

Universidade Federal de Pernambuco  
Departamento de Engenharia Mecânica  
Curso de Engenharia Naval

Ícaro Aragão Fonseca

**Compatibilização de Códigos Computacionais  
para a Otimização da Configuração de *Moon  
Pool* sob o Ponto de Vista da Operabilidade**

Recife

2016

Ícaro Aragão Fonseca

**Compatibilização de Códigos Computacionais para a  
Otimização da Configuração de *Moon Pool* sob o Ponto  
de Vista da Operabilidade**

Trabalho apresentado ao Curso de Graduação em Engenharia Naval, da Universidade Federal de Pernambuco, como requisito final para a obtenção do grau de Engenheiro Naval.

Orientadora: Prof.<sup>a</sup> MSc. Paula Suemy Arruda Michima.

Recife

2016

Catálogo na fonte  
Bibliotecária Margareth Malta, CRB-4 / 1198

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| F676c | <p data-bbox="395 1279 660 1305">Fonseca, Ícaro Aragão.</p> <p data-bbox="395 1312 1366 1406">Compatibilização de códigos computacionais para a otimização da configuração de Moon Pool sob o ponto de vista da operabilidade / Ícaro Aragão Fonseca. – 2016. 92 folhas, il., gráfs., tabs.</p> <p data-bbox="451 1442 1102 1469">Orientadora: Profa. MSc. Paula Suemy Arruda Michima.</p> <p data-bbox="403 1505 1278 1599">TCC (Graduação) – Universidade Federal de Pernambuco. CTG. Departamento de Engenharia Naval, 2016.<br/>Inclui Referências, Apêndices e Anexos.</p> <p data-bbox="403 1635 1278 1729">1. Engenharia Naval. 2. Moon Pool. 3. Operabilidade. 4. Otimização. 5. Algoritmo genético. 6. Malhas computacionais. I. Michima, Paula Suemy Arruda. (Orientadora). II. Título.</p> <p data-bbox="1046 1805 1117 1832">UFPE</p> <p data-bbox="451 1868 665 1895">623 CDD (22. ed.)</p> <p data-bbox="1051 1868 1246 1895">BCTG/2016-294</p> |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|



**DEMEC**  
**Departamento de Engenharia Mecânica**  
**Centro de Tecnologia e Geociências - CTG**

**Ata da reunião**

**Trabalho de Conclusão de Curso**

**Compatibilização de códigos computacionais para a otimização da  
configuração de Moon Pool sob o ponto de vista da operabilidade**

O candidato **ÍCARO ARAGÃO FONSECA** apresentou seu trabalho de Conclusão de Curso – TCC em **23/09/2016** perante a banca composta pelos professores **MSc Paula Suemy Arruda Michima (orientador)**; **PhD Ana Rosa Mendes Primo**, e **PhD Cesar Salhua Moreno** e **MSc Marco Antonio Ladislau Petkovic**. A apresentação do trabalho sobre o tema considerado teve duração de 30 minutos. Durante a apresentação, o candidato mostrou a relevância de seu trabalho e discorreu sobre o tema de forma clara e objetiva. Depois da apresentação, a banca passou a fazer comentários e sugestões sobre o trabalho, finalizando com algumas arguições ao candidato. Em separado a banca se uniu e deliberou pela aprovação do candidato, atribuindo-lhe a média oito e meio ( 8.5 ) e julgando-o apto a concluir seu curso de Engenharia Naval junto ao Departamento de Engenharia Mecânica da Universidade Federal de Pernambuco.

- |                                         |                     |
|-----------------------------------------|---------------------|
| 1. MSc. Paula Suemy Arruda Michima      | nota ( <u>8.5</u> ) |
| 2. PhD. Ana Rosa Mendes Primo           | nota ( <u>8.5</u> ) |
| 3. PhD. Cesar Salhua Moreno             | nota ( <u>8.5</u> ) |
| 4. MSc. Marco Antonio Ladislau Petkovic | nota ( <u>8.5</u> ) |

Recife 23 de setembro de 2016

**Juraci C. de C. Nobrega**  
**DEMEC/UFPE**  
**Coordenador de TCC**  
Prof. Dr. Juraci Carlos de Castro Nobrega  
**Coordenador de TCC 2**

*Dedico este trabalho à minha família,  
em especial à minha avó, Severina.*

# Agradecimentos

Agradeço à minha orientadora Paula Michima pela confiança e liberdade criativa que me foram delegadas ao longo do desenvolvimento deste trabalho.

Agradeço ao professor Hiroshi Kawabe pelo grande auxílio na utilização de seu código e pela inspiração transmitida em suas aulas como pesquisador visitante na UFPE.

Agradeço à Petrobras, à Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP), à Financiadora de Estudos e Projetos (FINEP) e ao Ministério da Ciência, Tecnologia, Inovações e Comunicações (MCTIC) pelo apoio financeiro oferecido por meio do Programa de Recursos Humanos da ANP para o Setor Petróleo e Gás (PRH-ANP/MCTIC), durante o período de realização deste curso de graduação.

Agradeço à equipe de desenvolvimento da suíte  $\text{abnT}_{\text{E}}\text{X}_2$  pelo fornecimento das ferramentas utilizadas na editoração deste documento.

# Resumo

Navios *drillship* são embarcações especializadas na perfuração *offshore*, seja para a exploração de poços de petróleo e gás ou para fins de pesquisa científica. A operabilidade de um navio desse tipo depende da segurança da tripulação, da integridade do navio e equipamentos, e da manutenção da localização e posicionamento em diferentes contextos ambientais.

O fator de maior influência no desempenho de um navio de perfuração com relação à operabilidade e segurança é a oscilação da água interna ao *moon pool*. Esta, por sua vez, varia com o formato e as dimensões da abertura para um mesmo estado de mar e atividade do navio. Estudam-se duas situações distintas: estacionário operando em ondas, ou em trânsito em águas calmas. Dado um casco previamente definido, o objetivo final do projeto a que este trabalho pertence é encontrar o *moon pool* que garanta o desempenho ótimo nos termos propostos, para um ou vários estados de mar característicos da região de operação.

Em estágios anteriores desse projeto, foi desenvolvido um algoritmo em Fortran que calcula a resposta hidrodinâmica do casco no domínio da frequência. Em um trabalho anterior, também foi desenvolvido, em MATLAB, um algoritmo genético que varia os parâmetros geométricos que definem o *moon pool* e outro que acopla, de forma limitada, esse *moon pool* à malha de um casco série 60. Neste trabalho, uma melhoria na integração entre as malhas do *moon pool* e do casco é apresentada, e esses módulos são combinados de forma a gerar automaticamente os *moon pools* definidos pelo algoritmo genético. O comportamento hidrodinâmico de cada configuração de *moon pool* é então testado pelo algoritmo de simulação hidrodinâmica, chamado com um comando de sistema pelo código do MATLAB. Os resultados são processados e convertidos em um índice de desempenho em operabilidade para cada indivíduo, que é utilizado como critério de avaliação. O algoritmo genético prossegue com as iterações até atingir a convergência da otimização, para várias dimensões de *moon pool*. Finalmente, o código escolhe o indivíduo ótimo dentre todos os analisados.

**Palavras-chave:** *Moon pool*. Operabilidade. Otimização. Algoritmo genético. Malhas computacionais.

# Abstract

Drillships are vessels specialized in offshore drilling, either for exploitation of oil and gas wells or for scientific research purposes. The operability of such a vessel depends on the crew safety, equipment and vessel integrity and the maintenance of location and positioning in different environmental contexts.

The most influential factor in the performance of a drillship regarding operability and safety is the oscillation of the water inside the moon pool. This, in turn, varies with the shape and dimensions of the opening for a given sea state and ship activity. Two distinct situations are studied: stationary operating in waves, or in transit in calm waters. Given a predetermined hull, the final objective of the project to which this work belongs is to find the moon pool that ensures optimum performance on the proposed terms, for one or various characteristic sea states of the operating region.

In previous stages of that project to which this work belongs, an algorithm that calculates hydrodynamic response of the hull in frequency domain was developed in Fortran. In a previous work, a genetic algorithm was also developed in MATLAB to vary the geometric parameters which define the moon pool, and another one to couple, in a limited manner, that moon pool to the grid of a series 60 hull. In this work, an improvement in the integration between the moonpool and hull grids is presented, and those modules are combined to automatically generate the moon pools defined by the genetic algorithm. Each moon pool is then tested by the hydrodynamic simulation algorithm, called with a system command by the MATLAB code. The results are processed and converted into an operability index for each individual, which is used as evaluation criterion. The genetic algorithm continues to iterate until it achieves the optimization convergence, for various moon pool dimensions. Finally, the code chooses the optimal individual among all evaluated.

**Keywords:** Moon pool. Operability. Optimization. Genetic algorithm. Computational grids.

# Lista de ilustrações

|                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------|----|
| Figura 1 – Modelo 3D de navio <i>drillship</i> . . . . .                                                  | 15 |
| Figura 2 – Formação de vórtices por separação do fluxo. . . . .                                           | 20 |
| Figura 3 – Excitação devido à separação do fluxo para o caso do navio em trânsito. . . . .                | 21 |
| Figura 4 – Excitação devido à incidência de ondas para o caso do navio estacionário. . . . .              | 22 |
| Figura 5 – Exemplos de apêndices de <i>moon pool</i> . . . . .                                            | 22 |
| Figura 6 – Gráfico de altura significativa aceitável para operação. . . . .                               | 27 |
| Figura 7 – Gráfico de região segura de operação. . . . .                                                  | 28 |
| Figura 8 – Parametrização do <i>moon pool</i> construído pelo código de acoplamento. . . . .              | 29 |
| Figura 9 – Convenção de coordenadas adotada pelo código de simulação. . . . .                             | 30 |
| Figura 10 – Ferramentas computacionais provenientes de trabalhos anteriores. . . . .                      | 33 |
| Figura 11 – Modelo de integração proposto. . . . .                                                        | 34 |
| Figura 12 – Fluxograma da ferramenta de geração de malha como rotina principal. . . . .                   | 35 |
| Figura 13 – Fluxograma da ferramenta de geração de malha como função. . . . .                             | 35 |
| Figura 14 – Estrutura das pastas de registro da otimização. . . . .                                       | 36 |
| Figura 15 – Painéis internos ao <i>moon pool</i> antes da modificação. . . . .                            | 38 |
| Figura 16 – Painéis internos ao <i>moon pool</i> após a modificação. . . . .                              | 39 |
| Figura 17 – Painéis de interface do <i>moon pool</i> com o casco antes da modificação. . . . .            | 39 |
| Figura 18 – Painéis de interface do <i>moon pool</i> com o casco após a modificação. . . . .              | 40 |
| Figura 19 – Quina do <i>moon pool</i> antes do ajuste introduzido. . . . .                                | 41 |
| Figura 20 – Quina do <i>moon pool</i> após o ajuste introduzido. . . . .                                  | 41 |
| Figura 21 – Convenções adotadas na malha do casco de entrada no código MATLAB. . . . .                    | 42 |
| Figura 22 – Convenções adotadas na malha do casco de entrada na análise hidrodinâmica. . . . .            | 42 |
| Figura 23 – Reestruturação da malha de entrada da análise hidrodinâmica. . . . .                          | 43 |
| Figura 24 – <i>Moon pool</i> com refinamento de onze divisões em sua seção prismática. . . . .            | 43 |
| Figura 25 – Comparação entre um espectro Pierson-Moskowitz e um espectro JONSWAP. . . . .                 | 45 |
| Figura 26 – Convenção direcional dos ângulos de incidência de onda (proa para a direita). . . . .         | 45 |
| Figura 27 – Malha do casco utilizado para a otimização (proa para a esquerda). . . . .                    | 47 |
| Figura 28 – Exemplos de <i>moon pools</i> analisados na segunda otimização (proa para a direita). . . . . | 49 |
| Figura 29 – Histórico das notas de operabilidade dos campeões da primeira execução. . . . .               | 50 |

|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| Figura 30 – Visualização das notas dos campeões da primeira execução. . . . .                 | 50 |
| Figura 31 – Malha do <i>moon pool</i> ótimo da primeira execução. . . . .                     | 52 |
| Figura 32 – Gráfico de operabilidade do <i>moon pool</i> ótimo da primeira execução.          | 52 |
| Figura 33 – Gráfico de operabilidade do <i>moon pool</i> ótimo da primeira execução.          | 53 |
| Figura 34 – Histórico das notas de operabilidade dos campeões da segunda<br>execução. . . . . | 54 |
| Figura 35 – Visualização das notas dos campeões da segunda execução. . . . .                  | 54 |
| Figura 36 – Malha do <i>moon pool</i> ótimo da segunda execução. . . . .                      | 55 |
| Figura 37 – Gráfico de operabilidade do <i>moon pool</i> ótimo da segunda execução.           | 56 |
| Figura 38 – Gráfico de operabilidade do <i>moon pool</i> ótimo da segunda execução.           | 57 |

# Lista de abreviaturas e siglas

|      |                                                       |
|------|-------------------------------------------------------|
| AG   | Algoritmo Genético                                    |
| ROV  | <i>Remotely Operated Vehicle</i>                      |
| SI   | <i>Sistema Internacional</i>                          |
| SPH  | <i>Smoothed Particle Hydrodynamics Method</i>         |
| MPS  | <i>Moving Particle Semi-implicit Method</i>           |
| VOF  | <i>Volume of Fluid</i>                                |
| OWC  | <i>Oscillating Water Column</i>                       |
| IACS | International Association of Classification Societies |
| RAO  | <i>Response Amplitude Operator</i>                    |
| CAD  | <i>Computer Aided Design</i>                          |

# Lista de símbolos

|                   |                                                                                                    |
|-------------------|----------------------------------------------------------------------------------------------------|
| $\pi$             | Razão entre o comprimento de uma circunferência e seu diâmetro, aproximadamente igual a 3,1415...  |
| $T_n$             | Período natural da oscilação da água interna ao <i>moon pool</i> .                                 |
| $d$               | Calado da embarcação.                                                                              |
| $g$               | Aceleração da gravidade.                                                                           |
| $h$               | Amplitude da oscilação da água interna ao <i>moon pool</i> .                                       |
| $l$               | Comprimento do <i>moon pool</i> .                                                                  |
| $U$               | Velocidade de avanço do navio.                                                                     |
| $U'$              | Velocidade de avanço do navio em que as oscilações da água interna ao <i>moon pool</i> se iniciam. |
| $\omega_0$        | Frequência natural da oscilação da água interna ao <i>moon pool</i> .                              |
| $N$               | Quantidade esperada de encontros com picos de onda em um estado de mar irregular.                  |
| $m$               | Duração em horas do estado de mar irregular.                                                       |
| $T_s$             | Período de ocorrência da onda de altura significativa no estado de mar irregular.                  |
| $x_q$             | Valor limite aceitável para avaliação do critério de operabilidade.                                |
| $R$               | Parâmetro da distribuição de Rayleigh das alturas de ondas num estado de mar irregular.            |
| $n_{ex}$          | Quantidade esperada de excedências do valor aceitável $x_q$ em um período de $m$ horas.            |
| $\beta$           | Ângulo de incidência das ondas em relação ao navio.                                                |
| $S$               | Espectro de distribuição da energia de resposta da embarcação.                                     |
| $H_{s-threshold}$ | Altura significativa limite para o critério de transbordo e de compensação de <i>heave</i> .       |
| $D_y$             | Força de deriva média devido à onda incidente no navio.                                            |

|                 |                                                                                                         |
|-----------------|---------------------------------------------------------------------------------------------------------|
| $D_y0$          | Força de deriva média limite para a operação do <i>drillship</i> .                                      |
| $H_{s-drift-y}$ | Altura significativa limite para o critério de força de deriva.                                         |
| $l1$            | Parâmetro geométrico que indica meio comprimento do <i>moon pool</i> .                                  |
| $l2$            | Parâmetro geométrico que indica meia largura (meia boca) do <i>moon pool</i> .                          |
| $m1$            | Parâmetro geométrico que indica a largura da seção transversal plana de proa do <i>moon pool</i> .      |
| $m2$            | Parâmetro geométrico que indica o comprimento da seção longitudinal plana de proa do <i>moon pool</i> . |
| $m3$            | Parâmetro geométrico que indica o comprimento da seção longitudinal plana de popa do <i>moon pool</i> . |
| $m4$            | Parâmetro geométrico que indica a largura da seção transversal plana de popa do <i>moon pool</i> .      |
| $n.f$           | Parâmetro geométrico que indica a quantidade de painéis na quina de vante do <i>moon pool</i> .         |
| $nt$            | Parâmetro geométrico que indica a quantidade de painéis na quina de ré do <i>moon pool</i> .            |
| $\Phi$          | Potencial de velocidades do escoamento da onda.                                                         |
| $x$             | Coordenada longitudinal do navio.                                                                       |
| $y$             | Coordenada transversal do navio.                                                                        |
| $z$             | Coordenada vertical do navio.                                                                           |
| $t$             | Coordenada temporal.                                                                                    |
| $i$             | Unidade imaginária.                                                                                     |
| $\omega$        | Frequência angular da onda.                                                                             |
| $j$             | Índice do modo de movimento do navio.                                                                   |
| $X_j$           | Força ou momento incidindo no $j$ -ésimo modo de movimento do navio.                                    |
| $\varphi$       | Componente espacial do potencial da onda.                                                               |
| $e$             | Número de Euler, aproximadamente igual a 2,7182...                                                      |

|               |                                                                                                                    |
|---------------|--------------------------------------------------------------------------------------------------------------------|
| $a$           | Amplitude da onda.                                                                                                 |
| $k$           | Número de onda.                                                                                                    |
| $K$           | Número de onda para o caso de águas profundas, aproximadamente igual a $\frac{\omega^2}{g}$ .                      |
| $S_H$         | Superfície do casco.                                                                                               |
| $n$           | Vetor normal a uma superfície. Generalizado como o cosseno direcional $(n_j)$ para os modos de movimento do navio. |
| $S_{FM}$      | Superfície livre do <i>moon pool</i> .                                                                             |
| $\varepsilon$ | Parâmetro fictício de viscosidade de Rayleigh.                                                                     |
| $G$           | Função de Green.                                                                                                   |
| $P$           | Ponto de localização da fonte pulsante definida pela função de Green.                                              |
| $Q$           | Ponto genérico do domínio de integração do teorema de Green.                                                       |
| $S_Q$         | Superfície de integração formada pelo conjunto dos pontos $Q$ .                                                    |
| $L_{pp}$      | Comprimento entre perpendiculares da embarcação.                                                                   |
| $B$           | Boca da embarcação.                                                                                                |
| $\Delta$      | Deslocamento da embarcação.                                                                                        |

# Sumário

|            |                                                               |           |
|------------|---------------------------------------------------------------|-----------|
| <b>1</b>   | <b>INTRODUÇÃO</b>                                             | <b>15</b> |
| <b>1.1</b> | <b>Apresentação do problema</b>                               | <b>15</b> |
| <b>1.2</b> | <b>Justificativa</b>                                          | <b>16</b> |
| <b>1.3</b> | <b>Objetivos</b>                                              | <b>17</b> |
| <b>2</b>   | <b>EMBASAMENTO TEÓRICO E LITERATURA</b>                       | <b>19</b> |
| <b>2.1</b> | <b>Revisão bibliográfica</b>                                  | <b>19</b> |
| 2.1.1      | Origens físicas do fenômeno                                   | 19        |
| 2.1.2      | Soluções empíricas para o problema da excitação               | 21        |
| 2.1.3      | Formulações empíricas                                         | 23        |
| 2.1.4      | Abordagens numéricas                                          | 23        |
| <b>2.2</b> | <b>Fundamentação teórica</b>                                  | <b>24</b> |
| 2.2.1      | Avaliação da operabilidade                                    | 24        |
| 2.2.2      | Metodologia de otimização                                     | 28        |
| 2.2.3      | Algoritmo de simulação hidrodinâmica                          | 29        |
| <b>3</b>   | <b>COMPATIBILIZAÇÃO DAS FERRAMENTAS COMPUTACIONAIS</b>        | <b>32</b> |
| <b>3.1</b> | <b>Modelo proposto</b>                                        | <b>32</b> |
| <b>3.2</b> | <b>Integração entre as ferramentas</b>                        | <b>35</b> |
| <b>3.3</b> | <b>Compatibilização da malha para a análise hidrodinâmica</b> | <b>38</b> |
| <b>3.4</b> | <b>Crterios de operabilidade e opções da otimização</b>       | <b>44</b> |
| <b>3.5</b> | <b>Testes e resultados</b>                                    | <b>46</b> |
| 3.5.1      | Avaliação da integração                                       | 48        |
| 3.5.2      | Resultados da primeira execução                               | 49        |
| 3.5.3      | Resultados da segunda execução                                | 53        |
| <b>4</b>   | <b>CONCLUSÃO</b>                                              | <b>58</b> |
| <b>5</b>   | <b>SUGESTÕES PARA TRABALHOS FUTUROS</b>                       | <b>60</b> |
|            | <b>REFERÊNCIAS</b>                                            | <b>61</b> |
|            | <b>APÊNDICE A – CÓDIGOS ALTERADOS PARA A COMPATIBILIZAÇÃO</b> | <b>63</b> |
|            | <b>ANEXO A – CÓDIGOS MANTIDOS PARA A COMPATIBILIZAÇÃO</b>     | <b>88</b> |

# 1 Introdução

Neste capítulo, é introduzido o problema do movimento de oscilação da água interna ao *moon pool*, suas consequências e influências na performance operacional de navios *drillship*. Depois, são explanados os objetivos pretendidos e sua relevância na questão abordada.

## 1.1 Apresentação do problema

Navios *drillship* são embarcações projetadas para executar perfuração *offshore*, seja para fins científicos ou de exploração de jazidas minerais, e, portanto, carregam dentro de si sistemas especializados nesta atividade. Uma das características dessa especialização é a presença de *moon pool*, abertura passante no casco, próximo à meia-nau, que serve para permitir o acesso dos equipamentos de perfuração ao mar quando a embarcação está operando. A Figura 1<sup>1</sup> ilustra como essa abertura pode alcançar grandes dimensões em relação ao casco da embarcação.

Figura 1 – Modelo 3D de navio *drillship*.



Fonte: Artigo no sítio Offshore Energy Today.

A presença de tal abertura impacta de diversas formas o projeto da embarcação. Por exemplo, os responsáveis pelo projeto devem se certificar de que a estrutura é plenamente capaz de suportar os esforços aos quais ela possa ser submetida, mesmo possuindo uma descontinuidade em sua seção mestra.

<sup>1</sup> Extraída de: <<http://www.offshoreenergytoday.com/brazil-schahin-group-receives-usd-692-million-loan-for-new-deepwater-drillship/>>

Este trabalho se concentra em outro aspecto da presença do *moon pool*: o hidrodinâmico. O *moon pool* é capaz de influenciar significativamente o comportamento hidrodinâmico do casco, gerando efeitos que não existiriam de outra forma. O mais notável é, em alguns casos, a excitação da água em seu interior a amplitudes maiores do que as observadas no exterior da embarcação, mesmo com o navio parado. Ou seja, a água no interior do *moon pool* pode ser induzida à ressonância.

As consequências de um fenômeno como este podem ser bastante inconvenientes. A finalidade do *moon pool* é de possibilitar uma operação segura, provendo uma área relativamente livre de intempéries externas. Logo, se ocorre oscilação excessiva da água dentro do próprio *moon pool*, por definição não será possível realizar a operação (DAY, 1990). Nesse sentido, a oscilação da água interior ao *moon pool* pode ser considerada o fator de maior importância no desempenho em operabilidade de um navio de perfuração.

Dependendo da amplitude alcançada por essa oscilação, pode haver transbordo para o convés através do *moon pool*, ou a integridade estrutural dos equipamentos envolvidos na operação pode ser ameaçada. De qualquer forma, a tripulação será obrigada a interromper as atividades, gerando transtornos e prejuízos.

Já existem na literatura propostas para contornar esse problema. Geralmente elas passam pela instalação de apêndices ou outras estruturas que limitam o movimento da água, mas que também limitam, em certa medida, a área disponível para passagem dos equipamentos no *moon pool*. Ademais, geralmente são feitas simplificações do problema estudado, como redução ao caso 2D e desconsideração dos efeitos hidrodinâmicos da interação casco-água no *moon pool*, a exemplo de (DAY, 1990). Portanto, do melhor conhecimento do autor, não foram encontrados na literatura trabalhos que relacionem o formato do *moon pool* ao desempenho hidrodinâmico em operabilidade, considerando a complexidade do sistema tridimensional e incluindo a interação entre a estrutura flutuante e a superfície livre interna ao *moon pool*.

## 1.2 Justificativa

A manutenção da operabilidade de um *drillship* se mostra importante sob diversos pontos de vista. Um navio capaz de operar em uma maior diversidade de condições pode, a princípio, oferecer um maior retorno financeiro ao armador, ao evitar que uma operação seja cancelada devido a condições climáticas adversas. Levando em conta a complexidade dessas operações e a quantidade de recursos nelas envolvidos, este se destaca como um ponto bastante favorável. Também se reduzem as chances de danos aos equipamentos embarcados por cargas de onda incidindo no interior do *moon pool* ou no convés (*green water*).

Além de fatores puramente financeiros, a operabilidade também está relacionada à segurança e bem-estar da tripulação. Ao garantir que o navio seja capaz de suportar estruturalmente o estado de mar característico da região de operação, reduzimos o risco de acidentes devido às cargas hidrodinâmicas incidentes e, a longo prazo, aumentamos a vida útil de sua estrutura. Ao reduzirmos as respostas do navio ao movimento das ondas, melhoramos a condição de trabalho da tripulação, que fica menos exposta a forças e acelerações desconfortáveis e potencialmente perigosas durante a operação.

### 1.3 Objetivos

Este projeto é parte de uma linha de pesquisa maior, correspondente à tese de doutoramento da orientadora do autor, prof.<sup>a</sup> Paula Suemy Arruda Michima. O projeto como um todo busca criar uma ferramenta para otimizar a operabilidade de um determinado casco de *drillship*, variando a geometria do seu *moon pool*.

Para alcançar esse objetivo, a ferramenta se divide em módulos, a saber:

- a) Código de simulação do comportamento em ondas com resultados de resposta hidrodinâmica no domínio da frequência, programado em Fortran. Esta é a parte do código que recebe a geometria do casco como entrada direta para computação dos cálculos hidrodinâmicos, realizados de acordo com a teoria potencial, pelo método dos painéis. Desenvolvido pelo Eng. PhD. Hiroshi Kawabe.
- b) Algoritmo de otimização, programado em MATLAB. Algoritmo genético (AG) criado para buscar a melhor combinação de casco e *moon pool*, iterativamente, para um dado estado de mar. Utiliza notas de avaliação da operabilidade, calculadas a partir dos resultados das simulações hidrodinâmicas, para selecionar os indivíduos mais promissores (CAVALCANTE, 2015).
- c) Algoritmo para acoplamento do *moon pool* na malha de um casco série 60 previamente definido (CAVALCANTE, 2015).

