para visualização no <i>Ensign</i> e processamento.
Fim do programa	

Quadro 5 - Particionador de malha de elementos tetraedros e interfaces

A Figura 12 ilustra a partição da malha de elementos finitos com tetraedros e interfaces, no sistema de cabeça de poço submarino, apresentamos o particionamento em 32 partes com mapeamento dos nós que delimitam as regiões particionadas. O gráfico 1 apresenta a distribuição de elementos tetraedros e interfaces por partição.

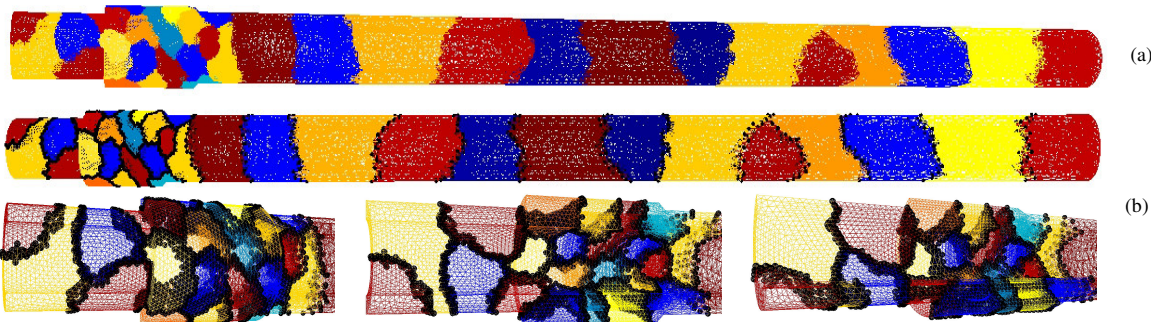


Figura 12 – Particionamento da malha de elementos tetraedros e interfaces com mapeamento dos nós que delimitam as regiões particionadas.

a) Particionamento em 32 partes, b) Mapeamento dos nós que delimitam as partições (preto).

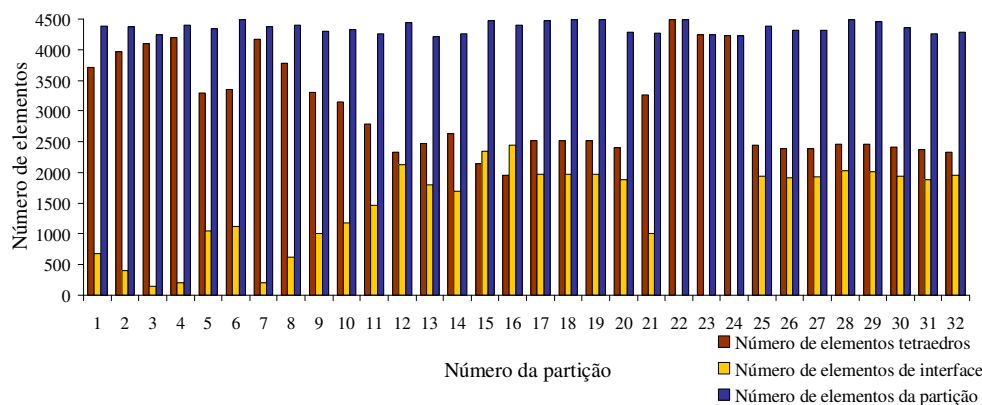


Gráfico 1. Distribuição do número de elementos do particionamento da malha.

A Tabela 1 lista os dados gerais da malha, a Figura 13 identifica os nós para estabelecer as condições de contorno e de carregamento do sistema de cabeça de poço submarino.

