

UNIVERSIDADE FEDERAL DE ALAGOAS  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS GRADUAÇÃO EM ENGENHARIA CIVIL

WEVERTON MARQUES DA SILVA

**MODELAGEM E IMPLEMENTAÇÃO DE UMA FERRAMENTA COMPUTACIONAL  
INTEGRADA PARA ANÁLISE DE FADIGA EM DUTOS SUBMARINOS EM  
VÃO-LIVRE**

Maceió-AL

Março de 2021

WEVERTON MARQUES DA SILVA

**MODELAGEM E IMPLEMENTAÇÃO DE UMA FERRAMENTA COMPUTACIONAL  
INTEGRADA PARA ANÁLISE DE FADIGA EM DUTOS SUBMARINOS EM  
VÃO-LIVRE**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Engenharia Civil do Centro de Tecnologia da Universidade Federal de Alagoas.

Orientador: Adeildo Soares Ramos Júnior  
Coorientador: Eduardo Setton S. da Silveira

Maceió-AL

Março de 2021

**Catálogo na fonte**  
**Universidade Federal de Alagoas**  
**Biblioteca Central**  
**Divisão de Tratamento Técnico**

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 – 1767

S586m Silva, Weverton Marques da.  
Modelagem e implementação de uma ferramenta computacional integrada para análise de fadiga em dutos submarinos de vão-livre / Weverton Marques da Silva. - 2021.  
66 f. : il.

Orientador: Adeildo Soares Ramos Júnior.  
Co-orientador: Eduardo Setton S. da Silveira.  
Dissertação (Mestrado em Engenharia Civil) – Universidade Federal de Alagoas. Centro de Tecnologia. Maceió, 2020.

Bibliografia: f. 53-54.  
Apêndices: f. 56-66.

1. Dutos. 2. Aço - Fadiga. 3. *Framework* (Arquivo de computador). I. Título.

CDU:624



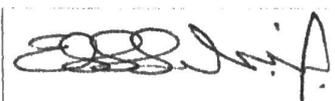
**MODELAGEM E DESENVOLVIMENTO DE UMA FERRAMENTA INTEGRADA PARA  
ANÁLISE DE FADIGA EM DUTOS SUBMARINOS EM VÃO-LIVRE**

**WEVERTON MARQUES DA SILVA**

Dissertação submetida à banca examinadora do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas e aprovada no dia 06 do mês de janeiro do ano de 2021.

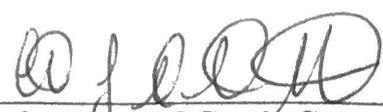
Banca Examinadora:

  
\_\_\_\_\_  
**Prof. Dr. Adeildo Soares Ramos Júnior**  
(Orientador – PPGEC/UFAL)

  
\_\_\_\_\_  
**Prof. Dr. Eduardo Setton Sampaio da Silveira**  
(Coorientador - PPGEC/UFAL)

  
\_\_\_\_\_  
**Prof. Dr. William Wagner Matos Lira**  
(Avaliador Interno – PPGEC/UFAL)

  
\_\_\_\_\_  
**Prof. Dr. Leandro Melo de Sales**  
(Avaliador Externo- IC/UFAL)

  
\_\_\_\_\_  
**Pesquisadora Dr.ª Cynthia Gonçalves da Costa Matt**  
(Avaliadora Externa – PETROBRAS)

*Ao meu falecido pai, Joel.*

## **AGRADECIMENTOS**

Aos meus familiares por todo apoio. Especialmente, ao meu irmão, Wagner, e a minha noiva, Jhulia, que estiveram sempre ao meu lado e me ajudaram a encontrar a disposição para seguir em frente.

Agradeço ao orientador e coorientador pelos direcionamentos e por acreditaram na importância dos frutos desse trabalho.

Ao Laboratório de Computação Científica e Visualização pela minha participação no projeto IntegriSpan. Especialmente aos meus amigos Emerson, Josué, Jéssica e Renato. Sem a ajuda e parceria inestimável deles este trabalho não seria possível.

Aos professores do Programa de Pós-Graduação em Engenharia Civil que, com os conhecimentos transmitidos nas disciplinas, me ajudaram a elaborar esse trabalho.

*“Seja curioso. Leia muito. Experimente coisas novas.  
Eu acho que muito do que as pessoas chamam de inteligência  
apenas se resume a curiosidade.”  
(Aaron Swartz)*