Este trabalho se concentra especificamente em compatibilizar as interfaces entre esses módulos, criando uma integração funcional entre eles. Esse objetivo se subdivide nos seguintes:

- a) estabelecer a criação automática da malha do casco com *moon pool* para quaisquer parâmetros geométricos definidos pelo algoritmo genético;
- b) adaptar o código existente para escrever essas malhas no padrão de entrada do módulo de simulação hidrodinâmica;

- c) implementar a obtenção das notas de desempenho em operabilidade a partir dos resultados da simulação hidrodinâmica, cujo cálculo já é previamente estabelecido, também de uma forma generalizada para qualquer *moon pool* instalado no casco. As notas devem ser calculadas de acordo com critérios de operabilidade definidos e depois devem ser utilizadas como parâmetro de avaliação pelo algoritmo de otimização.

Depois de cumpridos esses objetivos, devem ser realizadas considerações sobre os resultados obtidos a partir da ferramenta de otimização, apenas como apontamento de ajustes necessários para possíveis trabalhos futuros.

## 2 Embasamento teórico e literatura

Este capítulo se divide em duas seções. A primeira, da revisão bibliográfica, apresenta o fenômeno da excitação da água em *moon pools*, central a este trabalho, e as principais linhas de abordagem, teóricas, empíricas e numéricas encontradas a respeito dele na literatura. A segunda, da fundamentação teórica, introduz outros conceitos relevantes, relativos à operabilidade de embarcações e otimização por algoritmos genéticos. Nela, enumeram-se os principais trabalhos dos quais se fez uso durante a compatibilização dos códigos aqui proposta.

### 2.1 Revisão bibliográfica

*Moon pools* são aberturas no fundo do casco de navios, utilizadas para lançar e recuperar equipamentos como ROV's (*Remotely Operated Vehicle*), tubulações, *risers* ou até mesmo mergulhadores. São usadas em diferentes tipos de embarcações: *cable* e *pipe-laying* (instaladores de cabos e tubulações), exploração e perfuração (ou *drillships*, o objeto de estudo deste trabalho), navios de pesquisa e de suporte *offshore*. Têm como finalidade prover um ambiente seguro e livre das intempéries externas às quais o navio possa estar exposto (ondas, ventos e gelo, por exemplo). Ficam localizadas próximo ao centro de flutuação do navio, onde as oscilações por resposta hidrodinâmica são menores.

No entanto, em certas situações, a coluna de água no interior do *moon pool* pode ser excitada de forma a gerar amplitudes maiores do que as observadas ao redor da embarcação. Com o navio parado, essas oscilações podem impossibilitar a operação, danificar equipamentos próximos ao *moon pool*, causar transbordo e oferecer perigo à tripulação. Com o navio em trânsito, pode aumentar a resistência ao avanço e gerar vibrações estruturais.

#### 2.1.1 Origens físicas do fenômeno

Teoricamente, o problema da oscilação da água no *moon pool* costuma ser dividido em dois tipos básicos, com diferentes mecanismos de excitação da coluna de água:

- a) Navio em águas calmas com velocidade de avanço.
- b) Navio estacionário em mar com ondas.

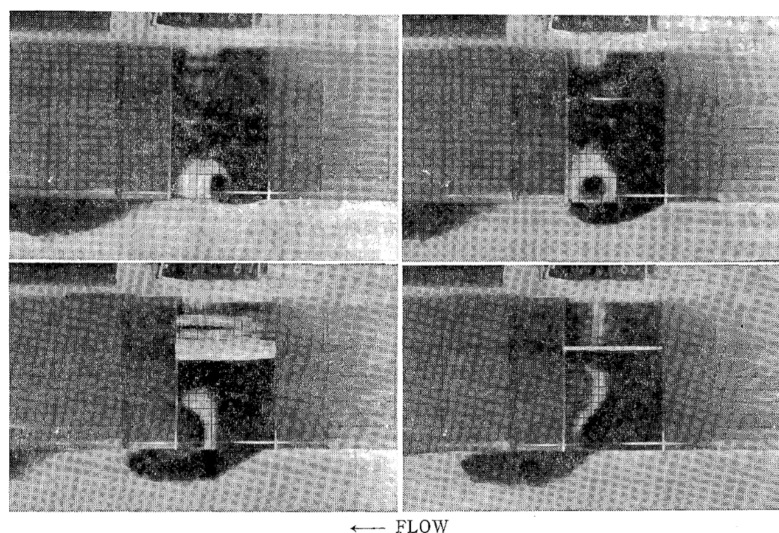
Na prática, podem ocorrer combinações dos efeitos observados nos dois casos.

No caso de navegação em águas calmas, a excitação da coluna de água é causada pela separação do fluido passando do casco do navio para a região do *moon pool*, o que introduz flutuações de velocidade e pressão no escoamento. Naudascher (1967) explica que três fatores determinam se essas perturbações vão evoluir a ponto de se tornar um padrão de forças significativo, possivelmente capaz de causar vibrações estruturais no casco. São esses:

- a) amplificação da intensidade dessas flutuações;
- b) intensificação de mecanismos através dos quais as excitações da coluna se autorregulam (correlação no espaço);
- c) distribuição da energia de resposta da coluna de água à excitação se concentra em torno de uma frequência dominante.

A perturbação do fluido é gerada pela formação de vórtices que se desprendem da aresta frontal do *moon pool*. Esses vórtices se originam das flutuações introduzidas pela brusca variação do campo de velocidades e pressões no escoamento quando do desprendimento. Essas flutuações se amplificam devido aos efeitos viscosos do fluido, até se tornarem vórtices. Tal mecanismo de geração foi descrito por Rockwell e Naudascher (1979) e reproduzido experimentalmente por Fukuda (1977), como se observa nas fotografias da Figura 2.

Figura 2 – Formação de vórtices por separação do fluxo.



Fonte: Fukuda (1977).

Adicionalmente, devido ao espaço confinado proporcionado pelo *moon pool*, podem surgir mecanismos de auto excitação do fenômeno (correlação no espaço). Toda vez que um vórtice colide com a superfície posterior do *moon pool*, ele perturba o campo de pressões na cavidade, o que pode causar o desprendimento precipitado

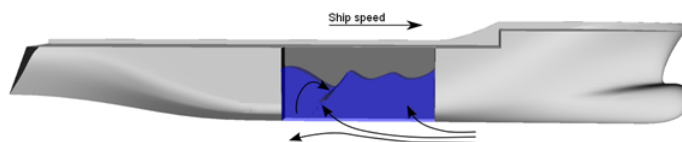
de um outro vórtice que ainda estaria se formando na parte de vante da mesma. Esse acoplamento pode causar um trancamento de frequência, concentrando a energia do fenômeno nessa frequência dominante.

Esse trancamento de frequência pode se agravar a ponto de causar vibrações estruturais quando combinada com o efeito de histerese, ou seja, quando o ciclo de formação de vórtices persiste mesmo que o navio acelere ou desacelere para uma velocidade de avanço diferente da que o originou.

Quando o movimento da água no interior do *moon pool* alcança certa amplitude, a estrutura do navio é excitada a ponto de reproduzir um movimento na mesma frequência. Este acoplamento pode ocorrer tanto para a combinação de movimentos de pistão do *moon pool* e *heave* do navio quanto a de *sloshing* do *moon pool* e *surge* do navio. Tais efeitos induzem a ressonância a ponto de, em alguns casos, causar trasbordo da água para o convés através do *moon pool*. A resistência ao avanço também aumenta consideravelmente nesses casos, e pode se intensificar com o aumento da velocidade, caso as amplitudes da oscilação da água no interior do *moon pool* também aumentem. Esse tipo de ocorrência já foi observado tanto em ensaios com modelos quanto a bordo de embarcações.

No caso do navio parado em ondas, a excitação da coluna de água ocorre devido às acelerações verticais e diferenças de pressão na base do *moon pool*, induzidas pelas ondas e movimentos aos quais o navio está sujeito. Aalbers (1984) escreveu as equações de movimento para a coluna de água nessa situação. Törnblom e Hammargren (2012) fizeram um modelo esquemático representando a origem da excitação da água interna ao *moon pool* para o caso do navio em trânsito (Figura 3) e estacionário (Figura 4).

Figura 3 – Excitação devido à separação do fluxo para o caso do navio em trânsito.

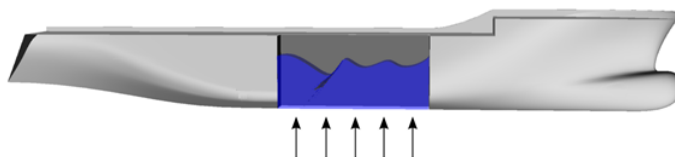


Fonte: Törnblom e Hammargren (2012).

### 2.1.2 Soluções empíricas para o problema da excitação

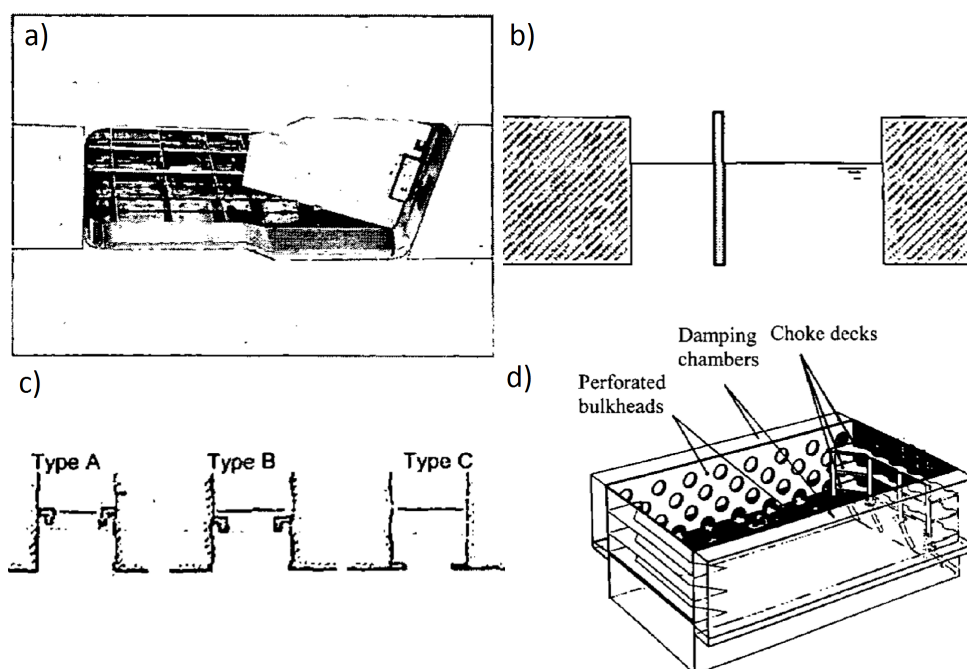
Já foram testadas diversas alternativas para a redução das oscilações da coluna de água no *moon pool*. Algumas delas passam pela redução da excitação causada pelo descolamento dos vórtices, geralmente com flaps ou anteparas posicionadas de

Figura 4 – Excitação devido à incidência de ondas para o caso do navio estacionário.



Fonte: Törnblom e Hammargren (2012).

forma a bloquear ou defletir o vórtice formado na condição de navio em trânsito. Outras soluções passam pela redução do movimento da água propriamente dito, com a adição de flanges ou câmaras de amortecimento no interior da cavidade. A Figura 5 exemplifica alguns desses apêndices. Em a), está representado um *flap* defletor de vórtices; em b), uma antepara vertical dividindo a abertura do *moon pool*; em c), alguns tipos de flanges, como testados por Fukuda (1977); e em d), uma câmara de amortecimento.

Figura 5 – Exemplos de apêndices de *moon pool*.

Fonte: Gaillard e Cotteleer (2004).

A grande maioria dessas soluções, no entanto, se baseia na adição de apêndices ou estruturas especialmente instaladas para reduzir as oscilações, e, portanto, diminuem a área de acesso do *moon pool* ou restringem de alguma forma a sua operação. Além disso, se não restringem, acabam não tendo função efetiva no amortecimento da oscilação no caso do navio em operação, sem velocidade de avanço, submetido a

ondas, pois são projetadas especialmente para inibir o mecanismo de excitação por vórtices, que surge apenas quando o navio está em trânsito.

### 2.1.3 Formulações empíricas

Tradicionalmente, os resultados mais confiáveis para o problema da excitação de água no *moon pool* são encontrados experimentalmente, principalmente quando se estuda o acoplamento dos movimentos do navio e *moon pool*. A partir da teoria e desses estudos experimentais, foram inferidas fórmulas semi-empíricas para parâmetros das oscilações em *moon pools*. O período natural, por exemplo, foi estimado por Faltinsen (1993) com a Equação 2.1:

$$T_n = 2\pi \sqrt{\frac{d}{g}} \quad (2.1)$$

Onde  $d$  é o calado e  $g$  é a aceleração da gravidade. Fukuda (1977) adicionou ao calado o termo  $0,41\sqrt{S}$ , sendo  $S$  a superfície livre do *moon pool*, para contabilizar o efeito da massa adicional e da elevação do nível da água.

Fukuda (1977) também estimou a amplitude do modo pistão da coluna d'água para o navio em trânsito, adimensionalizada na Equação 2.2, e Aalbers (1984) calculou equações de movimento para o navio parado em ondas.

$$\frac{h}{l} = \frac{3\pi}{16} \cdot \frac{U - U'}{l\omega_0} \quad (2.2)$$

Onde  $h$  é a amplitude da oscilação,  $l$  é o comprimento do *moon pool*,  $U$  é a velocidade do navio,  $U'$  é a velocidade do navio em que as oscilações se iniciam e  $\omega_0$  é a frequência natural da oscilação da coluna d'água, todos em unidades do Sistema Internacional (SI). A fórmula é válida apenas para frequências de oscilação abaixo da frequência natural.

Molin (2001) desenvolveu um método para encontrar os modos de ressonância de pistão e sloshing, e Covert (1970) desenvolveu outro para estimar a velocidade do navio na qual as oscilações se iniciam.

### 2.1.4 Abordagens numéricas

Os avanços no desenvolvimento de métodos numéricos capazes de simular com progressiva fidelidade os movimentos no *moon pool* são constantemente atualizados na literatura. Gaillarde e Cotteleer (2004) apontam como promissor o método de *Volume of Fluid* (VOF), que resolve as equações de Navier-Stokes utilizando uma função binária para identificar as regiões da malha que contém fluido.

Atualmente, se popularizam métodos que modelam o fluido como um conjunto de partículas. Um deles é o *Smoothed Particle Hydrodynamics Method*, ou SPH (MONAGHAN, 1992), onde as pressões são estimadas estatisticamente através de uma função núcleo, definida como soma das pressões relacionadas às partículas vizinhas. Outro, bastante similar ao primeiro, é o *Moving Particle Semi-implicit Method* (MPS), proposto por Koshizuka e Oka (1996), onde a função núcleo é substituída por outro método de estimativa.

Alguns pesquisadores adotam a estratégia de mesclar um método numérico baseado na teoria potencial com outro que leve em conta efeitos viscosos. Como exemplo, Fredriksen, Kristiansen e Faltinsen (2014) utilizaram um domínio híbrido, acoplando o método de volumes finitos na parte viscosa a um método potencial (no caso, o método de célula harmônica polinomial) na parte não-viscosa.

Outros trabalhos calibram a teoria potencial com efeitos viscosos obtidos independentemente, seja a partir de resultados experimentais ou de formulações puramente teóricas. Malta et al. (2006), por exemplo, utilizam a primeira abordagem em um artigo sobre a utilização de *moon pools* na redução de movimentos em plataformas monocoluna.

Bull (2015), por outro lado, utiliza a segunda abordagem em um artigo sobre o projeto de um dispositivo *Oscillating Water Column* (OWC), ou seja, um dispositivo capaz de gerar energia a partir do movimento da coluna de água em seu interior. Além disso, propõe a adição de termos de acoplamento entre os movimentos do sistema flutuante e do pistão de água para representar o movimento relativo entre ambos.

## 2.2 Fundamentação teórica

O sistema de otimização utilizado neste trabalho se apoia em grande parte em um programa de análise hidrodinâmica baseado na teoria potencial. Nesse programa, a resposta do navio é avaliada com funções de Green que consideram os efeitos da superfície livre interna ao *moon pool* com amortecimento potencial. Como o objeto de estudo desta pesquisa é a operabilidade, o caso de análise atém-se à situação do navio sem velocidade de avanço, submetido à excitação de ondas, portanto a teoria potencial deve ser suficiente para representar o fenômeno com fidelidade neste estágio do estudo.

### 2.2.1 Avaliação da operabilidade

As bases da otimização aqui proposta foram delineadas no artigo de Michima e Kawabe (2014).

Nele, os autores estabelecem as formulações que permitem estimar a altura significativa de onda que provavelmente fará com que cada parâmetro seja excedido um dado número de vezes durante um certo período de tempo, tomada como altura limite de operabilidade. Isto é feito a partir do tratamento estatístico da resposta da embarcação ao estímulo do estado de mar irregular.

O número esperado de encontro com picos em um estado de mar, definido por uma altura significativa  $H_s$  e um período significativo  $T_s$ , em um intervalo de  $m$  horas é dado pela equação [Equação 2.3](#).

$$N = \frac{m \cdot 60 \cdot 60}{T_s} \quad (2.3)$$

A altura significativa,  $H_s$ , ou  $H_{1/3}$ , ou  $H_{m0}$ , costuma ser definida como a altura média do terço de ondas mais altas do estado de mar irregular, ou quatro vezes o desvio padrão da elevação da superfície do mar, o mesmo que quatro vezes a raiz quadrada do momento de ordem zero,  $m_0$ , do espectro de onda. O período significativo  $T_s$  é, logicamente, o período de ocorrência da onda de altura significativa.

Assumindo que a distribuição estatística dos picos de resposta do navio possa ser descrita por uma distribuição de Rayleigh com parâmetro  $R$ , pois está relacionada à distribuição de alturas de um sinal cujo histograma apresenta distribuição normal, então o valor  $x_q$  que provavelmente será excedido  $n_{ex}$  vezes em  $m$  horas é dado pela [Equação 2.4](#).

$$x_q = R(H_s, T_s, \beta) \sqrt{2 \ln(N/n_{ex})} \quad (2.4)$$

Onde  $R$  é avaliado individualmente para cada ângulo de incidência da onda,  $\beta$ .

O parâmetro  $R$  pode ser relacionado com o espectro de resposta do navio através da [Equação 2.5](#).

$$R(H_s, T_s, \beta)^2 = \int_0^\infty S(\omega : H_s, T_s, \beta) d\omega \quad (2.5)$$

Sendo a resposta do navio proporcional à altura da onda incidente, é possível rearranjar a [Equação 2.4](#) na [Equação 2.6](#), a qual se aplica à avaliação da altura limitante de operabilidade para os critérios de transbordo do *moon pool* e compensação de *heave*.

$$H_{s-threshold} = \frac{x_q}{R(H_s = 1, T_s, \beta) \sqrt{2 \ln\left(\frac{N}{n_{ex}}\right)}} \quad (2.6)$$

Combinando as equações 2.3 e 2.6, obtemos a Equação 2.7:

$$H_{s-threshold} = \frac{x_q}{R(H_s = 1, T_s, \beta) \sqrt{2 \ln \left( \frac{m \cdot 3600}{T_s \cdot n_{ex}} \right)}} \quad (2.7)$$

Analogamente, definiu-se a altura significativa máxima para que a força de deriva devido à onda incidente  $D_y$  não supere a força de deriva permissível  $D_{y0}$ , conforme a Equação 2.8:

$$H_{s-drift-y} = \left( \frac{D_{y0}}{D_y(H_s^2 = 1, T_s, \beta) \frac{2\pi}{T_s}} \right)^{1/3} \quad (2.8)$$

Estas duas últimas equações (2.7 e 2.8) são as únicas necessárias para o cálculo das alturas significativas permissíveis, já que as simulações hidrodinâmicas fornecem os parâmetros relacionados à resposta do casco ( $R$ ,  $D_y$ ,  $T_s$ ) e os demais são definidos de acordo com os limites aceitáveis para a operação em questão ( $x_q$ ,  $D_{y0}$ ,  $m$ ,  $n_{ex}$ ).

Os critérios de operabilidade propostos no trabalho são:

- a) força de deriva, relacionada à capacidade de o navio manter seu posicionamento no plano azimutal;
- b) transbordamento do *moon pool*, que pode também causar a incidência de cargas hidrodinâmicas nos equipamentos de convés;
- c) amplitude oscilatória superior ao limite de compensação de *heave*, delimitado pelo curso do cilindro de compensação. Tem como finalidade evitar a incidência de impactos que ameacem a integridade do equipamento de perfuração;
- d) momento fletor máximo à meia-nau, relacionado à integridade estrutural da embarcação.

Os critérios, com seus respectivos valores limitantes, estão relacionados na Tabela 1.

A potência do sistema DPS foi adotada como idêntica ao do *drillship* Chikyū, e o limite de momento fletor foi estimado a partir das regras da IACS (International Association of Classification Societies), como o de intensidade máxima ao longo de 1 mês.

Depois de apresentar gráficos de RAO (*Response Amplitude Operator*) dos critérios para estados de onda regular e de espectro para estados de onda irregular, os resultados obtidos são resumidos em dois novos tipos de gráficos. O primeiro,

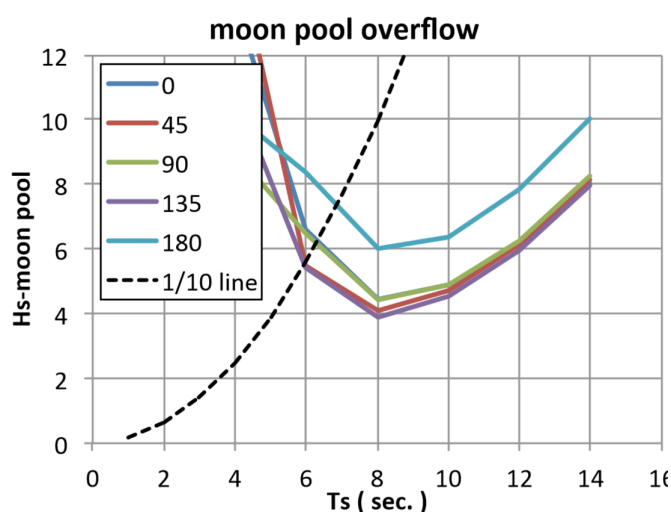
Tabela 1 – Limites aceitáveis dos critérios de operabilidade.

| Item                           | Critério             | Valor específico |
|--------------------------------|----------------------|------------------|
| Transbordo do <i>moon pool</i> | Borda-livre          | 4 m              |
| Compensador de <i>heave</i>    | Curso do compensador | 3,5 m            |
| Deriva                         | Potência DPS total   | 34.260 PS        |
| Momento fletor meia-nau        | IACS S11             | Máx. de 1 mês    |

Fonte: Michima e Kawabe (2014).

exemplificado na Figura 6, expõe os limites aceitáveis de altura significativa para um dado critério de operabilidade em função do período significativo, plotados para diversos ângulos de incidência, dentro da premissa linear para declividade de onda delimitada pela linha preta tracejada.

Figura 6 – Gráfico de altura significativa aceitável para operação.

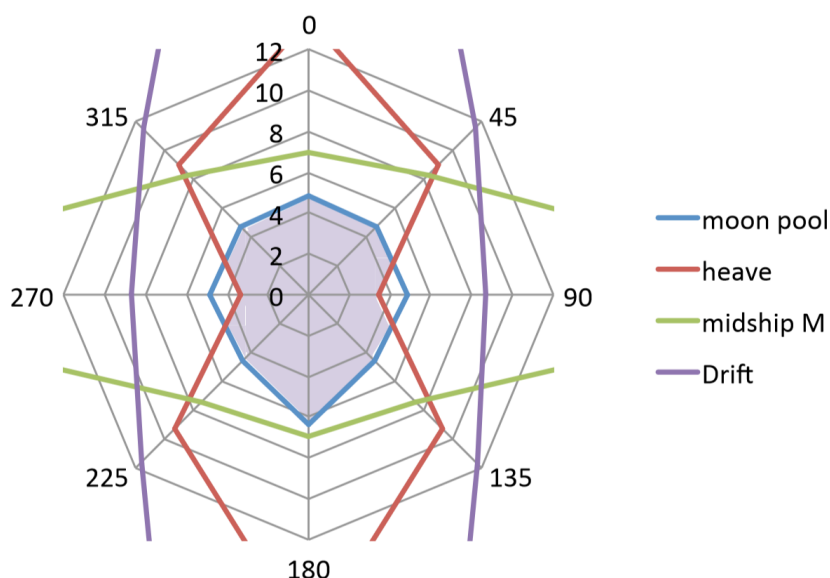


Fonte: Michima e Kawabe (2014).

Uma vez identificado o período crítico de altura de onda significativa, ele é selecionado para traçar o gráfico de radar. Esse tipo de gráfico, que será utilizado neste trabalho e está exemplificado na Figura 7, representa os limites aceitáveis de altura significativa em todos os critérios, para cada valor de período significativo, determinando a operabilidade de acordo com a área interna à intersecção entre os critérios. A altura significativa,  $H_s$ , é representada pela dimensão radial do gráfico e o ângulo de incidência da onda,  $\beta$ , pelo ângulo de cada eixo em relação à direção vertical.

O gráfico de radar apresenta a vantagem de permitir rápida interpretação da condição de operabilidade em um dado estado de mar, pois ele relaciona explicitamente a direção de encontro da onda com seu respectivo limite seguro de operação.

Figura 7 – Gráfico de região segura de operação.



Fonte: Michima e Kawabe (2014).

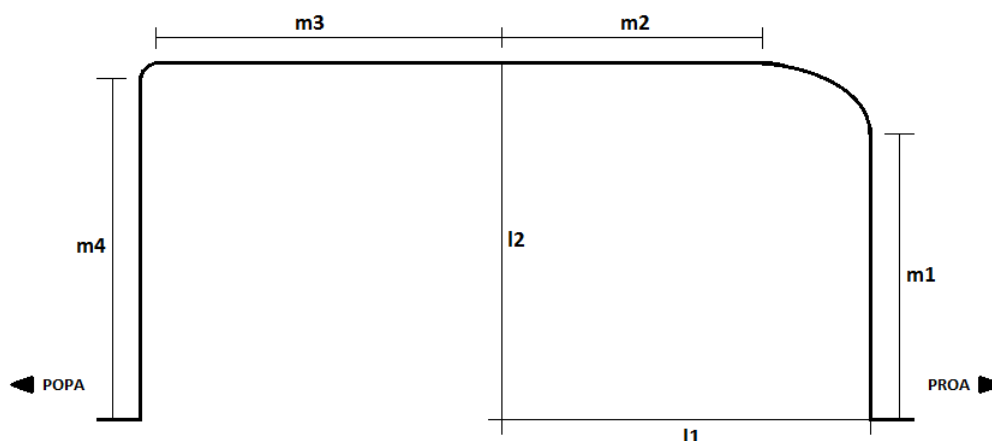
## 2.2.2 Metodologia de otimização

O formato dos *moon pools* projetados no casco analisado pelo algoritmo é definido parametricamente, e os parâmetros são variados por um código de otimização desenvolvido por Cavalcante (2015). Este utiliza uma metodologia mista, que inclui uma busca ingênua parâmetro a parâmetro para as dimensões principais, e, dentro de cada combinação de parâmetros nessa busca, uma otimização baseada em um algoritmo genético para definir o formato da abertura.