Tabela 1. Dados gerais da malha

Nº de total elementos	Nº de tetraedros	Nº de interfaces	Nº de nós	Nº de particionamentos da malha	Nº de nós que delimitam as partições
139563	96739	42824	94613	32	6329

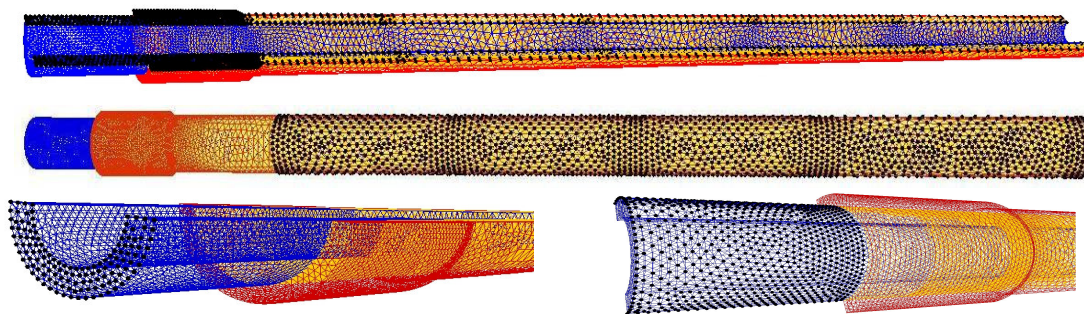


Figura 13 – Identificação dos nós (preto) para estabelecer condições de contorno e de carregamento do problema.

5. CONCLUSÃO

Sistemas de cabeça de poço submarino constituem-se, em última análise, no elo vital entre a jazida petrolífera e as sondas flutuantes de perfuração na superfície, em águas profundas e ultra-profundas. O sistema de cabeça de poço dá suporte estrutural aos sistemas de segurança (*BOP – blow out preventer*), cuja função é a de impedir o fluxo descontrolado dos fluidos da formação do reservatório para a superfície. Altos custos estão envolvidos na operação de perfuração de um poço em águas profundas, sem falar nos possíveis impactos ambientais advindos de alguma falha acidental nos sistemas. Estudos para a modelagem desses sistemas, com vistas ao seu comportamento estrutural, são de grande relevância e vitais para a indústria petrolífera nas atividades de exploração e produção em águas profundas e ultra-profundas.

Nesse trabalho, apresentamos a geração e pré-processamento de modelo, viabilizando a estrutura de dados do problema considerando os principais aspectos para a análise estrutural de um sistema de cabeça de poço submarino realista, em sintonia com os modernos recursos de modelagem e análise computacional. Utilizamos como ferramenta principal o Método dos Elementos Finitos para a geração do modelo levando em conta detalhes estruturais, conexões e interação com o solo adjacente. Os sistemas estruturais de cabeça de poço submarino utilizados na prática corrente incluem diversos componentes: alojadores, revestimentos em aço, compreendendo encaixes ajustados sob tensão e cimentados com a injeção de pasta cimentícia.

Foram desenvolvidos diversos módulos de pré-processamento, empregando técnicas especiais de geração do modelo de cada componente para sua posterior integração e especificação das interfaces. Além do

emprego de pacotes comerciais de geração, um programa foi especialmente desenvolvido para a inserção dos elementos de interface onde necessário. Considerando-se ainda o porte dos modelos em função de seu alto detalhamento, a análise por sua vez exige recursos computacionais especiais. O programa foi concebido tendo-se em vista o poder de processamento de alto desempenho hoje disponível em ambientes de memória distribuída, típico de “clusters”, incluindo as primitivas MPI “*Message Passing Interface*”.

Tornou-se necessário um passo adicional para viabilizar a análise nesse ambiente computacional, que é o particionamento do modelo. Esta fase de pré-processamento é necessária para um eficiente balanceamento da carga computacional entre os processadores. Para isso os modelos foram submetidos a um código especialmente desenvolvido para viabilizar o emprego da biblioteca *METIS*, que proporciona técnicas consagradas de particionamento de malhas de elementos finitos.

6. AGRADECIMENTOS

Este trabalho vem sendo parcialmente apoiado pelo projeto CTPETRO/CNPq 50.5184/2004-4, e desenvolvido na COPPE/UFRJ, no Laboratório de Métodos Computacionais em Engenharia - LAMCE do Programa de Engenharia Civil, associado ao Núcleo de Atendimento em Computação de Alto Desempenho – NACAD.