## RESUMO

A segurança no transporte de hidrocarbonetos e outros fluidos em dutos é um dos tópicos de maior atenção na indústria de petróleo e gás. Recentemente, essa indústria tem tido grandes preocupações sobre como o fenômeno da Vibração Induzida por Vórtice (VIV) afeta a vida à fadiga dos componentes e dutos submarinos, especialmente aqueles em vão-livre. Mundialmente, há várias dessas estruturas instaladas que já chegaram ao final ou metade de sua vida de projeto, o que torna imperativo revisitar as previsões iniciais e avaliar a necessidade de intervenção no sentido de estender a vida útil das mesmas, em especial no que diz respeito à VIV. Essas e outras análises se baseiam na correta representação no comportamento real do duto. O Método dos Elementos Finitos é amplamente empregado na modelagem desse comportamento. No entanto, essa não é uma tarefa trivial, uma vez que depende da utilização de vários softwares para etapas intermediárias. Além disso, essas análises envolvem manipulação de grandes quantidades de dados relacionados à batimetria do fundo marinho e propriedades dos materiais, geometria do duto, correnteza, entre outros. Dessa forma, todo esse processo costuma ser demorado e passível de erros. Concomitantemente, a indústria de óleo e gás tem passado por uma transformação digital nos anos recentes. Este cenário trouxe oportunidades para o desenvolvimento de novas ferramentas para o dia-a-dia do profissional responsável por estas análises. Neste contexto, este trabalho apresenta o desenvolvimento e estrutura de uma ferramenta computacional com funcionalidades do pré ao pós-processamento de dados da análise que visam melhorar o fluxo de análise de fadiga em dutos submarinos sujeitos a vãos-livres. Desenvolvida com base no estudo da rotina de trabalho de profissionais da PETROBRAS, a ferramenta consegue cobrir as etapas de pré-processamento e pós-processamento automatizando praticamente todo o fluxo da análise de estrutural e fadiga. Dessa forma, além de reduzir o tempo total das etapas de pré-processamento e pós-processamento de dias para minutos, a ferramenta conseguiu trazer padronização e redução na curva de aprendizagem para novos profissionais.

**Palavras-chaves:** Dutos Submarinos; Análise de fadiga; Framework Computacional.

## ABSTRACT

Safety in the transportation of hydrocarbons and other fluids in pipelines is one of the topics of greatest attention in the oil and gas industry. Recently, this industry has had great concerns about how the phenomenon of Vortex Induced Vibration (VIV) affects the fatigue life of submarine components and pipes, especially free-span pipelines. Worldwide, there are several of these structures installed that have already reached the end or half of their design lifespan, which makes it imperative to revisit the initial predictions and assess the need for intervention in order to extend their lifespan, especially with regard to VIV. These and other analyzes are based on the correct representation of the pipeline's real behavior. The Finite Element Method is widely used to model this behavior. However, this is not a trivial task, since it depends on the use of various software for intermediate steps. In addition, these analyzes involve manipulation of large amounts of data related to the bathymetry of the seabed and material properties, pipe geometry, current, among others. Thus, this entire process is usually time-consuming and prone to errors. Simultaneously, the oil and gas industry has undergone a digital transformation in recent years. This scenario brought opportunities for the development of new tools for the professional responsible for these analyzes. This work presents the development and structure of a computational framework with functionalities from pre to post-processing of analysis data that aim to improve the flow of fatigue analysis in subsea pipelines subject to free spans. Developed based on the study of the work routine of PETROBRAS professionals, the tool is able to cover the pre-processing and post-processing stages by automating practically the entire flow of structural and fatigue analysis. Thus, in addition to reducing the total time of the pre-processing and post-processing steps from days to minutes, the tool was able to bring standardization and reduction in the learning curve for new professionals.

**Keywords:** Subsea Pipeline; Fatigue Analysis; Computational Framework Modeling.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Fluxo de avaliação de vida à fadiga em dutos em vão-livre. . . . .	15
Figura 2 – Interface gráfica do SAGE Profile. . . . .	17
Figura 3 – Interface gráfica (planilha) da FatFree. . . . .	18
Figura 4 – Fluxograma das etapas do trabalho. . . . .	21
Figura 5 – Modelo de elementos finitos durante o lançamento. . . . .	23
Figura 6 – Tipos de elementos usados no modelo. . . . .	25
Figura 7 – Elemento PIPE31. . . . .	26
Figura 8 – Elemento rígido R3D4 (a), e suavização da superfície de elementos (b). . . .	27
Figura 9 – Exemplo de superfície analítica rígida. . . . .	28
Figura 10 – Esteira de Von Kármán. . . . .	29
Figura 11 – Duto em Vão-livre e direções das oscilações. . . . .	30
Figura 12 – Configurações típicas para vãos. . . . .	31
Figura 13 – Curva de modelo de resposta <i>in-line</i> . . . . .	32
Figura 14 – Curva de modelo de resposta <i>cross-flow</i> . . . . .	33
Figura 15 – Fluxo de operação proposto para a ferramenta. . . . .	36
Figura 16 – Digrama UML do pacote <code>analysis</code> . . . . .	39
Figura 17 – Digrama UML do pacote <code>odb_handler</code> . . . . .	39
Figura 18 – Digrama UML do pacote <code>mode_selector</code> . . . . .	40
Figura 19 – Digrama UML do pacote <code>dnv</code> . . . . .	41
Figura 20 – Exemplo de gráfico customizado criado com a biblioteca Bokeh. . . . .	41
Figura 21 – Digrama UML do pacote <code>plots</code> . . . . .	42
Figura 22 – Digrama UML global . . . . .	43
Figura 23 – Código para carregamento dos dados de entrada. . . . .	44
Figura 24 – Código para geração de Figura 25 com o perfil da batimetria . . . . .	45
Figura 25 – Perfil do modelo. . . . .	45
Figura 26 – Código para geração do gráfico do perfil da configuração deformada do duto sobre a batimetria. . . . .	46
Figura 27 – Configuração deformada do duto após a simulação. . . . .	47
Figura 28 – Modos de vibração <i>in-line</i> geradas na análise modal. . . . .	47
Figura 29 – Modos de vibração <i>cross-flow</i> geradas na análise modal. . . . .	47
Figura 30 – Modos de vibração <i>in-line</i> selecionados pelo algoritmo implementado. . . .	48
Figura 31 – Modos de vibração <i>cross-flow</i> selecionados pelo algoritmo implementado. .	48
Figura 32 – Dados inseridos na aba <i>Multimode</i> da FatFree. . . . .	49
Figura 33 – Código para a geração dos gráficos de vida à fadiga. . . . .	49
Figura 34 – Gráfico de vida à fadiga e perfil do duto. . . . .	50