Na Figura 9,  $l_1$  e  $l_2$  são os parâmetros definidos na varredura da busca ingênua, enquanto os demais ( $m_1$ ,  $m_2$ ,  $m_3$  e  $m_4$ ) são variados pelo algoritmo genético. Além desses parâmetros, existem também  $nf$  e  $nt$ , também variados pelo algoritmo genético, os quais definem a quantidade de subdivisões nas quinas do *moon pool*, representadas como elípticas na figura.

No ramo de inteligência artificial, algoritmos evolutivos são algoritmos que, para gerar soluções a problemas de otimização, recorrem a métodos inspirados de diversas maneiras em princípios da teoria da evolução natural. Algoritmos genéticos, os quais são uma classe específica de algoritmos evolutivos, utilizam conceitos de hereditariedade, mutação, seleção e cruzamento para gerar novos indivíduos na busca da otimização (MITCHELL, 1998).

Nesse contexto, os indivíduos têm suas características definidas por um conjunto de genes, cada um podendo assumir diferentes valores. Neste trabalho, os genes definem a configuração geométrica do *moon pool*. O mecanismo de hereditariedade

Figura 8 – Parametrização do *moon pool* construído pelo código de acoplamento.

Fonte: Cavalcante (2015).

permite a priorização dos indivíduos mais promissores, que transmitem seus genes às novas gerações. O mecanismo de cruzamento, no qual um indivíduo é gerado com características combinadas de dois outros, também, porém ele introduz a chance de obtenção de indivíduos que ainda não haviam sido analisados anteriormente.

Finalmente, a mutação modifica, com algum fator de aleatoriedade, os genes de certos indivíduos da busca da otimização, aumentando a variabilidade dos indivíduos analisados por ela numa tentativa de evitar que a varredura fique restrita a um ótimo local.

A escolha de aliar uma busca ingênua a um algoritmo genético foi tomada para garantir uma varredura extensiva das várias possibilidades de dimensões e formatos que o *moon pool* poderia adotar.

Já existe na literatura registro da aplicação de algoritmos genéticos à otimização da resposta hidrodinâmica de embarcações. Bagheri e Ghassemi (2014) apresentaram um algoritmo que varia o formato do casco e as dimensões principais em busca do melhor comportamento no mar. A função objetivo da otimização foi minimizar os valores de pico do movimento vertical absoluto na região de proa, próxima à perpendicular de vante.

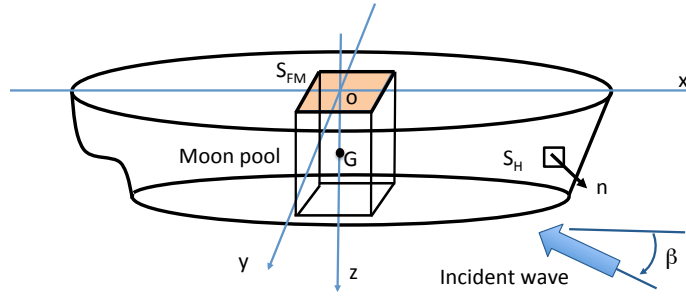
### 2.2.3 Algoritmo de simulação hidrodinâmica

O algoritmo de simulação hidrodinâmica incorporado na otimização foi utilizado tal qual fornecido por seu desenvolvedor. O código se baseia no método dos painéis, ou dos elementos de fronteira, para realizar a estimativa dos potenciais de velocidade da água que interage com o casco, utilizando funções de Green. Os cálculos são realizados

para o domínio da frequência. A metodologia básica de obtenção dos potenciais de velocidade, conforme implementada, é resumida a seguir.

Na convenção direcional adotada, representada na [Figura 9](#), o plano  $xy$  é coincidente com a superfície da água. O eixo  $z$  está posicionado no centro de gravidade do navio e aponta para baixo, no sentido do domínio fluido.

Figura 9 – Convenção de coordenadas adotada pelo código de simulação.



Fonte: [Michima e Kawabe \(2014\)](#).

Os potenciais de velocidade gerados pela radiação devido ao movimento do navio são definidos conforme a [Equação 2.9](#).

$$\Phi_j(x, y, z : t) = i\omega X_j \varphi_j(x, y, z) e^{i\omega t}, \quad j = 1, 2, \dots, 6. \quad (2.9)$$

Onde cada  $X_j$  corresponde à força ou momento do  $j$ -ésimo modo de movimento do navio, e  $t$  é o tempo.

Os potenciais de onda incidente e difratada são definidos conforme as equações [2.10](#) e [2.11](#), respectivamente.

$$\Phi_0(x, y, z : t) = \frac{ga}{i\omega} \varphi_0(x, y, z) e^{i\omega t}, \quad \varphi_0(x, y, z) = e^{-kz + ik(\cos\beta + \sin\beta)} \quad (2.10)$$

$$\Phi_7(x, y, z : t) = \frac{ga}{i\omega} \varphi_7(x, y, z) e^{i\omega t} \quad (2.11)$$

Sendo  $a$  a amplitude da onda e  $k$  o número de onda.

Os potenciais são governados pela equação de Laplace ([2.12](#)).

$$\nabla^2 \varphi_j = 0 \quad (2.12)$$

O problema está submetido a algumas condições de contorno:

- a) A superfície da água ao redor do casco está sujeita à condição linearizada de superfície livre, como nas equações 2.13 e 2.14:

$$\frac{\partial \varphi_j}{\partial z} + K \varphi_j = 0 \quad (2.13)$$

$$K = \frac{\omega^2}{g} \quad (2.14)$$

- b) A água ao redor da superfície do casco,  $S_H$ , está sujeita à condição de impermeabilidade do mesmo (Equação 2.15) e à relação cinemática entre as ondas incidente e difratada (Equação 2.16):

$$\frac{\partial \varphi_j}{\partial n} = n_j, \quad j = 1, 2, \dots, 6. \quad (2.15)$$

$$\frac{\partial \varphi_0}{\partial n} + \frac{\partial \varphi_7}{\partial n} = 0 \quad (2.16)$$

Sendo  $n_j$  o cosseno direcional correspondente ao  $j$ -ésimo modo de movimento do navio.

- c) A superfície da água interna ao *moon pool*,  $S_{FM}$ , está sujeita a uma condição similar à da Equação 2.13. A diferença é a adição de um termo envolvendo o parâmetro  $\varepsilon$ , o termo de amortecimento de Rayleigh, que atribui equilíbrio ao módulo da amplitude de resposta da superfície livre com relação ao fenômeno real. A condição final fica como na Equação 2.17.

$$\frac{\partial \varphi_j}{\partial z} + K \varphi_j + i\varepsilon K \varphi_j = 0 \quad (2.17)$$

$G(P, Q)$  é a função de Green que define uma fonte pulsante no ponto  $P$  da superfície do casco ou do *moon pool* (domínio  $Q$ ) e está sujeita às condições de contorno da superfície livre. Aplicando-a ao teorema de Green, é possível expressar o potencial de velocidades na superfície do casco,  $\varphi_j(P)$ , nos termos da Equação 2.18.

$$\begin{aligned} \frac{1}{2} \varphi_j(P) = & \iint_{S_H} \left( \frac{\partial \varphi_j(Q)}{\partial n_Q} G(P, Q) - \varphi_j(Q) \frac{\partial G(P, Q)}{\partial n_Q} \right) dS_Q \\ & - \iint_{S_{FM}} \left( K(1 + i\varepsilon) G(P, Q) + \frac{\partial G(P, Q)}{\partial n_Q} \right) \varphi_j(Q) dS_Q, \quad j = 1, 2, \dots, 7. \end{aligned} \quad (2.18)$$

A equação é resolvida numericamente para o potencial. Depois de encontrado, ele é pós-processado para se obter resultados de forças hidrodinâmicas, equações de movimento e, finalmente, valores de interesse para cada critério avaliado. O restante da dedução pode ser consultado em (MICHIMA; KAWABE, 2014).

## 3 Compatibilização das ferramentas computacionais

### 3.1 Modelo proposto

Quando do início deste trabalho, estavam disponíveis os três módulos apresentados na [seção 1.3](#), os quais constituem a base do sistema de otimização.

As ferramentas de acoplamento da malha do *moon pool* ao casco previamente definido e a ferramenta de otimização estão descritas em (CAVALCANTE, 2015). O módulo de simulações hidrodinâmicas segue as premissas expostas na [subseção 2.2.3](#).

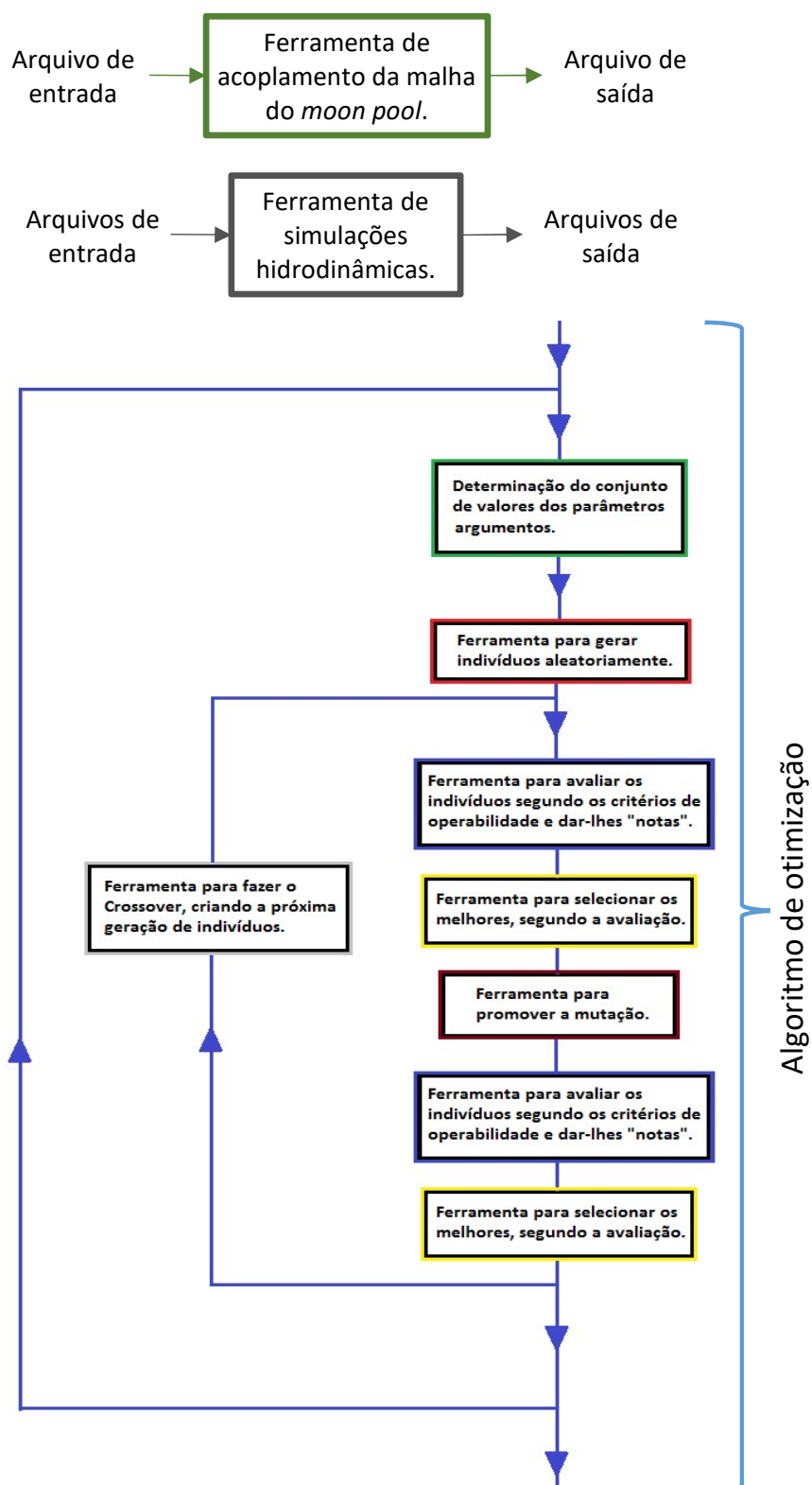
No entanto, não havia comunicação entre esses componentes, o que inviabilizava a execução da otimização propriamente dita. A [Figura 10](#) exibe um esquema dos três componentes originais. O algoritmo de otimização é representado de forma expandida para permitir a representação da integração proposta, mais à frente.

Da forma como estava implementado, o algoritmo de otimização não gerava automaticamente as malhas dos indivíduos por ele definidos. Além disso, presumia que os resultados das análises hidrodinâmicas já fossem dados em execução de processamento à parte. Na prática, os critérios de operabilidade utilizados eram fictícios, apenas para garantir a funcionalidade do código.

Neste trabalho, propõe-se a integração dos três módulos apresentados em um sistema de otimização coeso que, para cada *moon pool* definido dentro das iterações do AG, integre automaticamente a sua malha correspondente ao casco definido. O indivíduo gerado deve então ser analisado pelo código de simulações hidrodinâmicas, em Fortran, e os resultados dessas análises utilizados para avaliar a performance desse indivíduo automaticamente, através da área em que a operabilidade é garantida no seu respectivo gráfico de radar. A nota considerada para a otimização é sempre a menor dentre as calculadas para cada estado de mar escolhido pelo usuário, por representar a situação crítica encontrada pela embarcação. Obtida a nota, ela é repassada para o AG, que a toma como base para decidir a prioridade com que as características paramétricas desse indivíduo serão manuseadas na próxima iteração.

Este tipo de integração implicaria em um modelo como o proposto, simplificado, na [Figura 11](#). A primeira simplificação aplicada à figura consiste na omissão da representação de alguns arquivos que, apesar de necessários para a execução dos componentes envolvidos, não eram relevantes para a ilustração da integração proposta. A segunda é a representação do quadro azul, à esquerda, como uma única ferramenta

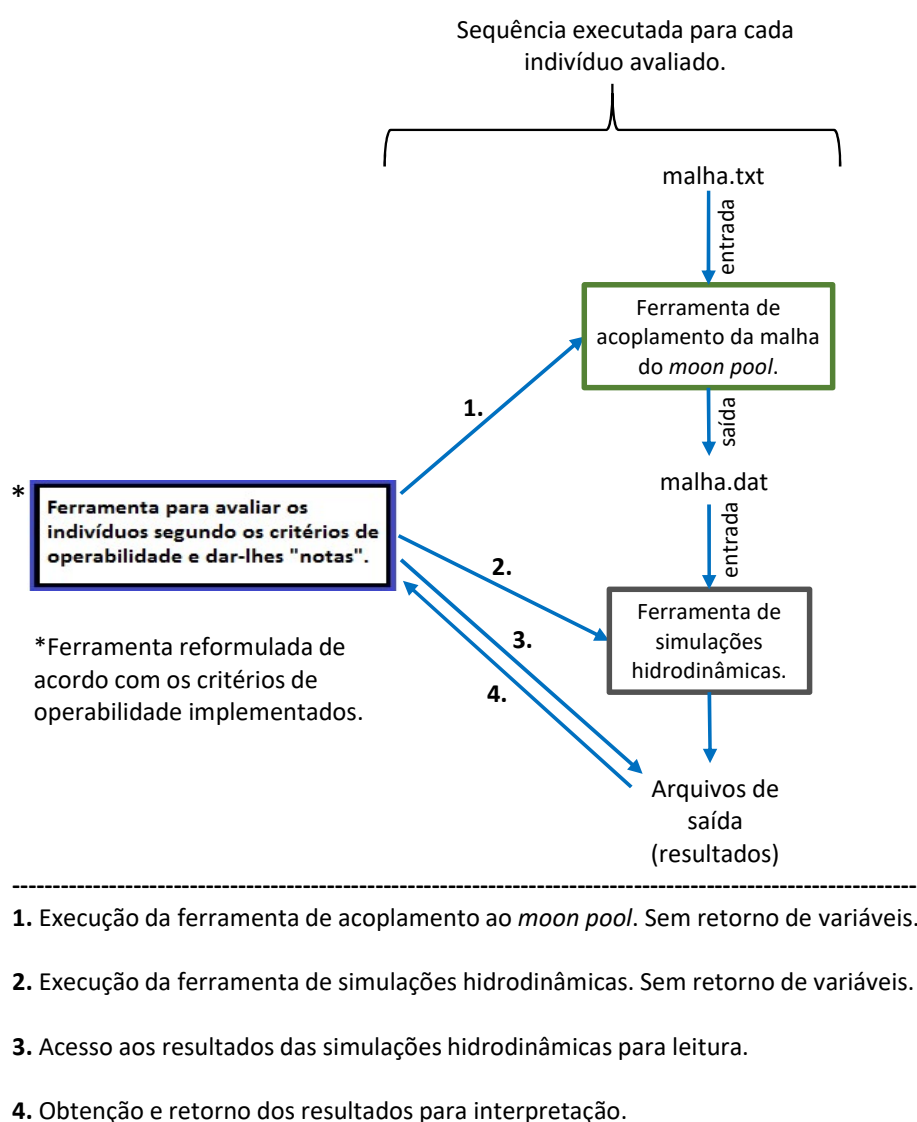
Figura 10 – Ferramentas computacionais provenientes de trabalhos anteriores.



Fonte: Adaptado de Cavalcante (2015).

responsável pela avaliação e atribuição de notas aos indivíduos definidos. Na prática, esta ferramenta tem alguns de seus processos terceirizados para funções externas, que são chamadas repetidas vezes para realizar a leitura e avaliação dos resultados das simulações. Ainda assim, todas as notas obtidas são retornadas para a ferramenta central ao fim da execução das funções auxiliares, de modo que a representação destas apenas adicionaria um fator de complicação desnecessário à compreensão do modelo.

Figura 11 – Modelo de integração proposto.



Fonte: Elaborado pelo autor.

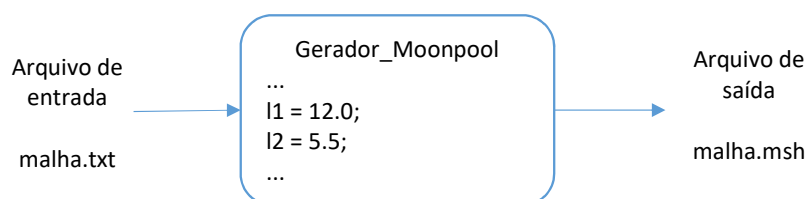
A junção dos componentes pré-existentes em um sistema tal como o representado será detalhada nas próximas seções deste capítulo.

## 3.2 Integração entre as ferramentas

O primeiro passo para integrar o código de geração de malha com *moon pool* ao algoritmo genético foi converter o gerador de malha em uma função do MATLAB.

Anteriormente, o código da malha era uma rotina principal no MATLAB, e os parâmetros geométricos para a construção do *moon pool* eram definidos explicitamente dentro do corpo da rotina, como na Figura 12.

Figura 12 – Fluxograma da ferramenta de geração de malha como rotina principal.



Fonte: Elaborada pelo autor.

Para que o *moon pool* possa ser gerado de acordo com os parâmetros definidos pelo AG, a rotina foi convertida numa função que recebe os parâmetros de interesse como variáveis de entrada, representada esquematicamente na Figura 13. O algoritmo também foi adaptado para criar a malha do indivíduo de acordo com o padrão de leitura das análises hidrodinâmicas.

Figura 13 – Fluxograma da ferramenta de geração de malha como função.



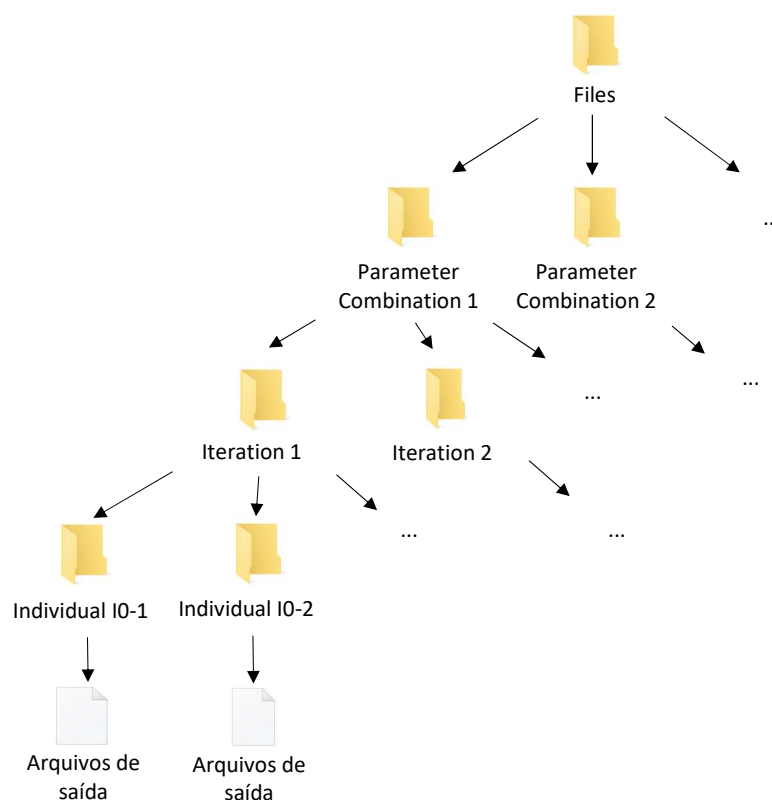
Fonte: Elaborada pelo autor.

Outra etapa da integração do algoritmo genético foi a criação de uma estrutura de pastas para manter um histórico de todos os indivíduos que foram gerados e avaliados durante a otimização.

Essa estrutura, contida dentro da pasta *Files*, é separada no primeiro nível pelas combinações de parâmetros da varredura do algoritmo genético (ou seja, combinação

de comprimento e largura do *moon pool*), no segundo nível pelas iterações realizadas para aquela dada combinação de parâmetros e no terceiro pelos indivíduos gerados e analisados em cada iteração. A estrutura das pastas salvas está representada na Figura 14.

Figura 14 – Estrutura das pastas de registro da otimização.



Fonte: Elaborada pelo autor.

Os arquivos armazenados nessas pastas incluem: os parâmetros geométricos do *moon pool*, a malha do indivíduo no formato de leitura do código hidrodinâmico, a mesma malha no formato de visualização pela ferramenta CAD (*Computer Aided Design*) do *software* Gmsh<sup>1</sup>, os resultados das análises hidrodinâmicas e os dados do gráfico radar crítico para os critérios de operabilidade analisados.

O código de análise hidrodinâmica é uma simplificação do código utilizado por Michima e Kawabe (2014). Como decorrência dessa simplificação, o algoritmo perdeu a capacidade de avaliar o momento fletor incidente na seção mestra do casco, portanto esse critério teve que ser suprimido da otimização.

De qualquer forma, o algoritmo é invocado como um programa executável à parte. Ele é chamado pela rotina principal através de um comando simples (a

<sup>1</sup> Software de código aberto para geração de malhas de elementos finitos.

saber, `system(3Dmotion.Moonpool.REGULAR-IRREGULAR-beta6.exe)`) toda vez que é necessário executar a análise de um indivíduo. Quando o programa termina a execução, os arquivos de output são movidos para a pasta do indivíduo correspondente.

Esse tratamento modular permite uma relativa independência entre o algoritmo de otimização e o de avaliação hidrodinâmica: este último pode ser modificado ou mesmo substituído, ainda assim, mantida a forma do arquivo texto contendo os resultados das simulações, a integração deverá continuar funcionando conforme aqui exposto.

Durante a compilação do código fonte desse módulo, foi identificado, através de uma ferramenta de depuração, que a rotina por vezes tentava acessar memória não alocada, o que causava erros de violação de acesso. De fato, mesmo depois de compilado, o código continuou apresentando esses erros, o que forçou a implementação de uma instrução `while` para continuar tentando executá-lo até que ele seja finalizado com sucesso.

Quando a análise hidrodinâmica é terminada e os arquivos de output estão devidamente impressos, a função `Eval_Data` processa os dados relevantes para a avaliação de operabilidade, calculando as coordenadas do gráfico de radar do indivíduo analisado para cada ângulo de incidência da onda. O indivíduo então recebe uma nota de acordo com o que seria a área resultante da interseção dos critérios do seu gráfico de radar, além de dois critérios de desempate, a serem consultados caso necessário. Esses são os três parâmetros utilizados pelo algoritmo genético na escolha dos melhores indivíduos, podendo ser armazenados caso o indivíduo em questão seja selecionado para continuar na otimização. Os critérios de operabilidade e o processamento dos resultados das análises hidrodinâmicas propriamente ditos serão explicados em seção posterior.

Na implementação do modelo aqui descrito, o cálculo da nota de operabilidade está em estágio preliminar: a área de operabilidade segura é calculada tomando um polígono definido com base na altura máxima aceitável, levando em conta todos os critérios, para cada eixo do gráfico radar. Destaca-se que o resultado é diferente de um polígono definido pela interseção entre os formatos dos gráficos dos diversos critérios de operabilidade, como proposto na área hachurada da [Figura 7](#), pois os vértices desse polígono não estão necessariamente localizados nos eixos do gráfico. Uma metodologia mais precisa para o cálculo da área real de operabilidade está em desenvolvimento para integração futura ao código.

### 3.3 Compatibilização da malha para a análise hidrodinâmica

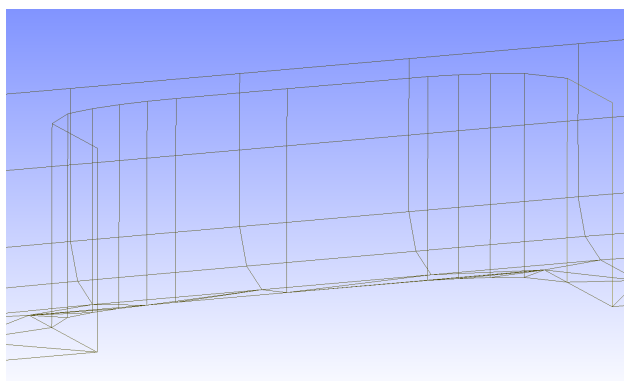
Uma série de modificações foi realizada no código gerador da malha do casco com *moon pool* integrado durante a etapa de integração entre os módulos. Algumas das alterações ajustaram o processo de geração dessa malha, enquanto outras a compatibilizaram com o padrão de entrada da análise hidrodinâmica.

Os ajustes realizados na geração de malha são listados abaixo:

Corrigida uma inconsistência na identificação numérica dos painéis da malha gerada pelo código. Na versão original, os painéis seguiam a sequência correta de numeração até certo ponto e depois voltavam para um número anterior, continuando a partir dele. Agora a indexação numérica mantém uma sequência contínua.