7. REFERÊNCIAS

- ALVES, J.L.D. and SILVA, E.F.; Performance Assessment for a Family of Continuum-based Shell Elements for Nonlinear Finite Element Analysis in Shared-memory Systems. **VECPAR 2006** - 7th International Meeting on High Performance Computing for Computational Science, Rio de Janeiro, 2006.
- ALVES, J.L.D., COELHO, L. C., BAUD P., GUEVARA J.R., N. O., and WONG., T-F., Constitutive Modeling of porous carbonate rock on wellbore stability analysis, **XXVIII CILAMCE 2007** – Iberian Latin American Congress on Computational Methods in Engineering. Porto Portugal, 13,15 June 2007.
- COELHO, L. C., SOARES, A. C., EBECKEN, N. F., ALVES, J. L. D., LANDAU, L., Modeling mechanical behaviour of limestone under reservoir conditions. **International Journal for Numerical and Analytical Methods in Geomechanics**, v. 30, p. 1477-1500, 2006.
- COUTINHO, A.L.G.A., MARTINS, M.A.D., SYDENSTRICKER, R.M., ALVES, J. L. D. and LANDAU, L. Simple Zero Thickness Kinematically Consistent Interface Elements. **Computers and Geotechnics**, 30, Issue 5, pp. 347-374, 11 de junho de 2003.
- CUTHILL, E, Several strategies for reducing the bandwidth of matrices, **Sparse Matrices and Their Applications** In D.J. Rose and R.A. Willoughby (Eds.), Plenum Press, New York, 1972 .
- DOGARA J., FOSTER I., FOX G., GROPP W., KENNEDY K., TORCZON L., and WRITE A., **Sourcebook of parallel computing**. San Francisco C.A. Morgan Kaufmann publishers, 2003. 842 p.
- FINNIE, I. M. S., and MORGAN, N. Torsional Loading of Subsea Structures. **Proc.of ISOPE 2004** Vol 1 pp 326, Toluse, France, 2004.
- LALLIER, E. B., ROSS, C. L., and DAVIS C. Protective Deployment of Subsea Equipment. **OTC015089** -I. 2003.
- LEHLE, D. BILBERRY - Optimizing Marginal Subsea Well Developments Through Application Of Intelligent Completions. **OTC015193**. 2003.
- MAHAIDI, R. A. -, and GRUNDY, P. Finite Element Analysis of Concrete Plugs Embedded in Tubular Steel Piles Abolghasem Nezamian. **Proc. of ISOPE 2004** Vol 1 pp 186, Toulouse, France, 2004.
- MENDES, R. B, ALVES, J. L. D. , SILVA, C. E. d. , Simulation of torpedo pile launching by discrete element method. In: SIMMEC 2006, 2006, Araxá, MG. Anais do VII Simpósio Mineiro de Mecânica Computacional, 2006.
- PAZ, C. N. M., MARTHA, L. F., ALVES, J. L.D., FAIRBAIRN, E. M.R., EBECKEN, N. F.F., and COUTINHO, A. L.G.A., Parallel Implementation for probabilistic Analysis of 3D Discrete Cracking Concrete. **Lecture Notes in Computer Science**, ISBN 3-5540-00852-7, 2565, 79-93. 2003.
- PAZ, C. N. M., ALVES, J. L.D., and EBECKEN, N. F.F., Assessment of computational performance for a vector parallel implementation: 3D probabilistic model discrete cracking in concrete. **Computer and Concrete**. Korea, v. 2, p. 345 - 366, 01 out., 2005.
- SYDENSTRICKER, R.M., MARTINS, M.A.D., COUTINHO, A.L.G.A., and ALVES, J.L.D. Edge-Based Interface Elements for Solution of Three-Dimensional Geomechanical Problems. **Lecture Notes in Computer Sciences**, Springer-Verlag, Vol. 2565, pp. 53-65, 01 de abril de 2003.