Figura 35 – Exemplo de arquivo de entrada de dados: chave MODEL. . . . .	56
Figura 36 – Exemplo de arquivo de entrada de dados: chave FILE_BAT. . . . .	57
Figura 37 – Exemplo de arquivo de entrada de dados: chave CONDITIONS. . . . .	58
Figura 38 – Exemplo de arquivo de entrada de dados: chave MODE_SELECTOR. . . . .	59
Figura 39 – Exemplo de arquivo de entrada de dados: chave PIPE_GEOMETRY. . . . .	59
Figura 40 – Exemplo de arquivo de entrada de dados: chave AUXILIARY_NODE. . . . .	60
Figura 41 – Exemplo de arquivo de entrada de dados: chave CURTAIN_SPRINGS. . . . .	60
Figura 42 – Exemplo de arquivo de entrada de dados: chave SPRING_STIFFNESS. . . . .	61
Figura 43 – Exemplo de arquivo de entrada de dados: chave PIPE_MATERIAL. . . . .	61
Figura 44 – Exemplo de arquivo de entrada de dados: chave MODE_SELECTOR. . . . .	62
Figura 45 – Exemplo de arquivo de entrada de dados: chave CONTACT_PIPE_PLAN. . . . .	63
Figura 46 – Exemplo de arquivo de entrada de dados: chave STEPS_DEFAULTS. . . . .	64
Figura 47 – Exemplo de arquivo de entrada de dados: chave SUPPORTS. . . . .	65
Figura 48 – Exemplo de arquivo de entrada de dados: chave FATFREE. . . . .	66

## LISTA DE TABELAS

Tabela 1 – Sequência típica de carregamentos em uma análise de dutos no ABAQUS. . .	24
Tabela 2 – Comparação entre valores para a primeira frequência. . . . .	49

## LISTA DE ABREVIATURAS E SIGLAS

CFD	<i>Computer Fluid Dynamics</i>
CLI	<i>Command Line Interface</i>
GUI	<i>Graphical user interface</i>
MEF	Método dos Elementos Finitos
VBA	<i>Visual Basic for Applications</i>
VIV	Vibração Induzida por Vórtice
UML	<i>Unified Modeling Language</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>14</b>
<b>1.1</b>	<b>Fluxo de avaliação de vida à fadiga . . . . .</b>	<b>15</b>
<b>1.2</b>	<b>Motivação . . . . .</b>	<b>17</b>
<b>1.3</b>	<b>Objetivos . . . . .</b>	<b>19</b>
<b>1.4</b>	<b>Delimitação do trabalho . . . . .</b>	<b>19</b>
<b>1.5</b>	<b>Organização do Trabalho . . . . .</b>	<b>19</b>
<b>2</b>	<b>METODOLOGIA . . . . .</b>	<b>21</b>
<b>3</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>22</b>
<b>3.1</b>	<b>Modelagem numérica de dutos submarinos . . . . .</b>	<b>22</b>
<b>3.2</b>	<b>Vibração induzida por vórtices em vãos-livres . . . . .</b>	<b>28</b>
<b>4</b>	<b>DESENVOLVIMENTO DA FERRAMENTA . . . . .</b>	<b>36</b>
<b>4.1</b>	<b>Fluxo de avaliação de vida à fadiga com uso da ferramenta . . . . .</b>	<b>36</b>
<b>4.2</b>	<b>Implementação computacional . . . . .</b>	<b>37</b>
<b>5</b>	<b>APLICAÇÕES E RESULTADOS . . . . .</b>	<b>44</b>
<b>5.1</b>	<b>Exemplo: análise de vida à fadiga de um pequeno trecho de duto . . . . .</b>	<b>44</b>
<b>6</b>	<b>CONCLUSÕES . . . . .</b>	<b>51</b>
<b>6.1</b>	<b>Sugestões de trabalhos futuros . . . . .</b>	<b>51</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>53</b>
	<b>APÊNDICE . . . . .</b>	<b>55</b>
	<b>APÊNDICE A – DESCRIÇÃO DO FORMATO DO ARQUIVO DE EN- TRADA . . . . .</b>	<b>56</b>

## 1 INTRODUÇÃO

Ao longo das últimas décadas, à medida que novos campos de petróleo e gás foram descobertos em águas profundas e distantes da costa, surgiu a necessidade de utilização de sistemas submarinos de coleta e exportação utilizando dutos rígidos cada vez mais extensos. Com uma maior extensão, aumentou-se a ocorrência de seções de duto que não ficam apoiadas sob o leito marinho devido as irregularidades do solo. Essas seções dos dutos são denominadas trechos em vão-livre. Estas irregularidades podem ser preexistentes durante a instalação ou devido a subsequentes movimentos horizontais de *scouring*<sup>1</sup> de dutos durante a operação.