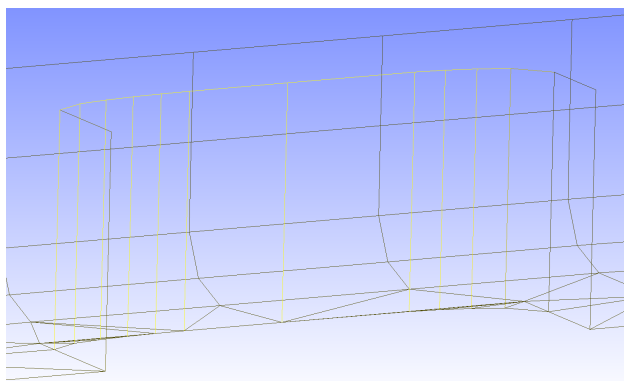
A construção dos painéis que compõem o *moon pool* foi revisada de forma a garantir que as suas normais apontem na direção da coluna d'água, como é necessário para a análise hidrodinâmica. Originalmente, as normais desses elementos apontavam no sentido externo ao *moon pool*, assim como as normais dos painéis que compõem o costado. Comparando a [Figura 15](#), extraída antes da alteração, com a [Figura 16](#), extraída com o código atualizado, verifica-se a coloração amarela nos painéis internos ao *moon pool* na segunda figura, o que indica que as suas normais apontam para dentro da água.

Figura 15 – Painéis internos ao *moon pool* antes da modificação.



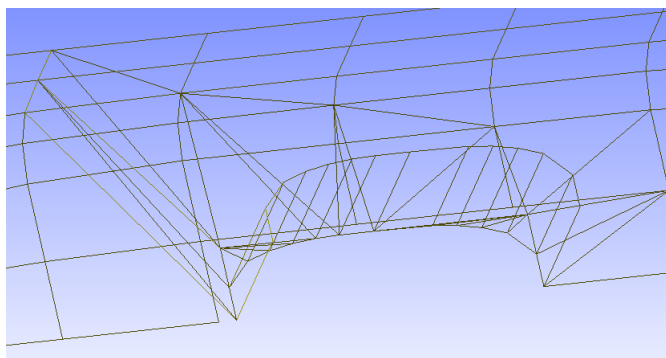
Fonte: Elaborada pelo autor.

Também foi necessário flexibilizar a criação da malha, habilitando o algoritmo a lidar com todos os tamanhos de *moon pool* que seriam utilizados na otimização. O algoritmo de construção da malha do *moon pool* possui uma limitação na forma de lidar com a região de interface com o casco: os pontos da interface precisam ser declarados explicitamente dentro do código. Isso significa que, quando o *moon pool* cresce ou

Figura 16 – Painéis internos ao *moon pool* após a modificação.

Fonte: Elaborada pelo autor.

diminui o suficiente para mudar os seus painéis de contorno no casco, a interface passa a conectar os nós de forma inadequada, conforme se observa na [Figura 17](#).

Figura 17 – Painéis de interface do *moon pool* com o casco antes da modificação.

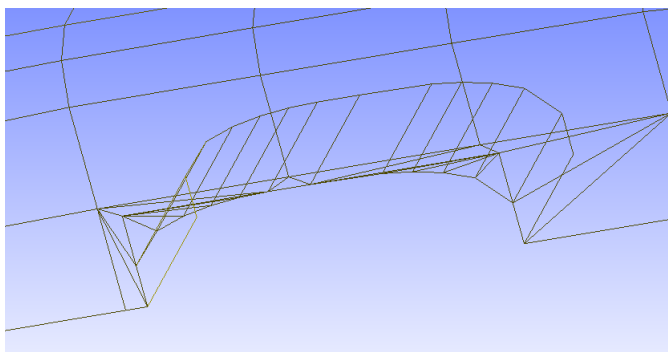
Fonte: Elaborada pelo autor.

Para contornar o problema, o código foi modificado para lidar com os diferentes tamanhos de *moon pool* da otimização numa abordagem caso a caso.

Foram criados quatro casos `if` dentro da função responsável pela construção desses elementos (`Geom_Elementos`), de acordo com as faixas de largura e comprimento do *moon pool* (2 faixas de largura x 2 faixas de comprimento = 4 casos). Para cada caso, ajustaram-se os pontos de fronteira manualmente dentro do código. No processo de reescrever tais pontos, alguns painéis da interface se colapsaram e viraram linhas, as quais foram apagadas. Depois, foram realizadas alterações para tentar garantir que, mesmo variando os parâmetros do *moon pool*, não haveria painéis se sobrepondo ou com a normal apontando para o sentido errado (isto é, para o interior do casco). A [Figura 18](#) ilustra como se deu a construção do mesmo *moon pool*

apresentado anteriormente, após os ajustes introduzidos.

Figura 18 – Painéis de interface do *moon pool* com o casco após a modificação.



Fonte: Elaborada pelo autor.

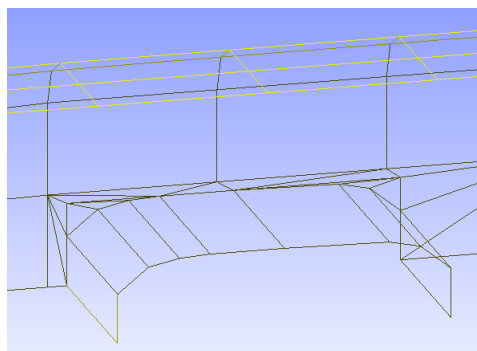
Neste ponto, fica evidente que seria bastante útil o desenvolvimento de um método capaz de lidar com malhas para qualquer tamanho de *moon pool*, em qualquer casco de entrada. Tal generalização permitiria que o código funcionasse em uma maior diversidade de contextos além da otimização aqui proposta, com menores necessidades de adaptação.

A última melhoria realizada na malha foi a inclusão do parâmetro  $m2$ , conforme exposto anteriormente na Figura 9, que havia sido acidentalmente omitido da construção do *moon pool*.

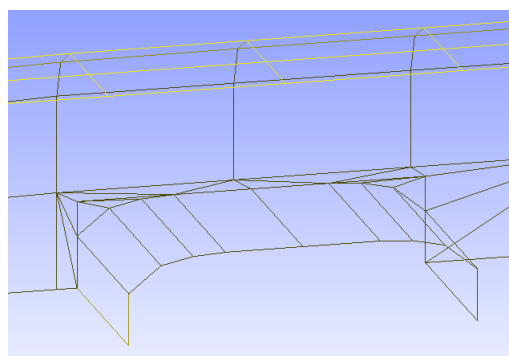
Antes do ajuste, o penúltimo painel da quina elíptica do lado de proa do *moon pool* era ligado diretamente à linha de centro da geometria, não existindo o painel longitudinal  $m2$  indicado na Figura 9. A Figura 19 e a Figura 20 mostram como o mesmo *moon pool* foi construído pelo código antes e depois do ajuste, respectivamente. Comparando-as, nota-se o painel faltante, cuja ausência fazia com que o *moon pool* deixasse de ser simétrico na malha gerada pelo código original.

Além das modificações listadas acima, implementou-se a conversão da malha do casco com *moon pool* para a forma de entrada da análise hidrodinâmica. Originalmente, as malhas geradas estavam no formato adequado para visualização na *engine* CAD do *software* Gmsh, portanto foi necessário adicionar linhas de código em diversas funções para que elas fossem capazes também de escrever essas malhas no padrão de entrada do código Fortran.

A primeira etapa realizada nessa conversão foi a adaptação do sistema de coordenadas e da ordem de declaração dos pontos para o novo padrão adotado. Na Figura 21 se observa o padrão válido para a malha de entrada no MATLAB, sem *moon pool*, e na Figura 22, o adotado no código Fortran.

Figura 19 – Quina do *moon pool* antes do ajuste introduzido.

Fonte: Elaborada pelo autor.

Figura 20 – Quina do *moon pool* após o ajuste introduzido.

Fonte: Elaborada pelo autor.

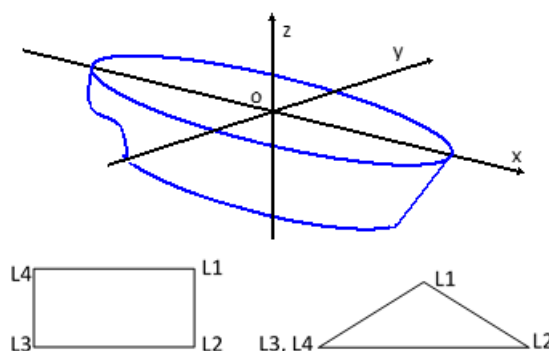
A segunda etapa foi a adição de código para criar automaticamente os elementos da superfície livre do *moon pool*, posicionados no topo da abertura, como uma “tampa”.

A última etapa da compatibilização da malha de entrada foi a reestruturação do seu padrão para o do código de análise. A Figura 23 resume essa adequação esquematicamente. À esquerda, observa-se a malha para visualização no Gmsh,<sup>2</sup> e à direita, a malha de leitura das análises para um indivíduo qualquer. Enquanto o arquivo da esquerda lista todos os pontos e elementos em sequência, sem distinção, o da direita é separado em três grupos: casco, *moon pool* e superfície livre, cada um com numeração de pontos e painéis independente.

Para realizar a reestruturação, o código passou por mudanças para listar os pontos e elementos em uma ordem que permitisse uma delimitação desses dados em blocos. Alguns pontos, como se observa, são listados como parte de dois blocos ao mesmo tempo. Isso acontece com os pontos de interface, ou seja, os pontos que fazem

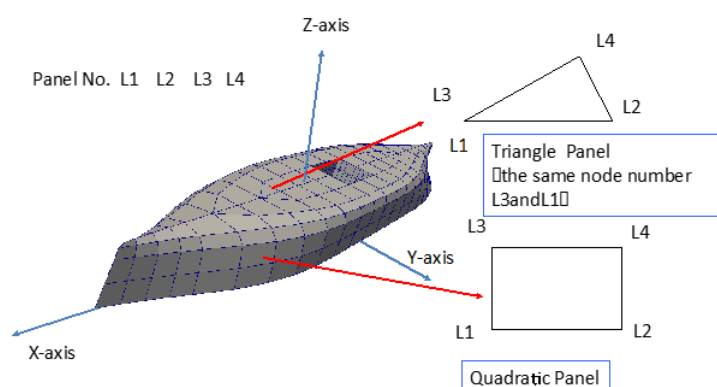
<sup>2</sup> <<http://gmsh.info/doc/texinfo/#MSH-ASCII-file-format>>

Figura 21 – Convenções adotadas na malha do casco de entrada no código MATLAB.



Fonte: Elaborada pelo Eng. PhD. Hiroshi Kawabe.

Figura 22 – Convenções adotadas na malha do casco de entrada na análise hidrodinâmica.



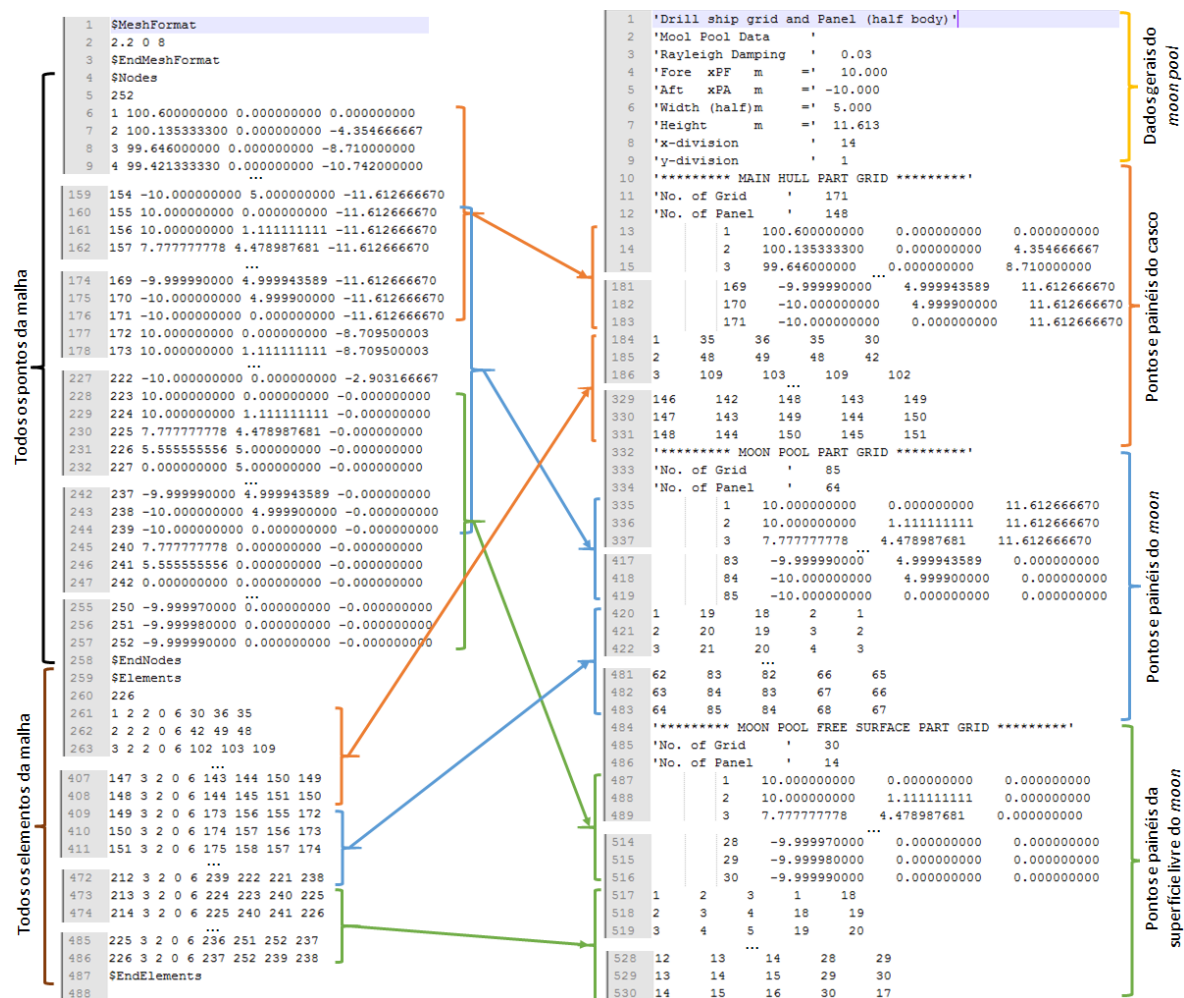
Fonte: Elaborada pelo Eng. PhD. Hiroshi Kawabe.

parte de dois grupos, a saber: pontos da base do *moon pool*, que também fazem parte do casco; e pontos do topo do *moon pool*, que também fazem parte da superfície livre.

Enquanto esses pontos e painéis são copiados para o novo formato, eles também precisam ser reindexados para adotar uma nova numeração, compatível com o grupo ao qual eles estão sendo adicionados. Isso inclui modificar o número de identificação dos pontos que definem cada painel, como mostrado na [Figura 23](#).

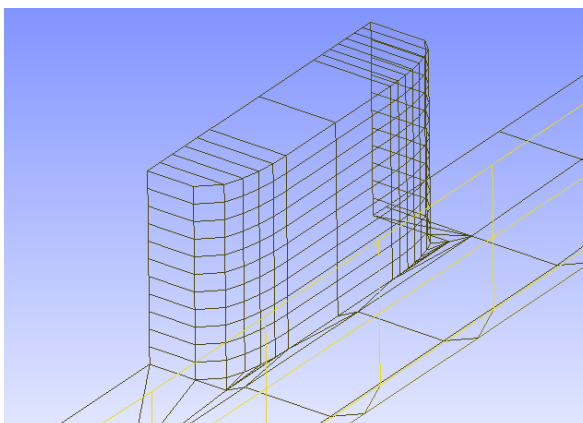
Além da compatibilização da malha propriamente dita, foi tomada a iniciativa de implementar o refinamento paramétrico da seção prismática do *moon pool*, controlável pelo usuário. A [Figura 24](#) mostra a malha de um *moon pool* gerado com esse parâmetro igualado a 11, ou seja, com 11 divisões de igual altura na seção prismática.

Figura 23 – Reestruturação da malha de entrada da análise hidrodinâmica.



Fonte: Elaborada pelo autor.

Figura 24 – Moon pool com refinamento de onze divisões em sua seção prismática.



Fonte: Elaborada pelo autor.

### 3.4 Critérios de operabilidade e opções da otimização

O tratamento dos dados provenientes da análise hidrodinâmica foi quase que totalmente executado conforme a estratégia apresentada por [Michima e Kawabe \(2014\)](#).

A única modificação em relação ao lá proposto foi a omissão de um dos quatro critérios originalmente definidos, o de resistência estrutural por momento fletor. Isso se deu porque a versão do código disponível no momento do desenvolvimento deste trabalho ainda não era capaz de fornecer os parâmetros necessários para avaliar esse quesito. No entanto, os outros três critérios (força de deriva, transbordo no *moon pool* e limite de compensação de *heave*) foram incorporados à otimização, e não seria difícil estender a metodologia apresentada a seguir para incluir um novo critério, assim que possível.

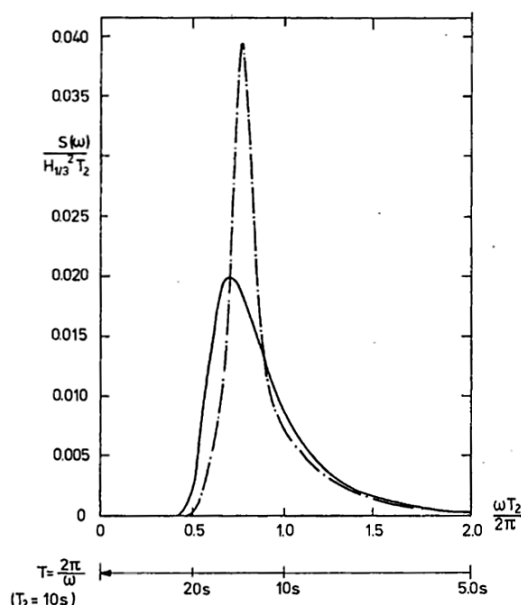
Em caso de empate entre as notas de dois indivíduos, o desempate ocorre conforme proposto por [Cavalcante \(2015\)](#):

- 1- Será considerado mais adequado o moonpool que apresentar o maior valor, no mapa de operabilidade, para 90°.
- 2- Em caso de empate no quesito (1), será considerado mais adequado o moonpool que apresentar o maior valor, no mapa de operabilidade, para 180°.

Diversos dos parâmetros da otimização são modificáveis pelo usuário. Estão disponíveis três famílias de espectros de distribuição de energia na onda: JONSWAP, IACS e Bretschneider. Os três são baseados no espectro de Pierson-Moskowitz ([PIERSON; MOSKOWITZ, 1964](#)). Enquanto os espectros IACS e Bretschneider representam estados de mar desenvolvidos e são bastante similares entre si, o JONSWAP foi construído com base em estudos no Mar do Norte, e representa um estado de mar que não se desenvolve plenamente, concentrando a energia de onda em uma frequência de pico. A [Figura 25](#) mostra uma comparação entre os formatos de um espectro Pierson-Moskowitz e um espectro JONSWAP.

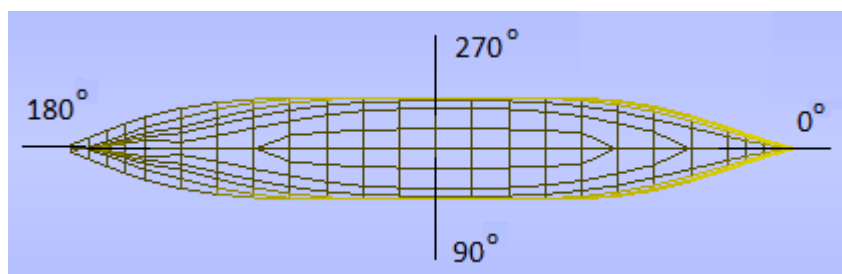
Cada um dos espectros está disponível na variação *short crested* e *long crested*, sendo a primeira correspondente a um estado de mar menos desenvolvido (ondas de cristas curtas) e a segunda a um estado de mar mais desenvolvido (ondas de cristas longas). Isso nos dá uma combinação de 3 famílias x 2 variações = 6 tipos de espectro. As respostas hidrodinâmicas de interesse são calculadas, por padrão do código, para 15 espectros dentro de cada um dos seis tipos apresentados. Cada espectro correspondente a uma combinação de período significativo, definido por um inteiro de 4 a 18 segundos, inclusive; e por um dentre 12 ângulos de incidência de ondas (0°, 30°, 60°, e assim por diante), conforme convenção direcional indicada na [Figura 26](#).

Figura 25 – Comparação entre um espectro Pierson-Moskowitz e um espectro JONSWAP.



Fonte: Faltinsen (1993).

Figura 26 – Convenção direcional dos ângulos de incidência de onda (proa para a direita).



Fonte: Cavalcante (2015).

O usuário pode executar a otimização incluindo ou ignorando qualquer um desses seis tipos de espectro, todos avaliados para uma combinação qualquer de períodos significativos, também à escolha do usuário. A nota utilizada como critério da otimização será sempre a menor área de gráfico de radar dentre todos os espectros e períodos significativos escolhidos, pois ela representa a situação crítica de operação.

Essas opções foram pensadas de forma a permitir que o usuário opte entre uma otimização mais especialista, voltada ao estado de mar característico da região de operação, ou mais generalista, em que se quer otimizar o casco para uma operação em várias regiões e estados de mar de interesse.

Em termos computacionais, a automatização da extração dos dados de interesse (respostas hidrodinâmicas) foi bastante direta devido à boa organização dos arquivos de saída do código. A localização desses dados é sempre anunciada pela mesma linha de cabeçalho cada vez que eles se repetem ao longo do arquivo texto, e a ordem dos resultados de acordo com o tipo de espectro é mantida inalterada em todos os arquivos.

Isso permitiu armazenar os números das linhas de localização de todos os resultados em um vetor e escolher, através desse vetor, quais resultados consultar no cálculo da nota de operabilidade, a depender das preferências de espectros definidas pelo usuário. As linhas de interesse, identificadas pelos índices armazenados no vetor, são consultadas, e os resultados transcritos em cada uma delas são convertidos em alturas significativas para as direções angulares do gráfico de radar. A nota do indivíduo para um dado espectro é calculada como a área do polígono definido pela menor altura significativa admissível em cada eixo do gráfico de radar traçado para esse espectro. A nota final atribuída ao indivíduo será a menor dentre todos os espectros escolhidos para a otimização.

### 3.5 Testes e resultados

Os parâmetros gerais das otimizações executadas foram mantidos conforme sugerido por [Cavalcante \(2015\)](#) a saber: varredura ingênua percorrendo *moon pools* de comprimento entre 20 e 26 metros e largura entre 10 e 14 metros, 2 avaliações por iteração dessa varredura, sendo a primeira com 10 indivíduos novos e a segunda com os 5 melhores da primeira avaliação mais 5 mutações desses indivíduos.

As otimizações foram realizadas para o estado de mar JONSWAP *short crested* com período significativo  $T_s$  de 6 segundos, apontado como um dos mais críticos em testes preliminares. Os limites aceitáveis de operabilidade foram definidos conforme a [Tabela 2](#).

Tabela 2 – Limites aceitáveis dos critérios de operabilidade utilizados na otimização.

| Item                           | Critério             | Valor específico | Tolerância                |
|--------------------------------|----------------------|------------------|---------------------------|
| Transbordo do <i>moon pool</i> | Borda-livre          | 4 m              | 15 ocorrências em 3 horas |
| Compensador de <i>Heave</i>    | Curso do compensador | 3,5 m            | 15 ocorrências em 3 horas |
| Deriva                         | Potência DPS total   | 34.260 PS        | -                         |

Fonte: Elaborada pelo autor.

As premissas acima descritas foram bastante similares às definidas no arquivo `Optimization_Parameters.dat`, disponível no [Apêndice A](#), a única diferença sendo a

adição de diversos períodos significativos para análise no arquivo apresentado, a fim de ilustrar ao leitor como esse tipo de entrada deve ser feita.

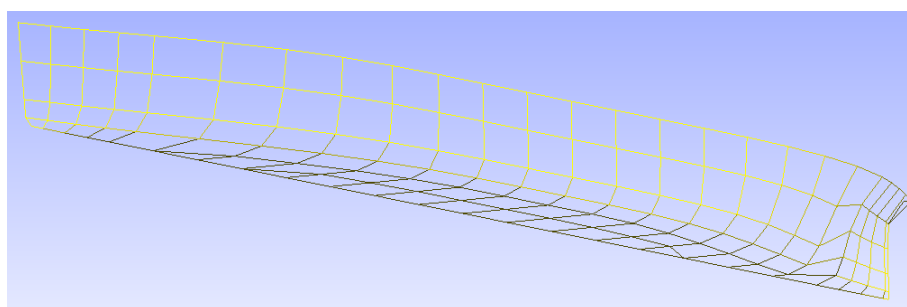
O casco utilizado na otimização, que também foi mantido de (CAVALCANTE, 2015), é um Série 60 adaptado para se assemelhar ao casco de um *drillship*, passando a ter as dimensões principais listadas na Tabela 3 e a malha mostrada na Figura 27. A Série 60 é uma família de cascos sistemática, isto é, uma família de cascos obtidos a partir da variação sistemática de um conjunto parâmetros de forma. Os cascos pertencentes a séries sistemáticas costumam ser estudados extensivamente e possuem seus resultados experimentais divulgados na literatura. No entanto, não foi a disponibilidade dos dados experimentais que motivou a escolha de um casco baseado nas características da série, afinal, eles não seriam de muito uso para este trabalho já que não levam em conta os efeitos causados pela presença do *moon pool*.

Tabela 3 – Dimensões principais do casco.

| Dimensão                                     | Valor         |
|----------------------------------------------|---------------|
| $L_{pp}$ - comprimento entre perpendiculares | 203,4 m       |
| $B$ - boca                                   | 29,0 m        |
| $d$ - calado                                 | 11,6 m        |
| $\Delta$ - deslocamento                      | 45.912,83 ton |

Fonte: Elaborada pelo autor.

Figura 27 – Malha do casco utilizado para a otimização (proa para a esquerda).



Fonte: Elaborada pelo autor.

Por outro lado, enquanto existem exemplos de resultados empíricos de *drillships* reais na literatura, não foi possível obter o plano de linhas de nenhum deles, levando à escolha pela forma baseada na Série 60. Ela continuou sendo utilizada desde então, e foi mantida neste trabalho em particular porque não é parte do seu escopo realizar adequações no casco da otimização. Isto também implica que todo o acoplamento do *moon pool* paramétrico, conforme exposto na seção 3.3, foi realizado especificamente para essa malha.

Dois outros conjuntos de parâmetros, referentes às características do espectro de mar e às propriedades físicas do casco (incluindo localização do centro de gravidade e momentos de inércia) foram mantidos conforme fornecidos junto ao código de simulação hidrodinâmica, não sofrendo qualquer tipo de modificação.

### 3.5.1 Avaliação da integração

O código da simulação hidrodinâmica utilizado neste trabalho não pode ser disponibilizado publicamente, como foram os códigos de otimização e acoplamento do *moon pool* ao fim deste texto. Infelizmente, isto impede o leitor de executar por conta própria a otimização proposta utilizando a ferramenta compatibilizada. Esta subseção foi incluída então para dirimir quaisquer dúvidas a respeito da funcionalidade da interface entre as ferramentas envolvidas na otimização, com evidências extraídas das duas execuções realizadas.

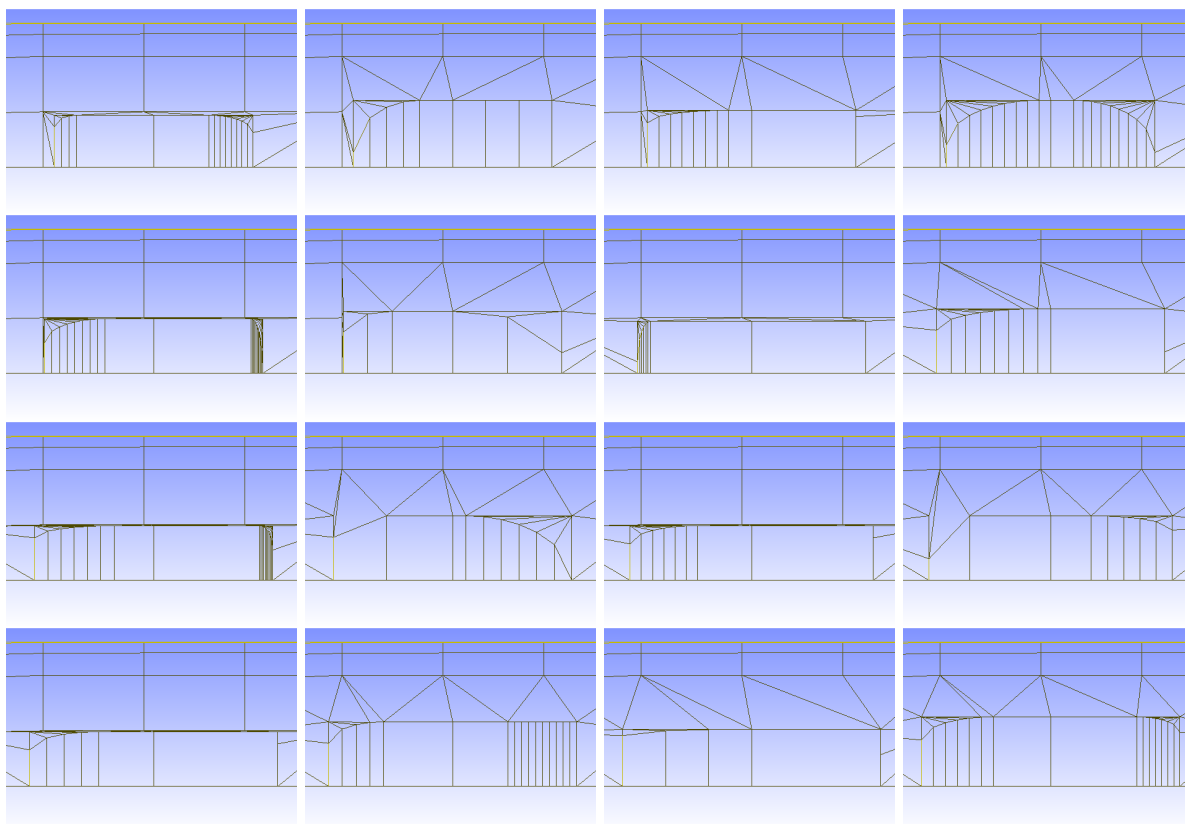
As otimizações foram executadas em um laptop modelo comercial Acer E5-571G-50K9, com processador Intel Core i5-4210U, com frequência de processamento entre 1,7 e 2,4 GHz, 8 GB de memória RAM DDR3 e disco rígido HDD. Os arquivos de registro da primeira otimização, incluindo respostas hidrodinâmicas, malhas e arquivos texto contendo parâmetros do casco, totalizaram aproximadamente 33 GB de dados. Sua execução durou aproximadamente 63 horas e 57 minutos.

A parte da varredura ingênua da otimização consistiu em 72 combinações de dimensões principais do *moon pool*. Para cada uma dessas combinações, foram realizadas iterações até que a operabilidade convergisse para um valor ótimo. Para as duas otimizações, a quantidade de iterações realizadas antes de alcançar a convergência variou entre um mínimo de 3 e um máximo de 8 iterações, sendo que 3 é o valor mínimo de iterações definido pelo código, a fim de evitar uma convergência prematura com 2 iterações. Não existe quantidade máxima de iterações definida (truncamento), então o algoritmo genético necessariamente alcançará a convergência antes de o código prosseguir para um novo laço da varredura ingênua.

Para as duas execuções, a quantidade de indivíduos gerados e analisados, contando repetições, superou 5.000. A avaliação de todos foi concluída com sucesso pelo código, isto é, sem retornar erros além das já mencionadas violações de acesso. Alguns exemplos dos *moon pools* da segunda execução estão ilustrados na [Figura 28](#).

A observação da figura confirma a variedade de geometrias analisadas pela otimização, mas também evidencia a necessidade de melhora da qualidade das malhas em trabalhos futuros. Isto inclui o refino do tamanho dos painéis excessivamente grandes na superfície livre e a reformulação dos elementos demasiadamente delgados na região de interface com o casco. Ambas as características podem afetar negativamente

Figura 28 – Exemplos de *moon pools* analisados na segunda otimização (proa para a direita).



Fonte: Elaborada pelo autor.

a precisão dos resultados numéricos fornecidos pelo código.

A quantidade de painéis na malha de cada indivíduo obviamente variou dependendo da geometria do seu *moon pool*, porém a grande maioria das malhas possuía entre 200 e 250 painéis, todas geradas com 4 divisões na seção prismática do *moon pool*.

### 3.5.2 Resultados da primeira execução

O histórico das notas dos ótimos para cada uma das 72 combinações de dimensões principais é exposto na Figura 29. Esses ótimos serão a partir daqui denominados de “campeões”, a fim de não serem confundidos com o ótimo geral da execução. A nota de operabilidade de cada indivíduo é a área segura do seu gráfico de radar.

É possível perceber na Figura 29 dois aspectos do comportamento das notas de operabilidade ao longo das iterações. Primeiro, uma tendência de aumento estável da performance dos indivíduos no decorrer da varredura ingênua (combinação de parâmetros). Segundo um padrão aparentemente periódico se repetindo ao longo

Figura 29 – Histórico das notas de operabilidade dos campeões da primeira execução.



Fonte: Elaborada pelo autor.

dessa varredura.

A visualização na Figura 30, que exibe as notas de cada combinação de parâmetros com uma escala de cores, permite uma melhor interpretação do comportamento observado. A leitura, quando comparada com a Figura 29, se dá no sentido convencional, sendo a Combinação de Parâmetros 1 a célula superior à esquerda. Dessa forma, a tendência de aumento de performance estável pontuada anteriormente corresponde à melhora linha a linha, e a tendência periódica, à melhora ao longo de cada linha.

Figura 30 – Visualização das notas dos campeões da primeira execução.

|    |       | I2       |          |          |          |          |          |          |          |          |
|----|-------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
|    |       | 5,00     | 5,25     | 5,50     | 5,75     | 6,00     | 6,25     | 6,50     | 6,75     | 7,00     |
| I1 | 10,00 | 11,54569 | 11,85636 | 12,34021 | 12,56561 | 12,78305 | 12,58012 | 12,77673 | 12,77218 | 14,71188 |
|    | 10,50 | 11,58345 | 12,12842 | 12,00749 | 12,9066  | 12,83522 | 12,45566 | 13,17037 | 13,18789 | 13,64466 |
|    | 11,00 | 11,51157 | 12,23161 | 11,91351 | 13,18091 | 13,24366 | 12,97739 | 13,49414 | 13,16803 | 13,76443 |
|    | 11,50 | 12,22648 | 12,68289 | 13,12121 | 13,00198 | 13,50847 | 13,31019 | 13,51139 | 13,75931 | 14,06942 |
|    | 12,00 | 12,83461 | 12,93727 | 12,8665  | 13,5132  | 13,61427 | 13,68388 | 13,6689  | 14,01186 | 14,15655 |
|    | 12,25 | 12,75253 | 13,18771 | 13,52803 | 13,22317 | 13,01514 | 13,62454 | 13,94449 | 13,96056 | 14,24809 |
|    | 12,50 | 12,34144 | 13,24297 | 13,92401 | 13,47964 | 13,73398 | 13,59233 | 13,80412 | 14,1399  | 14,40506 |
|    | 13,00 | 13,14771 | 12,53528 | 14,15079 | 13,544   | 13,92553 | 13,98769 | 14,22386 | 14,53675 | 14,64222 |

Fonte: Elaborada pelo autor.

Da figura, percebemos que, em geral, *moon pools* maiores apresentam melhor

desempenho de operabilidade. Nesse sentido, o indivíduo ótimo, cuja nota está destacada em negrito, mostra-se como exceção parcial dessa regra, por possuir o menor comprimento dentre a faixa de varredura, apesar de possuir a maior largura dentre as disponíveis. Esse tipo de exceção à tendência principal, observada também em outras células da visualização, pode estar relacionada à combinação das influências dos modos de pistão e *sloshing* do fluido, inclusive pelo fato das frequências de ressonância destes modos não serem necessariamente coincidentes.

O *moon pool* do indivíduo ótimo apresentou as características geométricas dadas pela Tabela 4, conforme a parametrização explanada na subseção 2.2.2. A malha correspondente a esse *moon pool* é exibida na Figura 31.

Tabela 4 – Parâmetros geométricos do *moon pool* ótimo da primeira execução.

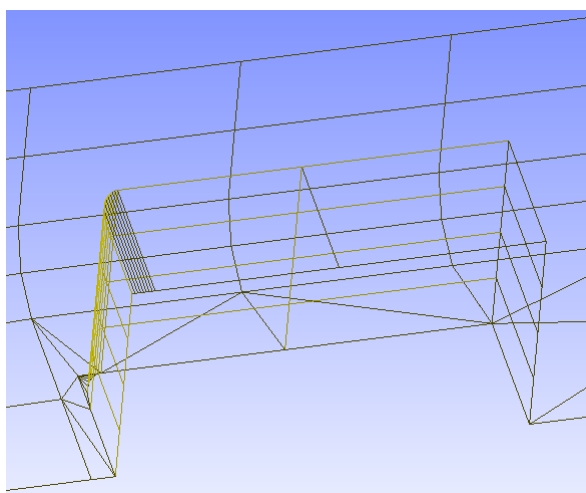
| Parâmetro | Valor  |
|-----------|--------|
| $l_1$     | 10,000 |
| $l_2$     | 7,000  |
| $m_1$     | 0,000  |
| $m_2$     | 10,000 |
| $m_3$     | 8,889  |
| $m_4$     | 4,667  |
| $nf$      | 1      |
| $nt$      | 8      |

Fonte: Elaborada pelo autor.

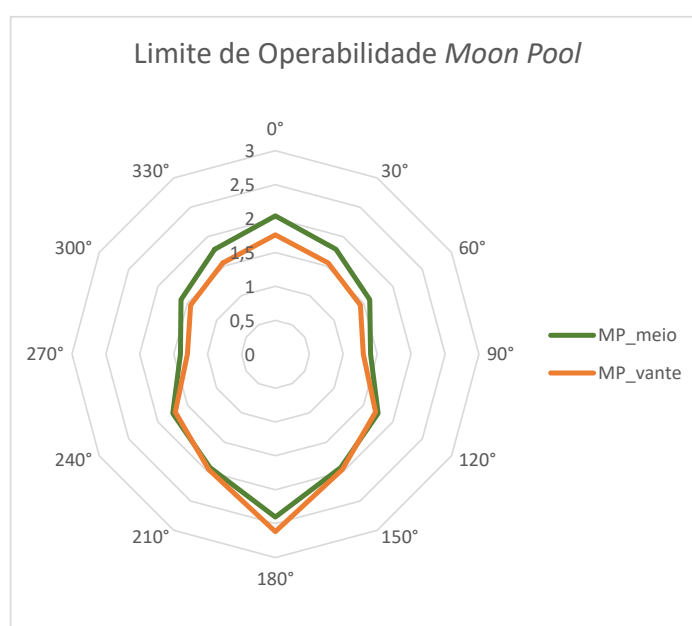
Apesar de o *moon pool* ótimo ser quase retangular, aproximando-se do formato comum para esse tipo de abertura, não podemos considerá-lo trivial. Afinal, comprimento e a largura de operabilidade ótima seriam desconhecidos, não fosse a otimização executada.

A Figura 32 expõe o gráfico radar de operabilidade do campeão da otimização para os critérios de transbordo na região de meio e vante do *moon pool*, e a Figura 33 expõe a combinação desses critérios em um gráfico de radar de limite crítico de operabilidade. Os outros critérios considerados não aparecem plotados nesses gráficos porque não se mostraram críticos para a operabilidade da embarcação ótima, apresentando valores limite de altura significativa de onda muito superiores à escala do eixo radial.

Dos gráficos, percebe-se que os limites de ambos os critérios são muito similares entre si. O *moon pool* como um todo é mais suscetível a transbordos causados por ondas de proa que por ondas de popa. Provavelmente, caso o casco e o *moon pool* fossem simétricos em relação ao plano da seção média, essa diferença nos limites dos

Figura 31 – Malha do *moon pool* ótimo da primeira execução.

Fonte: Elaborada pelo autor.

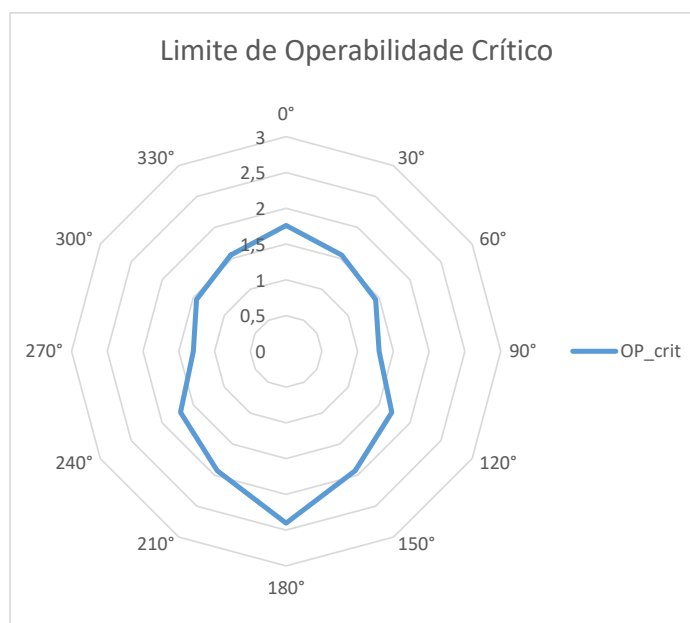
Figura 32 – Gráfico de operabilidade do *moon pool* ótimo da primeira execução.

Fonte: Elaborada pelo autor.

critérios não ocorreria. No entanto, é difícil precisar qual elemento dessa assimetria é responsável pelo fato observado sem realizar outros testes.

Nota-se também que, enquanto ondas de proa oferecem risco ligeiramente maior de transbordo para a região de vante que para a região de meio, o oposto acontece para ondas de popa. Este padrão pode ser gerado pelo movimento do fluido dentro do *moon pool*, pelo deslocamento relativo desse fluido em relação ao movimento global

Figura 33 – Gráfico de operabilidade do *moon pool* ótimo da primeira execução.



Fonte: Elaborada pelo autor.

do casco, ou por alguma composição entre estes dois movimentos, implicada por uma possível diferença de fase característica entre eles.

Tanto para a vante quanto para o meio do *moon pool*, as ondas que oferecem maior risco de transbordo são as de través (ângulo de incidência de 90°). Nesta direção, a diferença entre a onda aceitável para as duas regiões é mínima. Como a onda proveniente de través atinge igualmente ambas as regiões avaliadas e os efeitos viscosos da proximidade da parede de vante são pequenos, de acordo com Aalbers (1984), este comportamento é verossímil.

### 3.5.3 Resultados da segunda execução

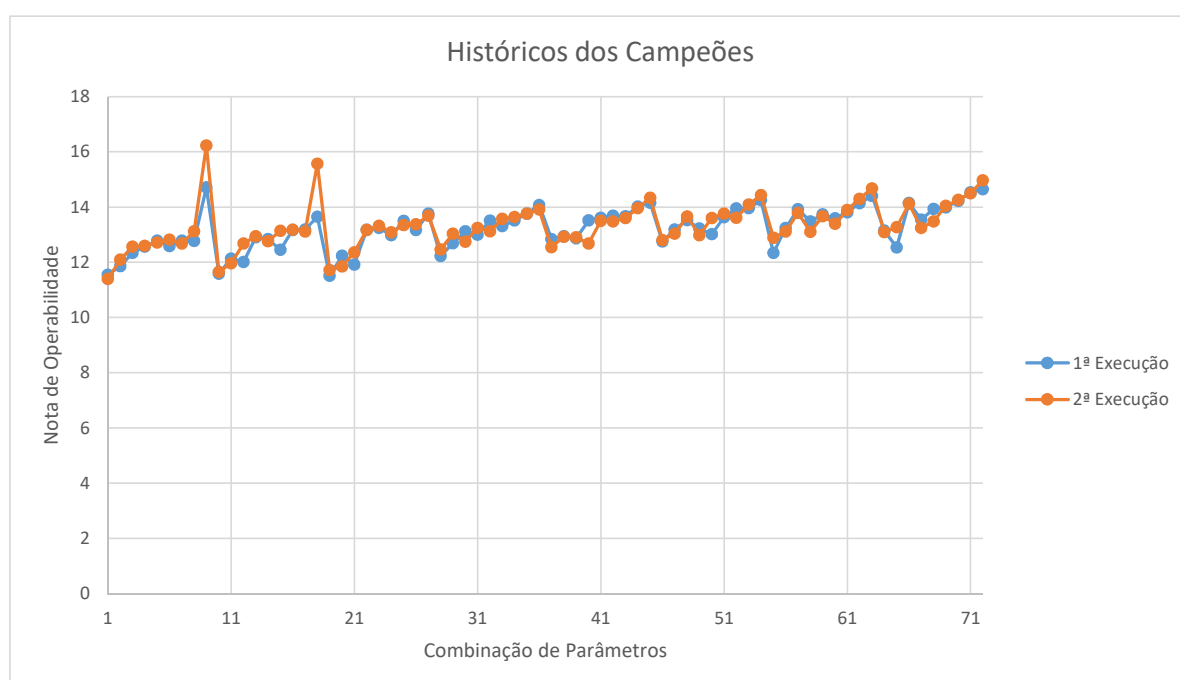
A segunda execução foi realizada com a finalidade de avaliar a aderência do algoritmo genético à solução encontrada na primeira execução. Como exposto, algoritmos genéticos são fortemente influenciados por fatores aleatórios na geração dos indivíduos avaliados, de forma que é difícil assegurar que a busca realizada foi de fato exaustiva.

Com essa premissa, a segunda otimização seguiu as mesmas definições gerais da primeira, porém com um novo elemento: na primeira iteração de cada combinação de parâmetros, um dos indivíduos gerados aleatoriamente foi substituído por um *moon pool* de dimensões proporcionais às do campeão da primeira otimização. Assim, enquanto se manteve aberta a possibilidade de encontrar um novo campeão de operabilidade

através de fatores aleatórios do AG, também se introduziu a chance de obter um indivíduo ótimo a partir de uma modificação ou derivação do primeiro.

Isto posto, o novo histórico das notas dos campeões pode ser observado na Figura 34, e a visualização correspondente na Figura 35. De fato, um novo ótimo de operabilidade foi encontrado, possuindo a mesma combinação de dimensões principais do indivíduo escolhido na execução anterior.

Figura 34 – Histórico das notas de operabilidade dos campeões da segunda execução.



Fonte: Elaborada pelo autor.

Figura 35 – Visualização das notas dos campeões da segunda execução.

|    |       | I2       |          |          |          |          |          |          |          |
|----|-------|----------|----------|----------|----------|----------|----------|----------|----------|
|    |       | 5,00     | 5,25     | 5,50     | 5,75     | 6,00     | 6,25     | 6,50     | 7,00     |
| I1 | 10,00 | 11,39665 | 12,09773 | 12,56206 | 12,59537 | 12,71425 | 12,8158  | 12,67508 | 13,1177  |
|    | 10,50 | 11,64796 | 11,9577  | 12,6812  | 12,93961 | 12,76117 | 13,14146 | 13,18117 | 13,1084  |
|    | 11,00 | 11,72015 | 11,84791 | 12,35709 | 13,17011 | 13,31549 | 13,08002 | 13,34467 | 13,37657 |
|    | 11,50 | 12,46305 | 13,03856 | 12,74083 | 13,23932 | 13,11737 | 13,57511 | 13,63449 | 13,76293 |
|    | 12,00 | 12,54318 | 12,92558 | 12,89996 | 12,67972 | 13,49257 | 13,4753  | 13,5989  | 13,95773 |
|    | 12,25 | 12,79841 | 13,03262 | 13,65979 | 12,97621 | 13,59585 | 13,76238 | 13,60684 | 14,09294 |
|    | 12,50 | 12,88579 | 13,10965 | 13,78711 | 13,10316 | 13,66826 | 13,3971  | 13,89498 | 14,29863 |
|    | 13,00 | 13,09022 | 13,27348 | 14,11957 | 13,23873 | 13,47577 | 14,04762 | 14,2599  | 14,49859 |
|    |       |          |          |          |          |          |          |          | 14,97039 |

Fonte: Elaborada pelo autor.

Através da visualização, identificamos os mesmos comportamentos das notas observados anteriormente. Para além disso, o indivíduo ótimo se destacou ainda mais

em relação aos campeões com outras combinações de dimensões principais, quando comparado com a execução anterior.

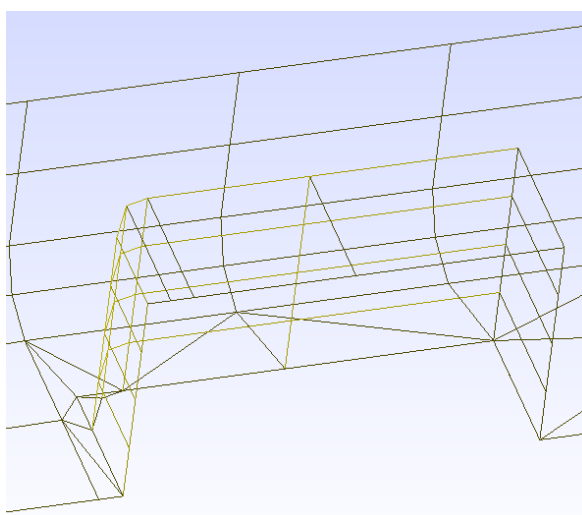
Da Tabela 5 e da Figura 36, podemos confirmar que a geometria ótima encontrada nesta segunda execução é uma derivação da encontrada na primeira, devido às visíveis semelhanças entre as duas.

Tabela 5 – Parâmetros geométricos dos *moon pools* ótimos da primeira e segunda execução.

| Parâmetro | 1ª Execução | 2ª Execução |
|-----------|-------------|-------------|
| $l1$      | 10,000      | 10,000      |
| $l2$      | 7,000       | 7,000       |
| $m1$      | 0,000       | 0,000       |
| $m2$      | 10,000      | 10,000      |
| $m3$      | 8,889       | 7,778       |
| $m4$      | 4,667       | 4,667       |
| $n.f$     | 1           | 1           |
| $nt$      | 8           | 2           |

Fonte: Elaborada pelo autor.

Figura 36 – Malha do *moon pool* ótimo da segunda execução.

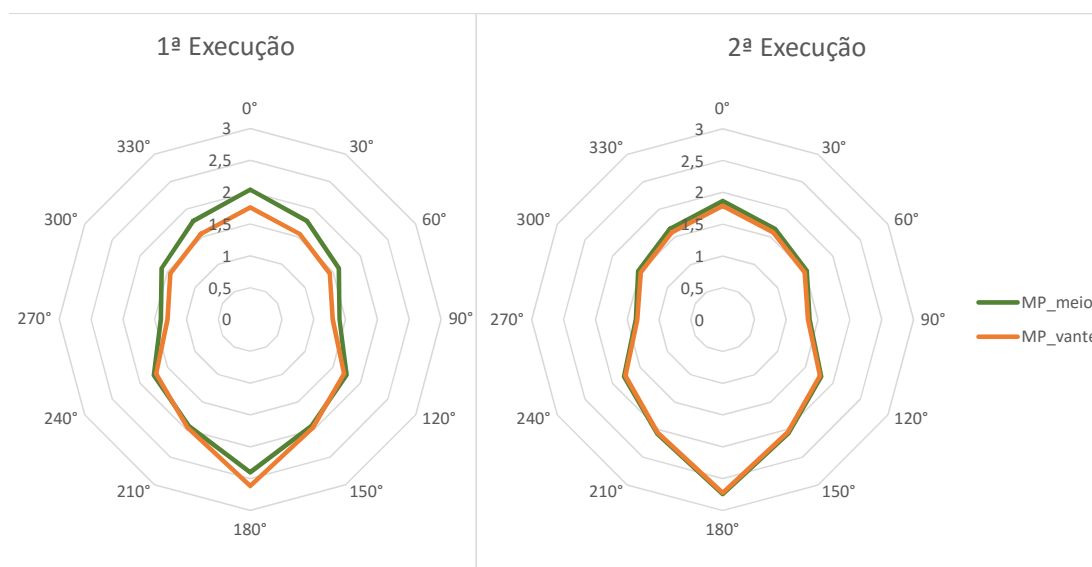


Fonte: Elaborada pelo autor.

Os formatos dos novos gráficos de radar, introduzidos na comparação da Figura 37, são bastante similares aos dos gráficos encontrados na execução anterior. O que era de se esperar, dado que a geometria do novo *moon pool* não sofreu mudanças

drásticas quando comparada à geometria do anterior. No entanto, para a segunda delas não se percebe diferenças significativas entre os critérios de transbordo na vante e no meio do *moon pool*, ao contrário da primeira. Este novo padrão pode ser explicado por um modo pistão dominante, por exemplo, o qual elevaria de forma homogênea o nível da coluna d'água dentro da abertura.

Figura 37 – Gráfico de operabilidade do *moon pool* ótimo da segunda execução.



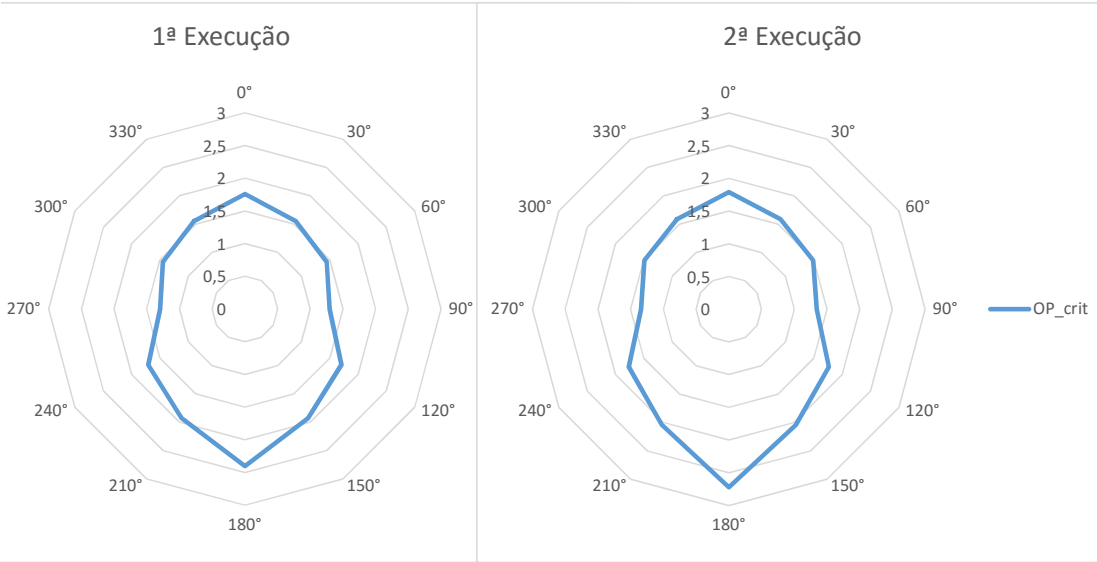
Fonte: Elaborada pelo autor.