A presença de trechos dos dutos em vão-livre exige uma avaliação para determinação da necessidade de ações corretivas para evitar danos aos mesmos. Ainda na fase de projeto, uma avaliação do perfil do fundo do mar ao longo da rota proposta pode ser realizada para identificar se é esperado que haja trechos do duto em vão-livre. Na existência de tais trechos, é necessária uma análise numérica que forneça previsões dos números e tamanhos dos vãos esperados, que são indicadores da necessidade de possíveis alterações na rota ou ações corretivas.

Devido aos elevados custos ambientais, financeiros, e à imagem da empresa, associada a possíveis acidentes, o transporte seguro de hidrocarbonetos e outros fluidos nos oleodutos é uma das prioridades da indústria de petróleo e gás. A vibração livre é uma grande preocupação na análise de fadiga de componentes de dutos submarinos, incluindo dutos em vãos-livres (GAMINO; ABANKWA; PASCALI, 2013).

Neste contexto, o comportamento estático e dinâmico do duto deve ser investigado para garantir a segurança, combatendo o dano estrutural por fadiga, mantendo-o em um estado aceitavelmente seguro. Se as condições necessárias à segurança não puderem ser garantidas, as ações corretivas na forma de mudança de rota, correção de vãos, supressão do VIV e similares são usadas para garantir que os critérios de projeto relativos aos níveis de tensão e possíveis danos por fadiga devido ao VIV não sejam excedidos. Para a definição mais assertivas e precisas, uma modelagem estrutural pode ser utilizada, e o Método dos Elementos Finitos (MEF) é amplamente usado nessa tarefa. A configuração de dutos no fundo do mar depende das características topográficas do leito marinho, características do solo, tensão residual de lançamento, rigidez do duto e seu peso submerso.

Para que as condições de contorno e características do problema simulado reproduzam o comportamento real, é necessário modelar desde a etapa de instalação até a operação do duto, assim como considerar efeito de carregamentos dos diferentes valores de pressões internas e externas nas respectivas etapas. Modelar a instalação de dutos em um software de elementos finitos para uso geral pode ser um trabalho demorado e tedioso, principalmente devido a grandes quantidades de dados de batimetria. Na maioria das vezes, é necessário até o uso de *script* na

---

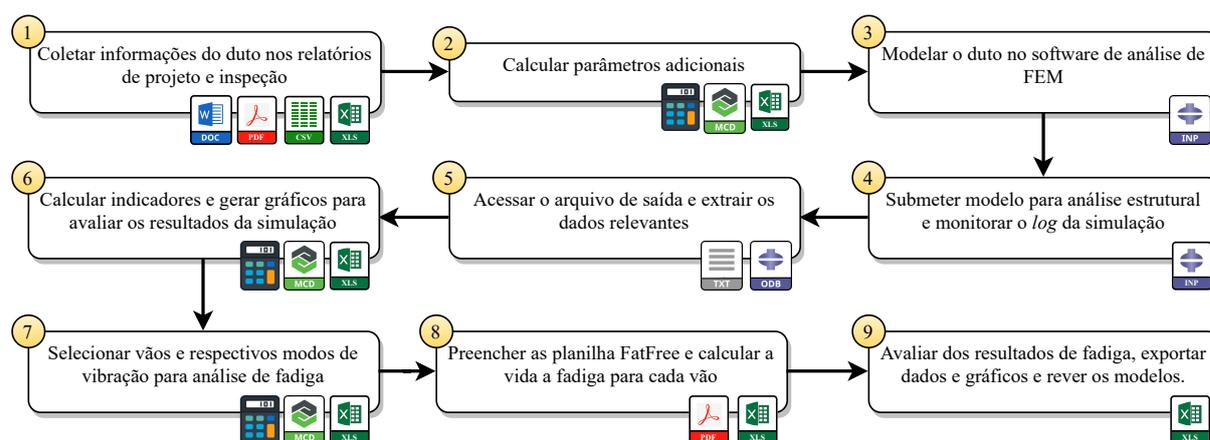
<sup>1</sup> retirada de solo que suporta o duto devido às intensas correntes de fundo.

definição do perfil do leito marinho, para poder simular o processo de assentamento (Van den Abeele; BOËL; HILL, 2013).

## 1.1 Fluxo de avaliação de vida à fadiga

O fluxo de trabalho de análise de vida à fadiga em dutos em vão-livre compreende uma série de tarefas que pode ser organizadas em um fluxograma como o apresentado na Figura 1.

Figura 1 – Fluxo de avaliação de vida à fadiga em dutos em vão-livre.