Da Figura 36, a qual combina os critérios de transbordo na vante e no meio do *moon pools* para a primeira e segunda execução da otimização, percebemos que os ganhos de operabilidade se deram principalmente para ondas incidindo pela direção de popa, enquanto os limites de altura aceitável de onda em outras direções de incidência sofreram aumentos pequenos.

O fato de a segunda execução da otimização ter encontrado uma solução ao problema da operabilidade diferente da primeira indica que há espaço para melhora na busca realizada pelo algoritmo genético, afinal, idealmente a busca teria retornado o melhor indivíduo logo na primeira execução. Inclusive, não se pode descartar a hipótese de um *moon pool* com operabilidade ainda melhor que a do aqui apresentado vir a ser encontrado em uma futura otimização.

A semelhança entre a primeira e segunda solução sugere que o cruzamento entre indivíduos possui grande importância na busca por um ótimo global na otimização. Então, é recomendável garantir que ele seja bem dosado ao longo do processo, permitindo diversidade de indivíduos e evitando vícios na busca.

Figura 38 – Gráfico de operabilidade do *moon pool* ótimo da segunda execução.



Fonte: Elaborada pelo autor.

## 4 Conclusão

Este trabalho alcançou os objetivos aos quais se propôs, realizando com sucesso a compatibilização dos códigos computacionais da otimização. Assim, fica consolidada a base dessa plataforma: é possível executá-la para um conjunto de estados de mar, com três importantes critérios de operabilidade.

Todo o código desenvolvido anteriormente provou-se de grande valia para alcançar esse objetivo. O entendimento sobre o processo de criação e manipulação dos painéis foi importante para melhorar o código de integração entre as malhas do casco e *moon pool*. O processo de integração do código previamente existente à leitura dos critérios de operabilidade foi bastante direto e o algoritmo genético convergiu sem dificuldades.

Como o código de cálculo hidrodinâmico utilizado neste trabalho ainda se encontra em estágio de desenvolvimento, é de se esperar que ele ainda passe por mudanças e validações no futuro. De qualquer forma, a otimização como um todo deve se integrar bem a qualquer módulo de análise hidrodinâmica, contanto que sejam mantidas as formas dos arquivos trocados entre os dois, ou seja, os arquivos de entrada e saída das análises executadas, e que o novo módulo possa ser chamado na forma de um programa executável.

A utilização da área segura do gráfico de operabilidade como nota da otimização foi bastante vantajosa, pois permitiu a condensação de critérios de operabilidade díspares, avaliados através dos extensos resultados estatísticos provenientes das simulações hidrodinâmicas, em um único índice de performance.

A partir deste ponto, os trabalhos devem se concentrar principalmente em aperfeiçoamentos do código desenvolvido até agora. A próxima seção detalha uma série de sugestões nesse sentido, algumas das quais são ideias complementares que surgiram ao longo do processo de desenvolvimento do código, enquanto outras são partes importantes para o projeto como um todo que não puderam ser implementadas devido a algum fator limitante que não dependia do autor ou não era pertinente ao escopo do presente trabalho.

Da mesma forma que as seções de desenvolvimento relatam detalhes do funcionamento do código, a seção de trabalhos futuros trata de cada ponto considerado relevante para a melhora do trabalho com certa minúcia. Intenciona-se com isso que este texto sirva como registro do estado e das limitações do código neste momento e como ponto de partida para qualquer pessoa, provavelmente futura colaboradora do projeto, que precise entender a lógica de funcionamento do mesmo para realizar

quaisquer alterações necessárias.

## 5 Sugestões para trabalhos futuros

No aspecto da geração das malhas numéricas, fica como sugestão o desenvolvimento de um método geral para a criação da interface entre as malhas do casco e do *moon pool*. O método atual cria os painéis dessa interface estaticamente, com os pontos de contorno designados através de suas respectivas identificações numéricas. Por esse motivo, ele exige um reajuste manual da interface toda vez que se deseja adaptar o *moon pool* a um novo casco, ou mesmo quando as dimensões do *moon pool*, em um dado casco, crescem o suficiente para que os seus painéis de contorno mudem.

Idealmente, o novo método permitiria a criação automática de um *moon pool* de qualquer dimensão em um casco qualquer, sem definição manual de pontos ou painéis. Oferecer ao usuário um maior controle sobre o refino da malha do *moon pool* também seria desejável. No entanto, não se sabe ainda o quão complexo seria desenvolver tal método, de forma que pode ser necessário realizar simplificações em relação a esse modelo ideal durante o desenvolvimento.

No aspecto da metodologia de otimização, uma série de aperfeiçoamentos já estão em curso neste estágio do trabalho. O cálculo preciso da nota de otimização com base na área do gráfico radar é um deles, por exemplo.

Os critérios de operabilidade poderão ainda passar por um incremento significativo antes de chegar ao seu formato final. A princípio, deve-se buscar normas que orientem sobre a prioridade dada a critérios de operabilidade específicos, para o caso de empate entre as notas de dois indivíduos distintos. De preferência, alguma que discorra especificamente sobre as condições no *moon pool*. Os critérios aqui utilizados, adotados conforme sugerido por [Cavalcante \(2015\)](#), não encontram suporte adicional na literatura.

A inclusão do critério de resistência estrutural por momento fletor deverá ser efetivada assim que o código necessário estiver disponível. Também seria pertinente a adição de um critério de conforto da tripulação, possivelmente de acordo com a norma [ISO \(1997\) 2631](#), que trata sobre enjoo.

# Referências

AALBERS, A. The water motions in a moonpool. *Ocean Engineering*, Elsevier BV, v. 11, n. 6, p. 557–579, jan 1984. Disponível em: <[http://dx.doi.org/10.1016/0029-8018\(84\)90001-5](http://dx.doi.org/10.1016/0029-8018(84)90001-5)>. 21, 23, 53

BAGHERI, H.; GHASSEMI, H. Genetic algorithm applied to optimization of the ship hull form with respect to seakeeping performance. *Transactions of FAMENA*, Fakultet strojarstva i brodogradnje, v. 38, n. 3, p. 45–58, 2014. Disponível em: <<http://hrcak.srce.hr/129612>>. 29

BULL, D. An improved understanding of the natural resonances of moonpools contained within floating rigid-bodies: Theory and application to oscillating water column devices. *Ocean Engineering*, Elsevier BV, v. 108, p. 799–812, nov 2015. Disponível em: <<http://dx.doi.org/10.1016/j.oceaneng.2015.07.007>>. 24

CAVALCANTE, T. de M. *FERRAMENTAS COMPUTACIONAIS PARA GERAÇÃO E SELEÇÃO DE GEOMETRIAS DE MOONPOOL DE DRILLSHIP COM BASE NA OPERABILIDADE*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2015. 17, 28, 29, 32, 33, 44, 45, 46, 47, 60

COVERT, E. E. An approximate calculation of the onset velocity of cavity oscillations. *AIAA Journal*, American Institute of Aeronautics and Astronautics (AIAA), v. 8, n. 12, p. 2189–2194, dec 1970. Disponível em: <<http://dx.doi.org/10.2514/3.6085>>. 23

DAY, A. The design of moonpools for subsea operations. *Marine Technology*, v. 27, n. 3, p. 167–179, 1990. 16

FALTINSEN, O. *Sea Loads on Ships and Offshore Structures*. [S.l.]: Cambridge University Press, 1993. ISBN 0521458706. 23, 45

FREDRIKSEN, A. G.; KRISTIANSEN, T.; FALTINSEN, O. M. Experimental and numerical investigation of wave resonance in moonpools at low forward speed. *Applied Ocean Research*, Elsevier BV, v. 47, p. 28–46, aug 2014. Disponível em: <<http://dx.doi.org/10.1016/j.apor.2014.03.005>>. 24

FUKUDA, K. Behavior of water in vertical well with bottom opening of ship, and its effects on ship-motion. *J. SNAJ, Nihon zousen gakkai ronbunshu*, Japan Society of Naval Architects and Ocean Engineers, v. 1977, n. 141, p. 107–122, 1977. Disponível em: <<http://dx.doi.org/10.2534/jjasnaoe1968.1977.107>>. 20, 22, 23

GAILLARDE, G.; COTTELEER, A. Water motion in moonpools empirical and theoretical approach. *Association Technique Maritime et Aeronautique (ATMA), Paris, France*, 2004. Disponível em: <[http://www.marin.nl/upload\\_mm/2/8/8/1806831027\\_1999999096\\_Water\\_motion\\_in\\_moonpools.pdf](http://www.marin.nl/upload_mm/2/8/8/1806831027_1999999096_Water_motion_in_moonpools.pdf)>. 22, 23

ISO. *Mechanical Vibration and Shock: Evaluation of Human Exposure to Whole-body Vibration. Part 1, General Requirements: International Standard ISO 2631-1: 1997 (E)*. [S.l.]: ISO, 1997. 60

KOSHIZUKA, S.; OKA, Y. Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nuclear science and engineering*, American Nuclear Society, v. 123, n. 3, p. 421–434, 1996. Disponível em: <[http://www.ans.org/pubs/journals/nse/a\\_24205](http://www.ans.org/pubs/journals/nse/a_24205)>. 24

MALTA, E. B.; CUEVA, M.; NISHIMOTO, K.; GONÇALVES, R.; MASETTI, I. Numerical moonpool modeling. In: *Volume 1: Offshore Technology, Offshore Wind Energy, Ocean Research Technology, LNG Specialty Symposium*. ASME International, 2006. Disponível em: <<http://dx.doi.org/10.1115/OMAE2006-92456>>. 24

MICHIMA, P. S. A.; KAWABE, H. Preliminary chart of drill ship operability considering moon-pool sloshing effects. In: INTERNATIONAL SOCIETY OF OFFSHORE AND POLAR ENGINEERS. *The Twenty-fourth International Ocean and Polar Engineering Conference*. 2014. Disponível em: <<https://www.onepetro.org/conference-paper/ISOPE-I-14-366>>. 24, 27, 28, 30, 31, 36, 44

MITCHELL, M. *An introduction to genetic algorithms*. [S.l.]: MIT press, 1998. 28

MOLIN, B. On the piston and sloshing modes in moonpools. *J. Fluid Mech.*, Cambridge University Press (CUP), v. 430, p. 27–50, mar 2001. Disponível em: <<http://dx.doi.org/10.1017/S0022112000002871>>. 23

MONAGHAN, J. J. Smoothed particle hydrodynamics. *Annual review of astronomy and astrophysics*, v. 30, p. 543–574, 1992. Disponível em: <<http://adsabs.harvard.edu/full/1992ARA%26A..30..543M>>. 24

NAUDASCHER, E. From flow instability to flow-induced excitation. *Journal of the Hydraulics Division*, ASCE, v. 93, n. 4, p. 15–40, 1967. 20

PIERSON, W. J.; MOSKOWITZ, L. A proposed spectral form for fully developed wind seas based on the similarity theory of sa kitaigorodskii. *Journal of geophysical research*, Wiley Online Library, v. 69, n. 24, p. 5181–5190, 1964. 44

ROCKWELL, D.; NAUDASCHER, E. Self-sustained oscillations of impinging free shear layers. *Annu. Rev. Fluid Mech.*, Annual Reviews, v. 11, n. 1, p. 67–94, jan 1979. Disponível em: <<http://dx.doi.org/10.1146/annurev.fl.11.010179.000435>>. 20

TÖRNBLOM, J.; HAMMARGREN, E. *Effect of the Moonpool on the Total Resistance of a Drillship*. Dissertação (Mestrado), 2012. Disponível em: <<http://publications.lib.chalmers.se/records/fulltext/162731.pdf>>. 21, 22

# APÊNDICE A – Códigos alterados para a compatibilização

Autor: Túlio de Moura Cavalcante.

Alterações: Ícaro Aragão Fonseca.

## 1. GAMMA.m

```
clear all
clc

% Read main input parameters.-----%
fileID = fopen('Optimization_Parameters.dat','r');

line_data = textscan(fileID, '%*[^']' %d',1);
ni = line_data{1}; % Total number of individuals to each evaluation.

line_data = textscan(fileID, '%f', 'HeaderLines',2);
L1 = transpose(line_data{1}); % Based on the tipical moonpool lenght and
                             % width, set values for L1.

line_data = textscan(fileID, '%f', 'HeaderLines',1);
L2 = transpose(line_data{1}); % Based on the tipical moonpool lenght and
                             % width, set values for L2.

line_data = textscan(fileID, '%*[^']' %f',1);
p_no = line_data{1}; % Percentage of the individuals to be selected as the
                    % best.

fclose(fileID);
% Finish reading main input parameters.-----%

y=1                                ; % Counter;

% Create main folder.-----%
mkdir('Files');

for w1=1:size(L1,2),
    for w2=1:size(L2,2),

        l1=L1(w1); % Varying the l1;
        l2=L2(w2); % Varying the l2;

        % Create main folder for a given L1 x L2 combination.-----%
        folder_name_par = ['Files\','Parameter combination ',num2str(l1),...
                            ' x ',num2str(l2)];
        mkdir(folder_name_par);

        % Parameter generation acording the chosen l1 and l2 -----%
        [ m1, m2, m3, m4, nf, nt, p ] = Geracao_de_Parametros (l1, l2);
        % Parameters which will be combined to construct individuals.-----%
```



---

```

    %-- Breaking criterion -----%
    F(t)=0.5*(nota4(1,1)+nota4(1,2))          ; % Calculate the breaking
                                                % parameter, average between
                                                % the bests.

    if t>2
        [ q ] = Avaliacao( F, q, t )          ; % With the breaking parameter,
                                                % evaluate if it is the moment
                                                % to break.
    end
    t=t+1                                     ; % Increase 't'.

end

% Keeping important parameters -----%
m1=I4(1,1); %
m2=I4(2,1); %
m3=I4(3,1); % ASSIGN PARAMETERS.
m4=I4(4,1); %
nf=I4(5,1); %
nt=I4(6,1); %

KI(:,y)=I4(:,1)      ;
Kn(:,y)=nota4(:,1)   ;
KR(y,:)=radar4(1,:);
Kl(y,:)= [l1 l2]     ;
Kd1(:,y)=des41(:,1);
Kd2(:,y)=des42(:,1);
y=y+1                ;
%-----%

save results.mat

end
end

%-- Select the best of the bests -----%
[ I10, nota10, des10, des20, GR10 ] = Selecao( Kn, Kd1, Kd2, KI, 1, KR );

% Then search for and store its important parameters -----%
for i=1:size(Kn,2),
    if (nota10==Kn(i)) && (des10==Kd1(i)) && (des20==Kd2(i))
        l10=Kl(i,1);
        l20=Kl(i,2);
    end
end

save results.mat
%-----%

```

---

## 2. Acrescenta\_Pontos.m

```

function [ v, num_vertices, ov ] = Acrescenta_Pontos( v, p, num_vertices, ov )
% Function which add the moonpool points to ship points.

K=size(v,1)          ; % 'v' is the matrix os the ship points, without those

```

```

for i=1:size(p,1), % which were in the moonpool region.
    v(K+i,:)=p(i,:); % Add moonpool points (p) to 'v'.
end %

num_vertices=size(v,1); % Adjust the new number of vertices.

for i=1:size(v,1), %
    ov(i)=i % Adjust the order number.
end %

end

```

---

### 3. Build\_Moonpools.m

```

function [ nota, desemp1, desemp2, radar ] = Build_Moonpools( I, l1, l2, folder_str, nstart )
% Build and evaluate moonpools -----%

nota=zeros(1,size(I,2));
desemp1=zeros(1,size(I,2));
desemp2=zeros(1,size(I,2));
radar=zeros(size(I,2),7);

for i=nstart:size(I,2),
    m1=I(1,i); %
    m2=I(2,i); %
    m3=I(3,i); % ASSIGN PARAMETERS.
    m4=I(4,i); %
    nf=I(5,i); %
    nt=I(6,i); %

    folder_name = [folder_str,num2str(i)];
    mkdir(folder_name);

    Gerador_Moonpool(l1, l2, m1, m2, m3, m4, nf, nt);
    status=1;

    %Run analysis until it is executed successfully -----%
    while status~=0
        command = '3Dmotion.Moonpool.REGULAR-IRREGULAR-beta6.exe';
        status = system(command)
        if status~=0
            delete('Series 60*');
        end
    end

    % Calculate the Operability of the ship-moonpool -----%
    [ A, D1, D2, GR ] = Eval_Data(folder_name);

    copyfile('Panel.Drill-Ship-model.DAT',folder_name);
    movefile('malha.msh',folder_name);
    movefile('Panel.Drill-Ship-model.DAT',folder_name);
    movefile('Series 60*',folder_name);

    % Writes the parameters in the file moonpool.txt
    fid = fopen(fullfile(folder_name,'moonpool.txt'),'wt');
    fprintf(fid,'Moon pool parameters\n');

```

```

fprintf(fid, ['l1 ', num2str(l1), '\n']);
fprintf(fid, ['l2 ', num2str(l2), '\n']);
fprintf(fid, ['m1', ' ', num2str(m1), '\n']);
fprintf(fid, ['m2', ' ', num2str(m2), '\n']);
fprintf(fid, ['m3', ' ', num2str(m3), '\n']);
fprintf(fid, ['m4', ' ', num2str(m4), '\n']);
fprintf(fid, ['nf', ' ', num2str(nf), '\n']);
fprintf(fid, ['nt', ' ', num2str(nt), '\n']);
fclose(fid);

% Keep the grades in 'nota' -----%
nota(i)=A;
desemp1(i)=D1;
desemp2(i)=D2;
radar(i,:)=GR;
%-----%
end

end

```

---

#### 4. Eval\_Data.m

```

function [ A, D1, D2, GR ] = Eval_Data (folder_name)
% A: radar chart area ; D1, D2: tiebreak grades; GR: resultant polygon points

% Read input parameters.-----%
fileID = fopen('Optimization_Parameters.dat','r');
line_data = textscan(fileID, '%*[^']' '%d', 6, 'HeaderLines', 18);

sea_opt = line_data{1}; % Sea states to perform analysis.

fclose(fileID);
% Finish reading input parameters.-----%

sea_states = {'IACS LONG CRESTED'; 'IACS SHORT CRESTED';
              'JONSWAP LONG CRESTED'; 'JONSWAP SHORT CRESTED';
              'Bretschneider LONG CRESTED'; 'Bretschneider SHORT CRESTED'};

% Read input parameters.-----%
fileID = fopen('Optimization_Parameters.dat','r');
line_data = textscan(fileID, '%f', 'HeaderLines', 25);

T_opt = transpose(line_data{1}); % Choose significant wave periods to
                                % perform analysis.

fclose(fileID);
% Finish reading input parameters.-----%

% from now on, finding data of interest location.
C = textread('Series 60 -MOTION-SPECTRUM.DAT', '%s', 'delimiter', '\n');

line_mark1 = [];

for i = 1:size(C,1);
    if strcmp(C{i}, '***** STANDARD DEVIATION OF RESPONSE *****')
        line_mark1(end+1) = i;
    end
end

```

```

    end
end

D = textread('Series 60 -MOON-POOL-SPECTRUM.DAT', '%s','delimiter', '\n');

line_mark2 = [];

for i = 1:size(D,1);
    if strcmp(D{i}, '***** STANDARD DEVIATION OF RESPONSE *****')
        line_mark2(end+1) = i;
    end
end

E = textread('Series 60 -DRIFT-FORCE-SPECTRUM.DAT', '%s','delimiter', '\n');

line_mark3 = [];

for i = 1:size(E,1);
    if strcmp(E{i}, '***** DRIFT FORCE RESPONSE *****')
        line_mark3(end+1) = i;
    end
end

exec_id = 0; % Optimization execution identifier.

for i=1:size(sea_opt,1)
    if sea_opt(i)
        j = 1+13*(i-1); % position of the matrix line_mark where the first
                        % occurrence of this spectrum's output data starts
        for m=1:size(T_opt,2)
            exec_id = exec_id+1;
            T = T_opt(m);

            % allocating treshold height matrices to save computing time.
            H_str = zeros(1,7);
            H_mid = zeros(1,7);
            H_for = zeros(1,7);
            H_dri = zeros(1,7);
            H_opt = zeros(1,7);

            for k = 0:6 % k is attack angle counter.
                str_motion = C{line_mark1{j+k}+T+1}; % placing the marker
                str_moon = D{line_mark2{j+k}+T+1}; % at the data line.
                str_drift = E{line_mark3{j+k}+T+1};
                % concluded finding data of interest location.

                % acquiring values from marked lines.
                resp_motion = textscan(str_motion,'%f');
                resp_moon = textscan(str_moon,'%f');
                resp_drift = textscan(str_drift,'%f');

                % Stroke compensator criterion.
                H_str(:,k+1) = Tres_Height(2, resp_motion{1}(4), T);

                % Overflow in the middle of the moon-pool criterion.
                H_mid(:,k+1) = Tres_Height(1, resp_moon{1}(2), T);

                % Overflow in the fore of the moon-pool criterion.
                H_for(:,k+1) = Tres_Height(1, resp_moon{1}(3), T);
            end
        end
    end
end

```

```

        % Drift force criterion.
        H_dri(:,k+1) = Tres_Height(4, (resp_drift{1}(4)^2 +...
            resp_drift{1}(5)^2)^0.5, T);
    end

    for l = 1:size(H_opt,2)
        H_opt(:,l) = min([H_str(:,l),H_mid(:,l),H_for(:,l),...
            H_dri(:,l)]);
    end

    % Calculate Radar Graph Area (neglecting intersections
    % between standard attack angles). -----%
    alpha=pi/6; % number of radar graph sectors.
    SA=0; % polygon area
    for n=1:6,
        v1=[0 H_opt(:,n) 0];
        v2=[H_opt(:,n+1)*cos(alpha) H_opt(:,n+1)*sin(alpha) 0];
        c=cross(v1,v2);
        sa=norm(c); % double of sector area.
        SA=SA+sa;
    end
    %-----%

    % Verify if the criteria for this combination of sea state
    % and significant wave period is critical and store it if
    % it is.-----%
    if exec_id == 1
        A=SA;
        H_crit=H_opt;
        crit_sea_state=sea_states(i);
        T_crit=T;
        H_str_min=H_str;
        H_mid_min=H_mid;
        H_for_min=H_for;
        H_dri_min=H_dri;
    elseif exec_id>1 && SA<A
        A=SA;
        H_crit=H_opt;
        crit_sea_state=sea_states(i);
        T_crit=T;
        H_str_min=H_str;
        H_mid_min=H_mid;
        H_for_min=H_for;
        H_dri_min=H_dri;
    end
    %-----%
end
end
end

% write in 'radar.txt' file the set of points of each criterion's radar
% chart values-----%
fid = fopen(fullfile(folder_name, '\radar.txt'), 'wt');
fprintf(fid, 'Critical sea state: %s\n', crit_sea_state{1});
fprintf(fid, 'Critical significant period: %d\n', T_crit);
fprintf(fid, ['stroke', ' ', num2str(H_str_min), '\n']);
fprintf(fid, ['mid', ' ', num2str(H_mid_min), '\n']);
fprintf(fid, ['for', ' ', num2str(H_for_min), '\n']);
fprintf(fid, ['drift', ' ', num2str(H_dri_min), '\n']);
fprintf(fid, ['op_score', ' ', num2str(A), '\n']);

```

```

fclose(fid);

% Tie break parameters ----- %
D1=H_crit(1,4);
D2=H_crit(1,7);
% ----- %

% Polygon Vertices Vector ----- %
GR=H_crit;
% ----- %
end

```

## 5. Geom\_Elementos.m

```

function [ tri, quad, num_elem_t, num_elem, num_elem_q, e, num_elem_moon, ...
    num_elem_free ] = Geom_Elementos( tri, p, quad, v, nf, nt, num_elem_t, ...
    num_elem, num_elem_q, e, l1, l2, num_vertices_moon, div_z )
% Function which assemble moonpool elements ----- %

if ((l1.16<=l1)&&(l1<=13)&&(5.594666667<=l2)&&(l2<=7))

    p1(1)=63; %
    p1(2)=62; %
    p1(3)=61; %
    p1(4)=68; % Handmade assemble of points which were already part of the hull
    p1(5)=73; % and will be part of moonpool. This is a limitation of this
    p1(6)=78; % program, we did not find a clever way to find, for a generic
    p1(7)=83; % ship, the points of the hull which will also be part of the
    p1(8)=84; % moonpool.
    p1(9)=85; %

    for i=1:size(v,1),
        for j=1:size(p,1),
            if v(i,:)==p(j,:)
                p1(j+9)=i; % Assemble of the moonpool points which will also
            end          % be part of the hull, but were not part of it yet.
        end
    end

    U=l1;
    H=size(e,1);
    T=H;

    % ----- %
    e(H+1,:)= [H+1 p1(2) p1(1) p1(12) p1(12)] ; %
    e(H+2,:)= [H+2 p1(12) p1(13) p1(2) p1(2)] ; %
    e(H+3,:)= [H+3 p1(2) p1(13) p1(U-1) p1(U-1)] ; %
    e(H+4,:)= [H+4 p1(3) p1(2) p1(U-1) p1(U-1)] ; %
    e(H+5,:)= [H+5 p1(3) p1(U-1) p1(4) p1(4)] ; %
    e(H+6,:)= [H+6 p1(4) p1(U-1) p1(13+nf) p1(13+nf)] ; % Assigning
    e(H+7,:)= [H+7 p1(4) p1(13+nf) p1(5) p1(5)] ; % of
    e(H+8,:)= [H+8 p1(5) p1(13+nf) p1(13+nf+1) p1(13+nf+1)] ; % moonpool
    e(H+9,:)= [H+9 p1(5) p1(13+nf+1) p1(13+nf+2) p1(13+nf+2)] ; % interface
    e(H+10,:)= [H+10 p1(6) p1(13+nf+2) p1(13+nf+3) p1(13+nf+3)] ; % elements.
    e(H+11,:)= [H+11 p1(5) p1(13+nf+2) p1(6) p1(6)] ; %
    e(H+12,:)= [H+12 p1(6) p1(13+nf+3) p1(U) p1(U)] ; %