Fonte: Autor (2020)

A seguir, uma breve descrição de cada item:

- (1) Nesta etapa, o profissional reúne as informações básicas para construção dos modelos e outros dados usados em cálculos posteriores, por exemplo: as cotas do perfil do duto e batimetria obtidas na inspeção, geometria e propriedades dos materiais das camadas que compõem seção do duto, os parâmetros do solo, dados meteoceanográficos, os coeficientes de segurança e outras constantes físicas, posição e tipos de suportes<sup>2</sup> ao longo do duto; Essa tarefa envolve analisar uma série de documentos (.doc, .pdf, etc) em busca desses valores, dispostos de forma não estruturada. Quando estruturados, em forma de arquivos CSV ou planilhas, por exemplo, é necessário ainda manipular esses dados de modo a extrair somente a informação necessária e/ou convertê-las no formato apropriado. Um exemplo disso são os dados de batimetria, que precisam ser convertidos nas coordenadas dos nós de uma malha de elementos finitos no formato de arquivo de entrada do software de simulação. De posse desses dados, pode-se iniciar a fase de pré-processamento de dados da análise de elementos finitos.
- (2) Uma vez que nem todos os dados a serem utilizados estão de acordo com as especificações dos softwares a serem empregadas nas análises numéricas, ainda é necessários manipular

<sup>2</sup> restrições nodais ou apoios

alguns desses valores, seja calculando constantes ou convertendo unidades. Para isso, geralmente utiliza-se softwares de planilhas e/ou folhas de cálculos (Microsoft Excel, MathCad, Maple, etc.). Esta etapa corresponde ao início da primeira fase de pré-processamento de dados da análise de elementos finitos;

- (3) Com todos os dados em mãos, é necessário transformá-los em um modelo numérico no software de elementos finitos, via interação com mouse e teclado (GUI), ou criando arquivos de entrada. Embora a reutilização de arquivos de entrada previamente criados facilite essa tarefa, nem todos os trechos desses arquivos são suficientes ou podem ser reaproveitados. Essas limitações são frequentes em trechos do arquivo que precisam ser repetidos a depender da quantidade de certas entidades no modelo (suportes, por exemplo). Ao fim desta etapa, encerra-se o pré-processamento dos dados da análise de elementos finitos;
- (4) Por mais simples que seja submeter o modelo para análise na maioria dos softwares de simulação numérica por elementos finitos (alguns cliques via GUI ou um comando via CLI), as análises costumam levar horas e envolver execuções sucessivas a fim de realizar intervenções no modelo que não podem ser modeladas previamente. Dessa forma, torna-se necessário o monitoramento do progresso da simulação. Esta etapa compreende a primeira parte do processamento da análise de elementos finitos;
- (5) Uma vez concluída a simulação via elementos finitos, é necessário analisar os resultados antes do pós-processamento. Por vezes, é preciso extrair as informações que estão armazenadas em arquivos proprietários, utilizando para isto, as funcionalidades das ferramentas dos próprios pacotes de software de elementos finitos. Esta tarefa, geralmente feita via GUI, costuma ser repetitiva e pode levar de alguns minutos ou horas. Esta etapa inicia parte do pós-processamento da análise de elementos finitos;
- (6) De posse dos resultados, é necessário calcular (e muitas vezes visualizar em gráficos) alguns indicadores a fim de avaliar a qualidade dos resultados. Embora poderosos, estes softwares ainda carecem de gráficos mais interativos, como possibilidades de manipulações diretas importantes, tais como ampliação e translação dos mesmos através do uso do mouse. Esta etapa encerra o pós-processamento da análise de elementos finitos;
- (7) Na metodologia presente nas recomendações técnicas — abordada na subseção 3.2.2 — o cálculo de fadiga é baseada em modelos de resposta, portanto, é necessário calcular a resposta de cada modo para as várias condições de carregamento ambiental, o que a torna impraticável sem automação. Embora a FatFree realize estas operações, dentre as dezenas de modos obtidos por solução modal na análise de elementos finitos costumam aparecer modos espúrios. Dessa forma, antes de realizar a análise na FatFree, é necessário escolher dentre os modos de vibração obtidos na simulação numérica aqueles que mais contribuem

para fadiga. Esta tarefa costuma ser feita via inspeção visual, observando a forma dos modos, mas é uma prática pouco precisa e subjetiva;

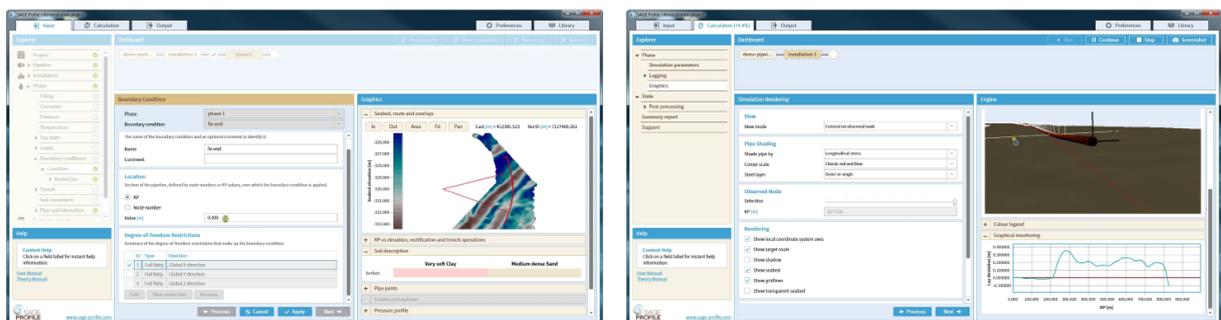
- (8) Uma vez selecionados os modos a serem usados para cálculo de fadiga, é necessário o pós-processamento dos dados de deslocamentos ou tensão adimensionalizados pelo diâmetro de cada modo, o que consiste em algumas centenas de valores. Além disso, é necessário manipular outras informações referentes a geometria e propriedades dos materiais da seção do duto, parâmetros do solo, coeficientes de segurança e condições de carregamento em várias páginas diferentes. Com todos esses dados, pode-se calcular os resultados de fadiga;
- (9) Finalmente, os dados de fadiga podem ser analisados e exportados para outras ferramentas a fim de gerar gráficos e relatórios.