```

```

e(H+13,:)= [H+13 p1(6) p1(U) p1(7) p1(7)] ; %
e(H+14,:)= [H+14 p1(7) p1(U) p1(8) p1(8)] ; %
e(H+15,:)= [H+15 p1(U) p1(13+nf+nt+2) p1(8) p1(8)] ; %
e(H+16,:)= [H+16 p1(13+nf+nt+2) p1(13+nf+nt+3) p1(8) p1(8)] ; %
e(H+17,:)= [H+17 p1(13+nf+nt+3) p1(9) p1(8) p1(8)] ; %
H=size(e,1) ; %
for i=1:nf, %
    e(H+i,:)= [H+i p1(U-1) p1(13+i-1) p1(13+i) p1(13+i)] ; %
end %
for i=1:nt-1, %
    e(H+nf+i,:)= [H+nf+i p1(13+nf+i+2) p1(13+nf+i+3) p1(U) p1(U)]; %
end
%-----%

elseif ((10<=l1)&&(l1<11.16)&&(5.594666667<=l2)&&(l2<=7))

p1(1)=63; %
p1(2)=62; %
p1(3)=61; %
p1(4)=68; % Handmade assemble of points which were already part of the hull
p1(5)=73; % and will be part of moonpool. This is a limitation of this
p1(6)=78; % program, we did not find a clever way to find, for a generic
p1(7)=78; % ship, the points of the hull which will also be part of the
p1(8)=79; % moonpool.
p1(9)=80; %

for i=1:size(v,1),
    for j=1:size(p,1),
        if v(i,:)==p(j,:)
            p1(j+9)=i; % Assemble of the moonpool points which will also
            % be part of the hull, but were not part of it yet.
        end
    end
end

U=11;
H=size(e,1);
T=H;

%-----%
e(H+1,:)= [H+1 p1(2) p1(1) p1(12) p1(12)] ; %
e(H+2,:)= [H+2 p1(12) p1(13) p1(2) p1(2)] ; %
e(H+3,:)= [H+3 p1(2) p1(13) p1(U-1) p1(U-1)] ; %
e(H+4,:)= [H+4 p1(3) p1(2) p1(U-1) p1(U-1)] ; %
e(H+5,:)= [H+5 p1(3) p1(U-1) p1(4) p1(4)] ; %
e(H+6,:)= [H+6 p1(4) p1(U-1) p1(13+nf) p1(13+nf)] ; % Assigning
e(H+7,:)= [H+7 p1(4) p1(13+nf) p1(5) p1(5)] ; % of
e(H+8,:)= [H+8 p1(5) p1(13+nf) p1(13+nf+1) p1(13+nf+1)] ; % moonpool
e(H+9,:)= [H+9 p1(5) p1(13+nf+1) p1(13+nf+2) p1(13+nf+2)] ; % interface
e(H+10,:)= [H+10 p1(U) p1(13+nf+2) p1(13+nf+3) p1(13+nf+3)] ; % elements.
e(H+11,:)= [H+11 p1(5) p1(13+nf+2) p1(6) p1(6)] ; %
e(H+12,:)= [H+12 p1(13+nf+2) p1(13+nf+3) p1(U) p1(U)] ; %
e(H+13,:)= [H+13 p1(7) p1(U) p1(8) p1(8)] ; %
e(H+14,:)= [H+14 p1(U) p1(13+nf+nt+2) p1(8) p1(8)] ; %
e(H+15,:)= [H+15 p1(13+nf+nt+2) p1(13+nf+nt+3) p1(8) p1(8)] ; %
e(H+16,:)= [H+16 p1(13+nf+nt+3) p1(9) p1(8) p1(8)] ; %
H=size(e,1) ; %
for i=1:nf, %
    e(H+i,:)= [H+i p1(U-1) p1(13+i-1) p1(13+i) p1(13+i)] ; %
end %
for i=1:nt-1, %

```

```

        e(H+nf+i,:)= [H+nf+i p1(13+nf+i+2) p1(13+nf+i+3) p1(U) p1(U)]; %
    end
    %-----%

elseif ((11.16<=l1)&&(l1<=13)&&(5<=l2)&&(l2<5.594666667))

    p1(1)=63; %
    p1(2)=62; %
    p1(3)=62; %
    p1(4)=69; % Handmade assemble of points which were already part of the hull
    p1(5)=75; % and will be part of moonpool. This is a limitation of this
    p1(6)=81; % program, we did not find a clever way to find, for a generic
    p1(7)=87; % ship, the points of the hull which will also be part of the
    p1(8)=87; % moonpool.
    p1(9)=88; %

    for i=1:size(v,1),
        for j=1:size(p,1),
            if v(i,:)==p(j,:)
                p1(j+9)=i; % Assemble of the moonpool points which will also
            end % be part of the hull, but were not part of it yet.
        end
    end

    end

    U=11;
    H=size(e,1);
    T=H;

    %-----%
    e(H+1,:)= [H+1 p1(2) p1(1) p1(12) p1(12)] ; %
    e(H+2,:)= [H+2 p1(12) p1(13) p1(2) p1(2)] ; %
    e(H+3,:)= [H+3 p1(2) p1(13) p1(U-1) p1(U-1)] ; %
    e(H+4,:)= [H+4 p1(3) p1(U-1) p1(4) p1(4)] ; %
    e(H+5,:)= [H+5 p1(4) p1(U-1) p1(13+nf) p1(13+nf)] ; % Assigning
    e(H+6,:)= [H+6 p1(4) p1(13+nf) p1(5) p1(5)] ; % of
    e(H+7,:)= [H+7 p1(5) p1(13+nf) p1(13+nf+1) p1(13+nf+1)] ; % moonpool
    e(H+8,:)= [H+8 p1(5) p1(13+nf+1) p1(13+nf+2) p1(13+nf+2)] ; % interface
    e(H+9,:)= [H+9 p1(6) p1(13+nf+2) p1(13+nf+3) p1(13+nf+3)] ; % elements.
    e(H+10,:)= [H+10 p1(5) p1(13+nf+2) p1(6) p1(6)] ; %
    e(H+11,:)= [H+11 p1(6) p1(13+nf+3) p1(U) p1(U)] ; %
    e(H+12,:)= [H+12 p1(6) p1(U) p1(7) p1(7)] ; %
    e(H+13,:)= [H+13 p1(U) p1(13+nf+nt+2) p1(8) p1(8)] ; %
    e(H+14,:)= [H+14 p1(13+nf+nt+2) p1(13+nf+nt+3) p1(8) p1(8)] ; %
    e(H+15,:)= [H+15 p1(13+nf+nt+3) p1(9) p1(8) p1(8)] ; %
    H=size(e,1) ; %
    for i=1:nf, %
        e(H+i,:)= [H+i p1(U-1) p1(13+i-1) p1(13+i) p1(13+i)] ; %
    end %
    for i=1:nt-1, %
        e(H+nf+i,:)= [H+nf+i p1(13+nf+i+2) p1(13+nf+i+3) p1(U) p1(U)]; %
    end
    %-----%

elseif ((10<=l1)&&(l1<11.16)&&(5<=l2)&&(l2<5.594666667))

    p1(1)=63; %
    p1(2)=62; %
    p1(3)=62; %
    p1(4)=69; % Handmade assemble of points which were already part of the hull
    p1(5)=75; % and will be part of moonpool. This is a limitation of this

```

```

p1(6)=81; % program, we did not find a clever way to find, for a generic
p1(7)=81; % ship, the points of the hull which will also be part of the
p1(8)=81; % moonpool.
p1(9)=82; %

for i=1:size(v,1),
    for j=1:size(p,1),
        if v(i,:)==p(j,:)
            p1(j+9)=i; % Assemble of the moonpool points which will also
        end          % be part of the hull, but were not part of it yet.
    end
end

U=11;
H=size(e,1);
T=H;

%-----%
e(H+1,:)=[H+1 p1(2) p1(1) p1(12) p1(12)] ; %
e(H+2,:)=[H+2 p1(12) p1(13) p1(2) p1(2)] ; %
e(H+3,:)=[H+3 p1(2) p1(13) p1(U-1) p1(U-1)] ; %
e(H+4,:)=[H+4 p1(3) p1(U-1) p1(4) p1(4)] ; %
e(H+5,:)=[H+5 p1(4) p1(U-1) p1(13+nf) p1(13+nf)] ; % Assigning
e(H+6,:)=[H+6 p1(4) p1(13+nf) p1(5) p1(5)] ; % of
e(H+7,:)=[H+7 p1(5) p1(13+nf) p1(13+nf+1) p1(13+nf+1)] ; % moonpool
e(H+8,:)=[H+8 p1(5) p1(13+nf+1) p1(13+nf+2) p1(13+nf+2)] ; % interface
e(H+9,:)=[H+9 p1(U) p1(13+nf+2) p1(13+nf+3) p1(13+nf+3)] ; % elements.
e(H+10,:)=[H+10 p1(5) p1(13+nf+2) p1(6) p1(6)] ; %
e(H+11,:)=[H+11 p1(6) p1(13+nf+2) p1(U) p1(U)] ; %
e(H+12,:)=[H+12 p1(U) p1(13+nf+nt+2) p1(8) p1(8)] ; %
e(H+13,:)=[H+13 p1(13+nf+nt+2) p1(13+nf+nt+3) p1(8) p1(8)] ; %
e(H+14,:)=[H+14 p1(13+nf+nt+3) p1(9) p1(8) p1(8)] ; %
H=size(e,1) ; %
for i=1:nf, %
    e(H+i,:)=[H+i p1(U-1) p1(13+i-1) p1(13+i) p1(13+i)] ; %
end %
for i=1:nt-1, %
    e(H+nf+i,:)=[H+nf+i p1(13+nf+i+2) p1(13+nf+i+3) p1(U) p1(U)] ; %
end %
%-----%

end

%-- Assemble the elements also in gmsh format to see the moonpool by gmsh --%
R=size(tri,1);

for i=1:size(e,1)-T,
    tri(R+i,1)=R+i ; %
    tri(R+i,2)=2 ; %
    tri(R+i,3)=2 ; %
    tri(R+i,4)=0 ; % Assign the triangles in gmsh format.
    tri(R+i,5)=6 ; %
    tri(R+i,6)=e(T+i,2) ; %
    tri(R+i,7)=e(T+i,3) ; %
    tri(R+i,8)=e(T+i,4) ; %
end

R1=size(tri,1); % Correct the amount of triangles.
num_elem_t=R1 ; %

```

---

```

for i=1:size(quad,1),
    quad(i,1)=R1+i; % Correct the quadrangle orders.
end

M=num_vertices_moon/(div_z+1); % Adjustment number to combine points.

for i=1:div_z,
    for j=1:M-1,
        eq((i-1)*(M-1)+j,:)= [p1(12+M+j+M*(i-1)) ... % Combine points
                               p1(12+j+M*(i-1)) p1(12+j-1+M*(i-1)) ... % to make moon
                               p1(12+M+j-1+M*(i-1))]; % pool elements.
    end
end

num_elem_moon=(M-1)*div_z;
K=(M-1)*div_z+1;

% Combine points to make free surface elements -----%
eq(K,:)= [p1(12+M*div_z+1) p1(12+M*div_z) p1(12+M*(div_z+1)) ...
          p1(12+M*div_z+2)];

for i=1:M-5,
    eq(K+i,:)= [p1(12+M*div_z+i+1) p1(12+M*(div_z+1)+i-1) ...
                p1(12+M*(div_z+1)+i) p1(12+M*div_z+i+2)];
end

eq(K+M-4,:)= [p1(12+M*(div_z+1)-3) p1(12+M*(div_z+1)+M-5) ...
               p1(12+M*(div_z+1)-1) p1(12+M*(div_z+1)-2)];

num_elem_free=M-3;

F=size(quad,1);
Q=F+R1;

for i=1:size(eq,1),
    quad(F+i,1)=Q+i ; %
    quad(F+i,2)=3 ; %
    quad(F+i,3)=2 ; %
    quad(F+i,4)=0 ; %
    quad(F+i,5)=6 ; % Assign the quadrangles in gmsh format.
    quad(F+i,6)=eq(i,1); %
    quad(F+i,7)=eq(i,2); %
    quad(F+i,8)=eq(i,3); %
    quad(F+i,9)=eq(i,4); %
end

F1=size(quad,1) ; %
num_elem_q=F1 ; % Adjust the related variables.
num_elem=num_elem-R+R1-F+F1; %
%-----%

end

```

---

## 6. Geom\_Pontos.m

```

function [ p, num_vertices_moon, num_vertices_free ] = Geom_Pontos( m1, ...

```

```

    m2, m3, m4, l1, l2, nf, nt, z, div_z )
% Function which makes the moonpool points data matrix.

if m2==0
    m2=0.0001 ; % To avoid singularity like some ellipse radius being
elseif m2==l1 % zero.
    m2=l1-0.0001; %
end
if m3==0
    m3=0.0001 ; % Same as above.
elseif m3==l1 %
    m3=l1-0.0001; %
end
if m1==0
    m1=0.0001 ; % Same as above.
elseif m1==l2 %
    m1=l2-0.0001; %
end
if m4==0
    m4=0.0001 ; % Same as above.
elseif m4==l2 %
    m4=l2-0.0001; %
end

dx=(l1-m2)/nf ; %
x1(1)=l1 ; % Determine the 'x' coordinates of the first quarter
for i=1:nf, % of ellipse.
    x1(i+1)=l1-dx*i; %
end

dx=(l1-m3)/nt ; %
x2(1)=-l1 ; % Determine the 'x' coordinates of the second quarter
for i=1:nt, % of ellipse.
    x2(i+1)=-l1+dx*i; %
end

a=l1-m2; % First radius of the first ellipse.
b=l2-m1; % Second radius of the first ellipse.
Y1=m1 ; % 'y' coordinate of first ellipse center.
X1=m2 ; % 'x' coordinate of first ellipse center.

for i=1:size(x1,2), % Determine the 'y' coordinates of
    y1(i)=b*sqrt(1-((x1(i)-X1)/a)^2)+Y1; % the first quarter of ellipse.
end

a=l1-m3; % First radius of the second ellipse.
b=l2-m4; % Second radius of the second ellipse.
Y2=m4 ; % 'y' coordinate of second ellipse center.
X2=-m3 ; % 'x' coordinate of second ellipse center.

t=1 ; %
for j=size(x2,2):-1:1, %
    f(t)=x2(j) ; % Reorder x2.
    t=t+1 ; %
end %
x2=f ; %

for i=1:size(x2,2), % Determine the 'y' coordinates of
    y2(i)=b*sqrt(1-((x2(i)-X2)/a)^2)+Y2; % the second quarter of ellipse.
end

```

---

```

x(1)=l1; % Set the first point of the moonpool to be (l1,0).
y(1)=0 ; %

for i=1:size(y1,2), %
    y(i+1)=y1(i); % Then assemble the first ellipse coordinates to
    x(i+1)=x1(i); % moonpool.
end

x(i+2)=0 ; % Assemble the coordinates of the middle moonpool point.
y(i+2)=l2 ; %
R=size(y,2); %

for i=1:size(y2,2), %
    y(i+R)=y2(i); % Then assemble the second ellipse coordinates to
    x(i+R)=x2(i); % moonpool.
end

R=size(y,2); % Set the last point of the moonpool to be (-l1,0)
x(R+1)=-l1 ; %
y(R+1)=0 ; %

% Writing vertical coordinates for each row of points, being 0 the top of
% the moon pool and z, which is a negative number, the base.
for i=1:(div_z+1);
    zc(:,i)=z/div_z*(div_z+1-i)*ones(1,size(y,2));
end;

p(1,:)=[ l1 l2 z]; % Build the moonpool points data matrix.
p(2,:)=[-l1 l2 z]; %

for i=1:(div_z+1),
    for j=1:size(zc,1), % Transferring the
        p((i-1)*size(zc,1)+j+2,:)=[x(j) y(j) zc(j,i)]; % three coordinates
    end % to the points
end % matrix.

for i=3:(size(zc,1)-2),
    p(i+size(zc,1)*size(zc,2),:)= [x(i) 0 zc(i,size(zc,2))]; % Free-surface points.
end

num_vertices_moon = size(zc,1)*size(zc,2);
num_vertices_free = 2*size(zc,1)-4;

end

```

---

## 7. Gerador\_Moonpool.m

```

function Gerador_Moonpool(l1, l2, m1, m2, m3, m4, nf, nt)

% Read input parameter.-----%
fileID = fopen('Optimization_Parameters.dat','r');
line_data = textscan(fileID,'%s[^\n]' %f',1,'HeaderLines',6);

div_z = line_data{1}; % Number of panel rows in the prismatic section.

```

```

fclose(fileID);
% Finish reading input parameter.-----%

% Read the ship file -----%
[ quad,tri,bound,pontos,v,ov,verifica_t,verifica_q,verifica_b, ...
  num_elem, num_vertices, num_elem_t, num_elem_q, ...
  num_elem_b, e ] = Pre_Leitor;
-----%

% 'v' is the points data matrix. 'z' is the lowest z coordinate and o is
% the nearest x coordinate from zero -----%
[ z, o ] = Z_O( v );
-----%

% Eliminate the points, lines and elements which where in moonpool area -----%
[ quad, v, num_elem_q, num_vertices, num_elem, tri ] = ...
  Elim_Linhas_Pontos( v, l1, l2, quad, num_elem_t, num_elem_q, ...
    num_vertices, num_elem, tri );
-----%

% Build the moonpool -----%
[ p, num_vertices_moon, num_vertices_free ] = Geom_Pontos( m1, m2, m3, ...
  m4, l1, l2, nf, nt, z, div_z );
[ v, num_vertices, ov ] = Acrescenta_Pontos( v, p, num_vertices, ov );
-----%

% Assemble moonpool elements -----%
[ tri, quad, num_elem_t, num_elem, num_elem_q, e, num_elem_moon, ...
  num_elem_free ] = Geom_Elementos( tri, p, quad, v, nf, nt, ...
  num_elem_t, num_elem, num_elem_q, e, l1, l2, num_vertices_moon, div_z );
-----%

% Generates a file called 'malha1.msh' for gmsh visualization and another
% one called 'malha1.dat' to be used as input to the hydrodynamic analysis
% algorithm.-----%
Pre_Escrevedor(quad,tri,bound,pontos,v,ov,verifica_t,verifica_q,verifica_b, ...
  num_elem, num_vertices, num_vertices_moon, num_vertices_free, num_elem_t, ...
  num_elem_q, num_elem_b, num_elem_moon, num_elem_free, l1, l2, div_z);

end

```

## 8. Pre\_Escrevedor.m

```

function Pre_Escrevedor(quad,tri,bound,pontos,v,ov,verifica_t,verifica_q, ...
  verifica_b, num_elem, num_vertices, num_vertices_moon, ...
  num_vertices_free, num_elem_t, num_elem_q, num_elem_b, ...
  num_elem_moon, num_elem_free, l1, l2, div_z)
%
fid = fopen('malha.msh','wt');
fprintf(fid,'%MeshFormat\n');
fprintf(fid,'2.2 0 8\n');
fprintf(fid,'%EndMeshFormat\n');
fprintf(fid,'%Nodes\n');
fprintf(fid,'%u\n',num_vertices);
for i=1:num_vertices,
  fprintf(fid,'%u %.9f %.9f %.9f\n',ov(i),v(i,1),v(i,2),v(i,3));

```

```

end
fprintf(fid, '$EndNodes\n');
fprintf(fid, '$Elements\n');
fprintf(fid, '%u\n', num_elem);
for i=1:size(pontos,1),
    fprintf(fid, '%u %u %u %u %u %u\n', pontos(i,1), pontos(i,2), pontos(i,3) ...
        , pontos(i,4), pontos(i,5), pontos(i,6));
end
if verifica_b==1
    for i=1:num_elem_b,
        fprintf(fid, '%u %u %u %u %u %u %u\n', bound(i,1), bound(i,2), bound(i,3) ...
            , bound(i,4), bound(i,5), bound(i,6), bound(i,7));
    end
end
if verifica_t==1
    for i=1:num_elem_t,
        fprintf(fid, '%u %u %u %u %u %u %u %u\n', tri(i,1), tri(i,2), tri(i,3) ...
            , tri(i,4), tri(i,5), tri(i,6), tri(i,7), tri(i,8));
    end
end
if verifica_q==1
    for i=1:num_elem_q,
        fprintf(fid, '%u %u %u %u %u %u %u %u %u\n', quad(i,1), quad(i,2), quad(i,3) ...
            , quad(i,4), quad(i,5), quad(i,6), quad(i,7), quad(i,8), quad(i,9));
    end
end
fprintf(fid, '$EndElements\n');
fclose(fid);

%
fid1 = fopen('Panel.Drill-Ship-model.DAT', 'w');
fprintf(fid1, ''Drill ship grid and Panel (half body)'\n');
fprintf(fid1, ''Mool Pool Data      '\n');
fprintf(fid1, ''Rayleigh Damping    ' 0.03\n');
fprintf(fid1, '%s %.3f\n', ''Fore xPF m      =' ' , l1);
fprintf(fid1, '%s %.3f\n', ''Aft xPA m      =' ' , -l1);
fprintf(fid1, '%s %.3f\n', ''Width (half)m    =' ' , l2);
fprintf(fid1, ''Height m      =' ' 11.613\n');
fprintf(fid1, '%s %u\n', ''x-division      ' ' , num_vertices_moon/(div_z+1)-3);
fprintf(fid1, '%s %u\n', ''y-division      ' ' , 1);
fprintf(fid1, ''***** MAIN HULL PART GRID *****'\n');
fprintf(fid1, '%s %u\n', ''No. of Grid      ' ' , num_vertices ...
    -num_vertices_moon*(div_z-1)/(div_z+1)-num_vertices_free);
fprintf(fid1, '%s %u\n', ''No. of Panel      ' ' , num_elem ...
    -num_elem_moon-num_elem_free);
for i=1:num_vertices-num_vertices_moon*(div_z-1)/(div_z+1)-num_vertices_free,
    fprintf(fid1, '          %u      %.9f      %.9f      %.9f\n', ov(i), v(i,1), ...
        v(i,2), abs(v(i,3)));
end
if verifica_t==1
    for i=1:num_elem_t
        fprintf(fid1, '%u      %u      %u      %u      %u\n', tri(i,1) ...
            , tri(i,8), tri(i,7), tri(i,8), tri(i,6));
    end
end
if verifica_q==1
    for i=1:num_elem_q-num_elem_moon-num_elem_free,
        fprintf(fid1, '%u      %u      %u      %u      %u\n', quad(i,1), ...
            quad(i,6), quad(i,9), quad(i,7), quad(i,8));
    end
end
end

```

```

fprintf(fid1, ''***** MOON POOL PART GRID *****'\n');
fprintf(fid1, '%s %u\n', ''No. of Grid    ''    ', num_vertices_moon);
fprintf(fid1, '%s %u\n', ''No. of Panel    ''    ', num_elem_moon);
%Re-indexing points and elements to start from number 1 -----%
for i=1:num_vertices_moon,
    index=ov(i+num_vertices-num_vertices_moon*div_z/(div_z+1) ...
        -num_vertices_free);
    ovindezm(i)=i;
    for j=1:num_elem_moon,
        quadindezm(j,1)=j;
        if index==quad(j+num_elem_q-num_elem_moon-num_elem_free,6)
            quadindezm(j,2)=ovindezm(i);
        end
        if index==quad(j+num_elem_q-num_elem_moon-num_elem_free,9)
            quadindezm(j,3)=ovindezm(i);
        end
        if index==quad(j+num_elem_q-num_elem_moon-num_elem_free,7)
            quadindezm(j,4)=ovindezm(i);
        end
        if index==quad(j+num_elem_q-num_elem_moon-num_elem_free,8)
            quadindezm(j,5)=ovindezm(i);
        end
    end
end
for i=(num_vertices-num_vertices_moon*div_z/(div_z+1)-num_vertices_free ...
    +1):(num_vertices-num_vertices_free+num_vertices_moon/(div_z+1)),
    fprintf(fid1, '          %u    %.9f    %.9f    %.9f\n', ...
        ovindezm(i-num_vertices+num_vertices_moon*div_z/(div_z+1) ...
        +num_vertices_free), v(i,1), v(i,2), abs(v(i,3)));
end
if verifica_q==1
    for i=1:num_elem_moon
        fprintf(fid1, '%u    %u    %u    %u    %u\n', quadindezm(i,1), ...
            quadindezm(i,2), quadindezm(i,3), quadindezm(i,4), quadindezm(i,5));
    end
end
fprintf(fid1, ''***** MOON POOL FREE SURFACE PART GRID *****'\n');
fprintf(fid1, '%s %u\n', ''No. of Grid    ''    ', num_vertices_free);
fprintf(fid1, '%s %u\n', ''No. of Panel    ''    ', num_elem_free);
%Re-indexing points and elements to start from number 1 -----%
for i=1:num_vertices_free,
    index=ov(i+num_vertices-num_vertices_free);
    ovindef(i)=i;
    for j=1:num_elem_free,
        quadindexf(j,1)=j;
        if index==quad(j+num_elem_q-num_elem_free,6)
            quadindexf(j,2)=ovindexf(i);
        end
        if index==quad(j+num_elem_q-num_elem_free,9)
            quadindexf(j,3)=ovindexf(i);
        end
        if index==quad(j+num_elem_q-num_elem_free,7)
            quadindexf(j,4)=ovindexf(i);
        end
        if index==quad(j+num_elem_q-num_elem_free,8)
            quadindexf(j,5)=ovindexf(i);
        end
    end
end
for i=num_vertices-num_vertices_free+1:num_vertices,

```

---

```

fprintf(fid1, '          %u      %.9f      %.9f      %.9f\n', ...
    oindexf(i-num_vertices+num_vertices_free), v(i,1), v(i,2), ...
    abs(v(i,3)));
end
if verifica_q==1
    for i=1:num_elem_free
        fprintf(fid1, '%u      %u      %u      %u      %u\n', quadindexf(i,1), ...
            quadindexf(i,2), quadindexf(i,3), quadindexf(i,4), ...
            quadindexf(i,5));
    end
end
fclose(fid1);

end

```

---

## 9. Selecao.m

```

function [ K4, f2, dt1, dt2, radarout ] = Selecao( f, d1, d2, K, a, radar )
% Function which select the individuals with best grades.

f1=f;                % Array of grades.
radarint=zeros(a,7); % Array of radar graphs.
Kint=zeros(6,a);     % Array of selected individuals.
chp_no=0;            % Number of champions found during scan.

% int means intermediary

% Looking for the best grades -----%
for j=1:a,           % 'a' is the amount of best individuals we want. It is the n
    % number of individuals set by the user to be applied to the population.

    max=0;
    dref1=0;
    dref2=0;
    chp_found = 0;
    for i=1:size(f1,2),
        if (f1(i)>max) || ((f1(i)==max)&&(d1(i)>dref1)) || ((f1(i)==max)&& ...
            (d1(i)==dref1)&&(d2(i)>dref2))
            ind_new = 1; % Is the individual a new one?
            for k=1:a,
                if (K(:,i)==Kint(:,k))
                    ind_new = 0;
                end
            end
            if ind_new % In case it is, store its information.
                max=f1(i)      ;
                dref1=d1(i)    ;
                dref2=d2(i)    ;
                c=i            ;
                chp_found = 1;
            end
        end
    end
    if chp_found % Verify if a new champion was found during this iteration
        Kint(:,j)=K(:,c); % and store it in case it was.
        fint(:,j)=f(:,c);
    end
end