## 1.2 Motivação

No cenário mundial existe a tendência da indústria de óleo e gás de investimento em transformação digital em todas as áreas da cadeia, com desenvolvimento de práticas e ferramentas. Esse movimento levou ao surgimento de ferramentas específicas ao auxílio do profissional responsável pela análise, visualização, predição dos resultados de VIV em dutos em vão-livre (MITTAL; SLAUGHTER; BANSAL, 2017). No entanto, pela especificidade dessas ferramentas, seu número ainda é reduzido, destacando-se apenas duas de nível comercial.

Desenvolvido pela empresa Fugro, que atua no monitoramento de dutos submarinos, o SAGE Profile (FUGRO, 2020) é um software disponível no mercado para análise global de dutos submarinos. Por ser específico para este uso, essa aplicação representa avanços em relação à modelagem com softwares de elementos finitos genéricos. No entanto, ela se limita a análise estrutural, deixando a análise de fadiga sob responsabilidade do usuário. Além disso, a interação com o usuário é limitada a interface gráfica (GUI<sup>3</sup>, como na Figura 2), o que dificulta a automação de tarefas corriqueiras.

Figura 2 – Interface gráfica do SAGE Profile.

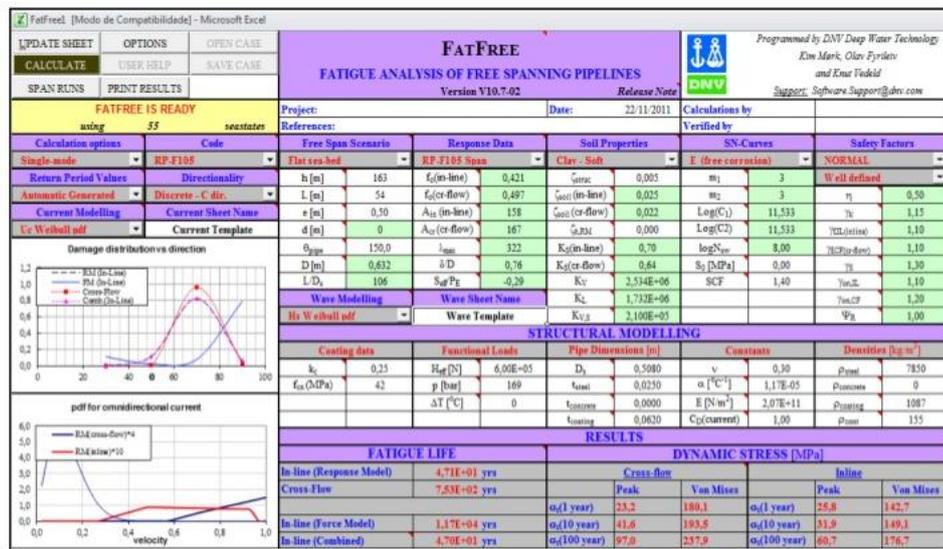


Fonte: [www.sage-profile.com](http://www.sage-profile.com).

<sup>3</sup> *Graphical User Interface*: interface de usuário gráfica

Em outra frente, tem-se a empresa DNG-GL, que é uma referência mundial, entre outras áreas, na análise de dutos em vão-livre. Essa empresa é responsável pelo desenvolvimento da suíte Sesam for pipelines (DNV-GL, 2020) focados na análise de dutos submarinos. Essa suíte consiste em 6 aplicações em VBA<sup>4</sup>, dentre as quais a mais destacada é a FatFree<sup>5</sup> (ver Figura 3), responsável pelo cálculo da vida à fadiga em si. Sendo desenvolvidas pela DNG-GL, as aplicações seguem as normas e recomendações práticas propostas pela mesma — que traz bastante confiabilidade nos resultados. Entretanto, apesar de conter uma aplicação para análise de comportamento mecânico, essas ferramentas são simples e independentes, e estão muito aquém de uma solução completa para simulação estrutural com interação solo-duto, como o SAGE Profile.

Figura 3 – Interface gráfica (planilha) da FatFree.



Fonte: Autor (2020)

Pelo exposto sobre o SAGE Profile e o Sesam for pipelines, nota-se que mesmo entre as melhores ferramentas atuais para analisar fadiga de duto em vão-livre ainda há espaço para melhorias, especialmente no que se refere à integração com outras ferramentas que auxiliam e outras etapas do processo. Diante dessas limitações, é comum usar um software para a análise de elementos finitos genérico (como ABAQUS (Dassault Systèmes Simulia, 2018) e ANSYS<sup>6</sup>) e realizar o cálculo de fadiga em folhas de cálculo pessoais ou comerciais, como a FatFree.

Neste cenário, são essenciais ferramentas que possam auxiliar não só no processamento (como os softwares de simulação), mas também o pré e pós-processamento de dados e, até mesmo, na automação de procedimentos. Uma ferramenta com essas características traz ganhos significativos para a produtividade e reduz a possibilidade de ocorrência de erros humanos. Além

<sup>4</sup> Uma linguagem pela qual se pode customizar e estender aplicações *desktop* da suíte Microsoft.