```

```

        dlint(:,j)=d1(c);
        d2int(:,j)=d2(c);
        radarint(j,:)=radar(c,:);
        f1(c)=0;
        chp_no=chp_no+1; % Total number of selected individuals during
    end % all iterations.
end
%-----%

if chp_no==a
    K4=Kint;
    f2=fint;
    dt1=dlint;
    dt2=d2int;
    radarout=radarint;
elseif a>chp_no
    a_double=double(a);
    int_part=fix(a_double/chp_no);
    rem_part=rem(a_double,chp_no);
    seq=zeros(1,chp_no);
    seq(:)=int_part;
    for i=1:rem_part
        seq(1,i)=seq(1,i)+1;
    end
    coord=1;
    for i=1:size(seq,2)
        for j=1:seq(1,i)
            K4(:,coord)=Kint(:,i);
            f2(:,coord)=fint(:,i);
            dt1(:,coord)=dlint(i);
            dt2(:,coord)=d2int(i);
            radarout(coord,:)=radarint(i,:);
            coord=coord+1;
        end
    end
end
end

end

```

---

## 10. Tres\_Height.m

```

function [H_tres] = Tres_Height(mode, rao, T)

% Choose mode according to criterion:
% 1: wave overflow
% 2: stroke compensation
% 3: bending moment
% 4: drift force

if mode == 1 || mode == 2 || mode == 3
    % Read input parameters.-----%
    fileID = fopen('Optimization_Parameters.dat','r');
    line_data = textscan(fileID, '%s[^\n]' %f', 3, 'HeaderLines', ...
        7+3*(mode-1));

    xq = line_data{1}(1);

```

```

n = line_data{1}(2);
m = line_data{1}(3);

fclose(fileID);
% Finish reading input parameters.-----%
elseif mode == 4
% Read input parameters.-----%
fileID = fopen('Optimization_Parameters.dat','r');
line_data = textscan(fileID, '%*[^']' %f', 1, 'HeaderLines', 16);

Dy0 = line_data{1}(1);

fclose(fileID);
% Finish reading input parameters.-----%
end

if mode == 1 || mode == 2 || mode == 3
H_tres = xq/(rao*(2*log(m*3600/(T*n)))^0.5);
elseif mode ==4
H_tres = (Dy0/(rao*(2*pi/T)))^(1/3);
end

end

```

---

## 11. Optimization\_Parameters.dat

```

'Total number of individuals to each evaluation. ni =' 10
'Based on the typical moon pool lenght, set values for L1 ='
10 10.5 11 11.5 12 12.25 12.5 13
'Based on the typical moon pool width, set values for L2 ='
5 5.25 5.5 5.75 6 6.25 6.5 6.75 7
'Percentage of the individuals to be selected as the best. p_no =' 0.5
'Number of panel rows in the prismatic section of the moon pool. div_z =' 4
'Wave overflow threshold value xq =' 4
'Wave overflow exceedence number n =' 15
'Wave overflow time span m =' 3
'Stroke compensation threshold value xq =' 3.5
'Stroke compensation exceedance number n =' 15
'Stroke compensation time span m =' 3
'Bending moment threshold value xq =' 9999
'Bending moment exceedence number n =' 9999
'Bending moment time span m =' 9999
'Drift force Dy0 =' 34260
'Choose sea states to perform analysis. Binary value.'
'IACS LONG CRESTED' 0
'IACS SHORT CRESTED' 0
'JONSWAP LONG CRESTED' 0
'JONSWAP SHORT CRESTED' 1
'Bretschneider LONG CRESTED' 0
'Bretschneider SHORT CRESTED' 0
'Choose significant wave periods to perform analysis. Integer from 4 to 18 seconds, inclusive.'
4 5 6 7 9 10 11 12 13 14 18

```

---

## 12. Teste.txt - Malha de entrada do casco para análise.

|     |              |             |              |
|-----|--------------|-------------|--------------|
| 154 | 126          |             |              |
| 1   | 100.6        | 0           | 0            |
| 2   | 100.13533333 | 0           | -4.354666667 |
| 3   | 99.646       | 0           | -8.71        |
| 4   | 99.42133333  | 0           | -10.742      |
| 5   | 98.09066667  | 0           | -11.61266667 |
| 6   | 95.52        | 1.364       | 0            |
| 7   | 95.52        | 1.212       | -4.354666667 |
| 8   | 95.52        | 1.042666667 | -8.71        |
| 9   | 95.52        | 0.658666667 | -10.742      |
| 10  | 95.52        | 0           | -11.61266667 |
| 11  | 90.44        | 2.815333333 | 0            |
| 12  | 90.44        | 2.619333333 | -4.354666667 |
| 13  | 90.44        | 2.288666667 | -8.71        |
| 14  | 90.44        | 1.478666667 | -10.742      |
| 15  | 90.44        | 0           | -11.61266667 |
| 16  | 85.36        | 4.499333333 | 0            |
| 17  | 85.36        | 4.172666667 | -4.354666667 |
| 18  | 85.36        | 3.65        | -8.71        |
| 19  | 85.36        | 2.459333333 | -10.742      |
| 20  | 85.36        | 0           | -11.61266667 |
| 21  | 80.28        | 6.240666667 | 0            |
| 22  | 80.28        | 5.842       | -4.354666667 |
| 23  | 80.28        | 5.069333333 | -8.71        |
| 24  | 80.28        | 3.574666667 | -10.742      |
| 25  | 80.28        | 0           | -11.61266667 |
| 26  | 70.12        | 9.506666667 | 0            |
| 27  | 70.12        | 8.896666667 | -4.354666667 |
| 28  | 70.12        | 7.966666667 | -8.71        |
| 29  | 70.12        | 6.048       | -10.742      |
| 30  | 70.12        | 0           | -11.61266667 |
| 31  | 59.96        | 12.00266667 | 0            |
| 32  | 59.96        | 11.466      | -4.354666667 |
| 33  | 59.96        | 10.48666667 | -8.71        |
| 34  | 59.96        | 8.4         | -10.742      |
| 35  | 59.96        | 4.89        | -11.61266667 |
| 36  | 59.96        | 0           | -11.61266667 |
| 37  | 49.8         | 13.57       | 0            |
| 38  | 49.8         | 13.236      | -4.354666667 |
| 39  | 49.8         | 12.39866667 | -8.71        |
| 40  | 49.8         | 10.52333333 | -10.742      |
| 41  | 49.8         | 7.228666667 | -11.61266667 |
| 42  | 49.8         | 0           | -11.61266667 |
| 43  | 39.64        | 14.296      | 0            |
| 44  | 39.64        | 14.158      | -4.354666667 |
| 45  | 39.64        | 13.64466667 | -8.71        |
| 46  | 39.64        | 12.042      | -10.742      |
| 47  | 39.64        | 9.287333333 | -11.61266667 |
| 48  | 39.64        | 4.644       | -11.61266667 |
| 49  | 39.64        | 0           | -11.61266667 |
| 50  | 29.48        | 14.51333333 | 0            |
| 51  | 29.48        | 14.45533333 | -4.354666667 |
| 52  | 29.48        | 14.25266667 | -8.71        |
| 53  | 29.48        | 13.036      | -10.742      |
| 54  | 29.48        | 10.50733333 | -11.61266667 |
| 55  | 29.48        | 5.254       | -11.61266667 |
| 56  | 29.48        | 0           | -11.61266667 |
| 57  | 19.32        | 14.51333333 | 0            |
| 58  | 19.32        | 14.51333333 | -4.354666667 |
| 59  | 19.32        | 14.48466667 | -8.71        |

|     |        |             |              |
|-----|--------|-------------|--------------|
| 60  | 19.32  | 13.43933333 | -10.742      |
| 61  | 19.32  | 11.16733333 | -11.61266667 |
| 62  | 19.32  | 5.584       | -11.61266667 |
| 63  | 19.32  | 0           | -11.61266667 |
| 64  | 9.16   | 14.51333333 | 0            |
| 65  | 9.16   | 14.51333333 | -4.354666667 |
| 66  | 9.16   | 14.48466667 | -8.71        |
| 67  | 9.16   | 13.43933333 | -10.742      |
| 68  | 9.16   | 11.19       | -11.61266667 |
| 69  | 9.16   | 5.594666667 | -11.61266667 |
| 70  | 9.16   | 0           | -11.61266667 |
| 71  | -1     | 14.51333333 | 0            |
| 72  | -1     | 14.51333333 | -4.354666667 |
| 73  | -1     | 14.48466667 | -8.71        |
| 74  | -1     | 13.43933333 | -10.742      |
| 75  | -1     | 11.19       | -11.61266667 |
| 76  | -1     | 5.594666667 | -11.61266667 |
| 77  | -1     | 0           | -11.61266667 |
| 78  | -11.16 | 14.51333333 | 0            |
| 79  | -11.16 | 14.51333333 | -4.354666667 |
| 80  | -11.16 | 14.48466667 | -8.71        |
| 81  | -11.16 | 13.39933333 | -10.742      |
| 82  | -11.16 | 11.19       | -11.61266667 |
| 83  | -11.16 | 5.594666667 | -11.61266667 |
| 84  | -11.16 | 0           | -11.61266667 |
| 85  | -21.32 | 14.51333333 | 0            |
| 86  | -21.32 | 14.51333333 | -4.354666667 |
| 87  | -21.32 | 14.32533333 | -8.71        |
| 88  | -21.32 | 13.144      | -10.742      |
| 89  | -21.32 | 10.75333333 | -11.61266667 |
| 90  | -21.32 | 5.376666667 | -11.61266667 |
| 91  | -21.32 | 0           | -11.61266667 |
| 92  | -31.48 | 14.51333333 | 0            |
| 93  | -31.48 | 14.46266667 | -4.354666667 |
| 94  | -31.48 | 13.876      | -8.71        |
| 95  | -31.48 | 12.324      | -10.742      |
| 96  | -31.48 | 9.567333333 | -11.61266667 |
| 97  | -31.48 | 4.784       | -11.61266667 |
| 98  | -31.48 | 0           | -11.61266667 |
| 99  | -41.64 | 14.38266667 | 0            |
| 100 | -41.64 | 14.05666667 | -4.354666667 |
| 101 | -41.64 | 12.848      | -8.71        |
| 102 | -41.64 | 10.95333333 | -10.742      |
| 103 | -41.64 | 7.888666667 | -11.61266667 |
| 104 | -41.64 | 3.944666667 | -11.61266667 |
| 105 | -41.64 | 0           | -11.61266667 |
| 106 | -51.8  | 13.94733333 | 0            |
| 107 | -51.8  | 13.026      | -4.354666667 |
| 108 | -51.8  | 11.124      | -8.71        |
| 109 | -51.8  | 9.072       | -10.742      |
| 110 | -51.8  | 5.953333333 | -11.61266667 |
| 111 | -51.8  | 0           | -11.61266667 |
| 112 | -61.96 | 13.01866667 | 0            |
| 113 | -61.96 | 11.25533333 | -4.354666667 |
| 114 | -61.96 | 8.763333333 | -8.71        |
| 115 | -61.96 | 6.854       | -10.742      |
| 116 | -61.96 | 3.849333333 | -11.61266667 |
| 117 | -61.96 | 0           | -11.61266667 |
| 118 | -72.12 | 11.43666667 | 0            |
| 119 | -72.12 | 8.642666667 | -4.354666667 |

|     |              |             |              |
|-----|--------------|-------------|--------------|
| 120 | -72.12       | 6.184666667 | -8.71        |
| 121 | -72.12       | 4.542666667 | -10.742      |
| 122 | -72.12       | 2.081333333 | -11.61266667 |
| 123 | -72.12       | 0           | -11.61266667 |
| 124 | -82.28       | 8.911333333 | 0            |
| 125 | -82.28       | 6.168       | -2.903333333 |
| 126 | -82.28       | 4.557333333 | -5.806666667 |
| 127 | -82.28       | 3.548666667 | -8.71        |
| 128 | -82.28       | 2.580666667 | -10.742      |
| 129 | -82.28       | 0           | -11.61266667 |
| 130 | -87.36       | 7.242       | 0            |
| 131 | -87.36       | 4.238       | -2.903333333 |
| 132 | -87.36       | 3.004666667 | -5.806666667 |
| 133 | -87.36       | 2.39        | -8.71        |
| 134 | -87.36       | 1.693333333 | -10.742      |
| 135 | -87.36       | 0           | -11.61266667 |
| 136 | -92.44       | 5.341333333 | 0            |
| 137 | -92.44       | 2.38        | -2.903333333 |
| 138 | -92.44       | 1.552666667 | -5.806666667 |
| 139 | -92.44       | 1.289333333 | -8.71        |
| 140 | -92.44       | 0.940666667 | -10.742      |
| 141 | -92.44       | 0           | -11.61266667 |
| 142 | -97.52       | 3.309333333 | 0            |
| 143 | -97.52       | 0.624       | -2.903333333 |
| 144 | -97.52       | 0.348       | -5.806666667 |
| 145 | -97.52       | 0.318666667 | -8.71        |
| 146 | -97.52       | 0.295333333 | -10.742      |
| 147 | -97.52       | 0           | -11.61266667 |
| 148 | -102.6       | 1.292       | 0            |
| 149 | -99.20666667 | 0           | -2.903333333 |
| 150 | -98.942      | 0           | -5.806666667 |
| 151 | -98.88133333 | 0           | -8.71        |
| 152 | -98.84066667 | 0           | -10.742      |
| 153 | -98.79       | 0           | -11.61266667 |
| 154 | -102.7666667 | 0           | 0            |

|    |    |    |    |    |
|----|----|----|----|----|
| 1  | 1  | 2  | 7  | 6  |
| 2  | 2  | 3  | 8  | 7  |
| 3  | 3  | 4  | 9  | 8  |
| 4  | 4  | 5  | 10 | 9  |
| 5  | 6  | 7  | 12 | 11 |
| 6  | 7  | 8  | 13 | 12 |
| 7  | 8  | 9  | 14 | 13 |
| 8  | 9  | 10 | 15 | 14 |
| 9  | 11 | 12 | 17 | 16 |
| 10 | 12 | 13 | 18 | 17 |
| 11 | 13 | 14 | 19 | 18 |
| 12 | 14 | 15 | 20 | 19 |
| 13 | 16 | 17 | 22 | 21 |
| 14 | 17 | 18 | 23 | 22 |
| 15 | 18 | 19 | 24 | 23 |
| 16 | 19 | 20 | 25 | 24 |
| 17 | 21 | 22 | 27 | 26 |
| 18 | 22 | 23 | 28 | 27 |
| 19 | 23 | 24 | 29 | 28 |
| 20 | 24 | 25 | 30 | 29 |
| 21 | 26 | 27 | 32 | 31 |
| 22 | 27 | 28 | 33 | 32 |
| 23 | 28 | 29 | 34 | 33 |
| 24 | 29 | 30 | 35 | 34 |
| 25 | 30 | 36 | 35 | 35 |

---

|    |    |     |     |     |
|----|----|-----|-----|-----|
| 26 | 31 | 32  | 38  | 37  |
| 27 | 32 | 33  | 39  | 38  |
| 28 | 33 | 34  | 40  | 39  |
| 29 | 34 | 35  | 41  | 40  |
| 30 | 35 | 36  | 42  | 41  |
| 31 | 37 | 38  | 44  | 43  |
| 32 | 38 | 39  | 45  | 44  |
| 33 | 39 | 40  | 46  | 45  |
| 34 | 40 | 41  | 47  | 46  |
| 35 | 41 | 42  | 48  | 47  |
| 36 | 42 | 49  | 48  | 48  |
| 37 | 43 | 44  | 51  | 50  |
| 38 | 44 | 45  | 52  | 51  |
| 39 | 45 | 46  | 53  | 52  |
| 40 | 46 | 47  | 54  | 53  |
| 41 | 47 | 48  | 55  | 54  |
| 42 | 48 | 49  | 56  | 55  |
| 43 | 50 | 51  | 58  | 57  |
| 44 | 51 | 52  | 59  | 58  |
| 45 | 52 | 53  | 60  | 59  |
| 46 | 53 | 54  | 61  | 60  |
| 47 | 54 | 55  | 62  | 61  |
| 48 | 55 | 56  | 63  | 62  |
| 49 | 57 | 58  | 65  | 64  |
| 50 | 58 | 59  | 66  | 65  |
| 51 | 59 | 60  | 67  | 66  |
| 52 | 60 | 61  | 68  | 67  |
| 53 | 61 | 62  | 69  | 68  |
| 54 | 62 | 63  | 70  | 69  |
| 55 | 64 | 65  | 72  | 71  |
| 56 | 65 | 66  | 73  | 72  |
| 57 | 66 | 67  | 74  | 73  |
| 58 | 67 | 68  | 75  | 74  |
| 59 | 68 | 69  | 76  | 75  |
| 60 | 69 | 70  | 77  | 76  |
| 61 | 71 | 72  | 79  | 78  |
| 62 | 72 | 73  | 80  | 79  |
| 63 | 73 | 74  | 81  | 80  |
| 64 | 74 | 75  | 82  | 81  |
| 65 | 75 | 76  | 83  | 82  |
| 66 | 76 | 77  | 84  | 83  |
| 67 | 78 | 79  | 86  | 85  |
| 68 | 79 | 80  | 87  | 86  |
| 69 | 80 | 81  | 88  | 87  |
| 70 | 81 | 82  | 89  | 88  |
| 71 | 82 | 83  | 90  | 89  |
| 72 | 83 | 84  | 91  | 90  |
| 73 | 85 | 86  | 93  | 92  |
| 74 | 86 | 87  | 94  | 93  |
| 75 | 87 | 88  | 95  | 94  |
| 76 | 88 | 89  | 96  | 95  |
| 77 | 89 | 90  | 97  | 96  |
| 78 | 90 | 91  | 98  | 97  |
| 79 | 92 | 93  | 100 | 99  |
| 80 | 93 | 94  | 101 | 100 |
| 81 | 94 | 95  | 102 | 101 |
| 82 | 95 | 96  | 103 | 102 |
| 83 | 96 | 97  | 104 | 103 |
| 84 | 97 | 98  | 105 | 104 |
| 85 | 99 | 100 | 107 | 106 |

---

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 86  | 100 | 101 | 108 | 107 |
| 87  | 101 | 102 | 109 | 108 |
| 88  | 102 | 103 | 110 | 109 |
| 89  | 103 | 104 | 111 | 110 |
| 90  | 104 | 105 | 111 | 111 |
| 91  | 106 | 107 | 113 | 112 |
| 92  | 107 | 108 | 114 | 113 |
| 93  | 108 | 109 | 115 | 114 |
| 94  | 109 | 110 | 116 | 115 |
| 95  | 110 | 111 | 117 | 116 |
| 96  | 112 | 113 | 119 | 118 |
| 97  | 113 | 114 | 120 | 119 |
| 98  | 114 | 115 | 121 | 120 |
| 99  | 115 | 116 | 122 | 121 |
| 100 | 116 | 117 | 123 | 122 |
| 101 | 118 | 119 | 125 | 124 |
| 102 | 119 | 120 | 126 | 125 |
| 103 | 120 | 121 | 127 | 126 |
| 104 | 121 | 122 | 128 | 127 |
| 105 | 122 | 123 | 129 | 128 |
| 106 | 124 | 125 | 131 | 130 |
| 107 | 125 | 126 | 132 | 131 |
| 108 | 126 | 127 | 133 | 132 |
| 109 | 127 | 128 | 134 | 133 |
| 110 | 128 | 129 | 135 | 134 |
| 111 | 130 | 131 | 137 | 136 |
| 112 | 131 | 132 | 138 | 137 |
| 113 | 132 | 133 | 139 | 138 |
| 114 | 133 | 134 | 140 | 139 |
| 115 | 134 | 135 | 141 | 140 |
| 116 | 136 | 137 | 143 | 142 |
| 117 | 137 | 138 | 144 | 143 |
| 118 | 138 | 139 | 145 | 144 |
| 119 | 139 | 140 | 146 | 145 |
| 120 | 140 | 141 | 147 | 146 |
| 121 | 142 | 143 | 149 | 148 |
| 122 | 143 | 144 | 150 | 149 |
| 123 | 144 | 145 | 151 | 150 |
| 124 | 145 | 146 | 152 | 151 |
| 125 | 146 | 147 | 153 | 152 |
| 126 | 148 | 149 | 154 | 154 |

# ANEXO A – Códigos mantidos para a compatibilização

## 1. Avaliacao.m

```
function [ q ] = Avaliacao( F, q, t )
%Function which evaluate if it is the moment to stop looping, changing 'q' from 1 to 0.

a1=F(t)-F(t-1) ; % Compare the current value of 'F' with the last one.

a2=F(t-1)-F(t-2); % Compare the last value of 'F' with the value before it.

if (a1<0.1)&&(a2<0.1)
    q=0; % If in the tree last loops, 'F' did not increase so much, it is better to stop.
end

end
```

---

## 2. Crossover.m

```
function [ I ] = Crossover( I4, I, ni )
% Function which generate new individuals as sons of two input individuals.
% The 'father' is the first column of 'I4' and the 'mother' is the second.

for i=1:ni,
    for j=1:size(I4,1),
        a=randi(2,1,1); % If 'a'=1, the new individual receives
                        % information from its 'father'.

        I(j,i)=I4(j,a); % If 'a'=2, the new individual receives
                        % information from its 'mother'.
    end
end

end
```

---

## 3. Geracao\_de\_Individuos.m

```
function [ I ] = Geracao_de_Individuos( p, n )
% This is the random generation of individuals just for the first loop.

for i=1:n,
    for j=1:6, % There are 6 parameters.
        e=randi(size(p,1),1,1); % Choose a random position in 'p'.
        I(j,i)=p(e,j); % Assign that position value in 'I'.
    end
end
```

```
end
end
end
```

#### 4. Geracao\_de\_Parametros.m

```
function [ m1, m2, m3, m4, nf, nt, p ] = Geracao_de_Parametros (l1, l2)
% Function which fix the available values for each parameter.]

dy=l2/9; % This way, we can have ten possible values for m1 and m5
dx=l1/9; % This way, we can have ten possible values for m2 and m3.

t=1;
for i=0:dy:l2,
    m1(t)=i; % Construction of m1.
    m4(t)=i; % Construction of m-2
    t=t+1;
end

t=1;
for i=0:dx:l1,
    m2(t)=i; % Construction of m2.
    m3(t)=i; % Construction of m3.
    t=t+1;
end

nf=[1 2 3 4 5 6 7 8 9 10]; % number of sides of the quarter of polygon
                                % which will represent the first rounding
                                % corner. A 'Ten sides' is enough refined.

nt=[1 2 3 4 5 6 7 8 9 10]; % number of sides of the quarter of polygon
                                % which will represent the second rounding
                                % corner. A 'Ten sides' is enough refined.

p(:,1)=m1; %
p(:,2)=m2; %
p(:,3)=m3; % ASSIGNING VALUES.
p(:,4)=m4; %
p(:,5)=nf; %
p(:,6)=nt; %

end
```

## 5. Mutacao.m

[illegible]

```

                                % with the originals to be modified.

for i=1:size(I2,2),
    e1=randi(size(I2,1),1,3);
    [ H ] = Geracao_de_Individuos( p, no ); % Generate random individuals
                                           % in 'H'.

    I21(e1(1),i)=H(e1(1),i)                ; % Modify the individuals from
                                           % 'I21' with parts of the 'H'
                                           % ones.

    I21(e1(2),i)=H(e1(2),i)                ; % Same as above.
    I21(e1(3),i)=H(e1(3),i)                ; % Same as above.
end

for i=1:size(I2,2),
    I22(:,i)=I2(:,i); % Put the originals in 'I22'.
end

b=size(I2,2);

for i=1:size(I21,2),
    I22(:,b+i)=I21(:,i); % Put the mutants in 'I22'.
end

end

```

---

## 6. Pre\_Leitor.m

```

function [ quad,tri,bound,pontos,v,ov,verifica_t,verifica_q,verifica_b, ...
          num_elem, num_vertices, num_elem_t, num_elem_q, ...
          num_elem_b, e ] = Pre_Leitor
% Function which reads the data from the ship geometry file and make data
% matrixes. -----%

% Catch the amount of points and elements -----%
[num_vertices num_elem]=textread('teste.txt',' %u %u', 1);
v=zeros(num_vertices,2)                ;
ov=zeros(num_vertices,1)                ;
% -----%

% Catch the points coordinates from the ship geometry file -----%
for i=1:num_vertices,
    [a b c d]=textread('teste.txt','%f %f %f %f',1,...
        'headerlines',i);
    v(i,1)=b                ;
    v(i,2)=c                ;
    v(i,3)=d                ;
    ov(i)=i                 ;
end
% -----%

% Catch the elements from the ship geometry file -----%
for i=1:num_elem,
    [e(i,1) e(i,2) e(i,3) e(i,4) e(i,5)]=textread('teste.txt',...
        '%u %u %u %u %u',1,'headerlines',i+num_vertices);

```

```

end
%-----%

% Write the element data in gmsh format, this way we can visualize the ship
% computational model by gmsh -----%
pontos=[] ;
bound=[] ;
num_elem_b=0;
num_elem_t=1;
num_elem_q=1;

for i=1:num_elem,
    if e(i,4)==e(i,5)
        tri(num_elem_t,1)=num_elem_t+num_elem_b;
        tri(num_elem_t,2)=2 ;
        tri(num_elem_t,3)=2 ;
        tri(num_elem_t,4)=0 ;
        tri(num_elem_t,5)=6 ;
        tri(num_elem_t,6)=e(i,2) ;
        tri(num_elem_t,7)=e(i,3) ;
        tri(num_elem_t,8)=e(i,4) ;
        num_elem_t=num_elem_t+1 ;
    else
        quad(num_elem_q,1)=num_elem_q+num_elem_b+num_elem_t;
        quad(num_elem_q,2)=3 ;
        quad(num_elem_q,3)=2 ;
        quad(num_elem_q,4)=0 ;
        quad(num_elem_q,5)=6 ;
        quad(num_elem_q,6)=e(i,2) ;
        quad(num_elem_q,7)=e(i,3) ;
        quad(num_elem_q,8)=e(i,4) ;
        quad(num_elem_q,9)=e(i,5) ;
        num_elem_q=num_elem_q+1 ;
    end
end

num_elem_t=num_elem_t-1;
num_elem_q=num_elem_q-1;

for i=1:size(quad,1),
    quad(i,1)=num_elem_t+i;
end

if num_elem_t==0
    verifica_t=0;
else
    verifica_t=1;
end
if num_elem_q==0
    verifica_q=0;
else
    verifica_q=1;
end
if num_elem_b==0
    verifica_b=0;
else
    verifica_b=1;
end
%-----%

```

end

---

## 7. Z\_0.m

```
function [ z, o ] = Z_0( v )
% Function which find the 'x' value which is nearest from '0' and the
% lowest value for 'z'. -----%

s=v(:,3); % 'z' values.
z=min(s); % Lowest 'z' values.

% It is known that, in our type of ship file, the 'zero point' is put on
% the midship position, so we need to find the nearest coordinate from the
% 'zero point' to keep the moonpool centralized. -----%

for i=1:size(v,1),
    if (v(i,1)>0)&&(v(i+1,1)<0)
        o=v(i+1,1); % Nearest x coordinate from 'zero point'.
    end
end

end
```