<sup>5</sup> Planilha Microsoft Office Excel desenvolvida pela DNVGL-RP-F105 focada no cálculo de vida à fadiga de dutos submarinos.

<sup>6</sup> <https://www.ansys.com>

de aumento de produtividade, e redução de falhas associadas a erro humano, tem as questões relacionadas a difícil mobilização e custos dessas operações. Esses itens, caso sejam executados de forma inadequada, podem até amplificar o problema. Adicionalmente, uma ferramenta que integre softwares de uso específico (para análise e visualização, por exemplo), pode reduzir atritos e padronizar o fluxo de trabalho, em comparação ao uso isolado destes softwares.

### 1.3 Objetivos

Este trabalho tem como objetivo geral desenvolver uma ferramenta para a análise de fadiga em dutos submarinos em vãos-livres, integrando um software de análise de elementos finitos (ABAQUS) e uma planilha de cálculo de vida à fadiga (FatFree) em um fluxo de trabalho.

Além disso, este trabalho tem como objetivos específicos:

- Contribuir para a metodologia de análise de fadiga em dutos por meio da criação de uma metodologia de seleção de modos de vibração;
- Modelar e implementar uma ferramenta utilizando o paradigma da programação orientada a objetos, através da linguagem *Python*;
- Apresentar a aplicação de algumas das funcionalidades da ferramenta em um estudo de caso.

### 1.4 Delimitação do trabalho

Este trabalho descreve o processo de desenvolvimento de uma ferramenta computacional para auxílio no processo de análise de vida à fadiga de dutos submarinos em vão-livre. A entrega de valor da ferramenta reside na automação de tarefas das etapas de pré e pós-processamento dos dados de entrada do ABAQUS e FatFree, bem como a possibilidade de execução integrada dessas ferramentas. Portanto, a ferramenta desenvolvida não pretende implementar o processo de análise numérica de elementos finitos, nem o cálculo de vida à fadiga, estes são responsabilidade de ferramentas específicas com esta finalidade, como, por exemplo, ABAQUS e FatFree.

### 1.5 Organização do Trabalho

Nesta seção são apresentados, de forma resumida, os assuntos que serão tratados com mais detalhes em cada capítulo do presente trabalho. No capítulo introdutório, apresenta-se o contexto no qual o problema está inserido, bem como algumas informações relevantes para reforçar a importância do tema em estudo. Além disso, são definidos alguns conceitos iniciais a fim de garantir uma melhor compreensão do que está sendo tratado.

O Capítulo 2 descreve sucintamente a sequência de passos seguidos para o desenvolvimento do trabalho aqui apresentado.

O Capítulo 3 apresenta os conceitos básicos para compressão do problema de determinação da vida à fadiga em dutos em vão-livre. Este capítulo está dividido em duas sessões. A primeira (seção 3.1) apresenta a modelagem computacional do comportamento de dutos submarinos, os passos de carga, e os tipos de elementos empregados para modelar os elementos necessários a simulação. Já a seção 3.2, apresenta os conceitos e as formulações por trás da Vibração Induzida por Vórtice (VIV), e as recomendações dados pela referência técnica DNVGL-RP-F105.

O Capítulo 4 é apresentado o fluxo os aspectos da implementação computacional da ferramenta, bem como os princípios norteadores de algumas escolhas, como a escolha da linguagem e paradigma de programação, a estrutura proposta para os módulos e classe.

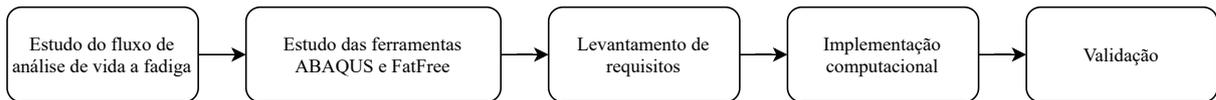
No Capítulo 5 tem-se a apresentação do uso da aplicação da ferramenta em uma análise de vida à fadiga para um caso particular. Neste capítulo é possível observar como se faz o uso das principais funções implementadas e a forma de apresentação dos resultados.

No Capítulo 6 são tratadas as considerações finais do trabalho, bem como são apresentadas algumas sugestões para trabalhos futuros.

## 2 METODOLOGIA

Para alcançar os objetivos propostos neste trabalho, a metodologia de desenvolvimento foi baseada em 5 etapas, conforme ilustra a Figura 4, a seguir:

Figura 4 – Fluxograma das etapas do trabalho.



Fonte: Autor (2021).

A primeira etapa consistiu em estudar o fluxo básico de trabalho de profissional, desde o acesso às informações até chegar nos resultados de fadiga. Essa etapa foi desenvolvida através de reuniões e oficinas ministrados por engenheiros da PETROBRAS que atuam na realização das análises de dutos em vão-livre (interessados diretos no desenvolvimento da ferramenta). Nessa etapa foram identificados as principais dificuldades do processo, destacando aqueles de maior potencial de automatização.

Em seguida, na segunda etapa, buscou-se conhecer os softwares que seriam integrados pela ferramenta: ABAQUS e FatFree. Com isso, pôde-se conhecer as oportunidades e limitações para integração de cada uma delas.

Na terceira etapa, fez-se a definição dos requisitos do sistema. Por exemplo: elaboração da especificação do fluxo dos dados na ferramenta, culminando na especificação de um arquivo de entrada para facilitar a estruturação dos dados para consumo pela aplicação. E assim chegou-se a um novo fluxo de trabalho que será apresentado mais adiante na seção 4.1.

Na quarta etapa, houve o desenvolvimento de uma série de *scripts* em linguagem Python para automação de algumas tarefas. Essa fase, de caráter exploratório, permitiu o desenvolvimento de pequenas ferramentas que podiam ter *feedback* mais rápido, melhorando o entendimento dos requisitos e compressão da visão geral do fluxo de trabalho. Com esse conjunto inicial de *scripts*, fez-se então a modelagem dos módulos da aplicação com base nas diferentes funcionalidades previstas para a ferramenta. Os códigos dos *scripts* foram, então, reagrupados nesses módulos, que se comunicam por meio da utilização das diversas classes e respectivos métodos.

Finalmente, na quinta e última etapa, fez-se a validação da ferramenta usando dados de dutos reais da PETROBRAS. No entanto, devido à confidencialidade destes dados, neste trabalho não serão apresentados os casos usados na validação, mas casos com dados fictícios para exemplificar as funcionalidades da ferramenta.

### 3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os dois principais tópicos cujo conhecimento é fundamental para compreender o estudo de vida à fadiga de duto submarinos em vão livre: a modelagem numérica com elementos finitos e a vibração induzida por vórtices.

#### 3.1 Modelagem numérica de dutos submarinos

A simulação numérica do duto projetado em um ambiente tridimensional realista obtido por medições da topografia do fundo marinho, permite que os engenheiros explorem quaisquer oportunidades que o comportamento do mesmo pode oferecer para desenvolver soluções seguras e econômicas. Por exemplo, o projetista pode analisar primeiro o comportamento do duto na batimetria original. Se alguns dos casos de carga resultam em tensões além do limite aceitável, pode-se simular uma alteração dos parâmetros ou até um intervenção do leito marinho (como colocação de suportes). A análise é executada novamente para confirmar que as modificações levaram à diminuição desejada de tensão ou deformação.

O modelo de elementos finitos pode ser uma forma para analisar o comportamento *in-situ* de um duto. Entende-se por comportamento *in-situ* de um duto, a resposta do mesmo às cargas ao longo de um histórico de carregamento (BAI; BAI, 2014). Isto pode consistir em vários casos de carga em sequência, como, por exemplo:

1. Instalação;
2. Testes de pressão (enchimento de água e do teste hidrostático);
3. Operação (enchimento com conteúdo, pressão de projeto e temperatura);
4. Ciclos de carga/descarga;
5. Flambagem lateral e vertical (*upheaval*);
6. Onda dinâmica e/ou de corrente<sup>1</sup>.

Bai e Bai (2014) apresentam o processo de análise do comportamento real desses dutos através do MEF, que será detalhado mais adiante.

##### 3.1.1 Análise estática

A modelagem da instalação do duto é o primeiro passo para o estudo do comportamento *in-situ* do duto. Visa reproduzir a configuração indeformada do duto assim que lançado sobre a leito marinho. Essa configuração é o ponto de partida para as etapas posteriores da análise.

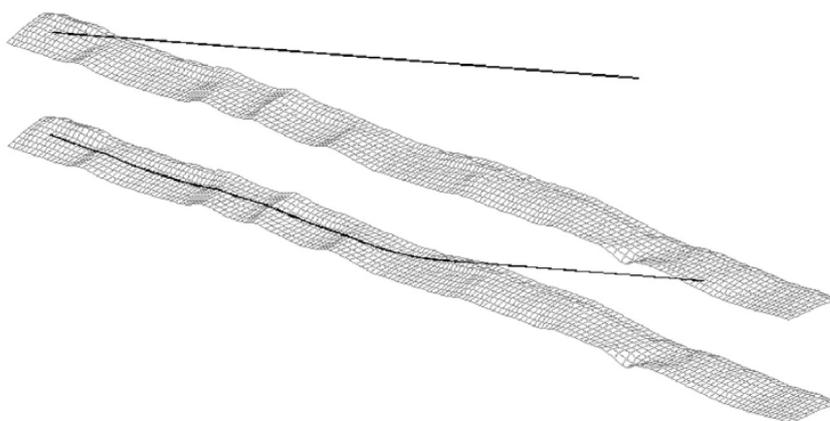
---

<sup>1</sup> Não incorporado a ferramenta.

Mais importante do que investigar o comportamento do duto durante a instalação é garantir que a correta representação da tração e ângulo de lançamento de tal modo que consigam gerar forças residuais no duto, oriundas do atrito quando o mesmo se apoia sobre a batimetria.

Por simplicidade, neste trabalho, assume-se o ângulo entre o duto e a horizontal nulo, isto é, o duto está em um plano horizontal que desce em direção a superfície batimétrica. Assim, o modelo permite especificar somente a tração de lançamento. Essa modelagem visa garantir a correta representação do contato entre o duto e a batimetria (forças de contato e ponto onde o duto toca o solo). A Figura 5 mostra o duto antes e durante o processo de instalação.

Figura 5 – Modelo de elementos finitos durante o lançamento.



Fonte: Bai e Bai (2014)

A medida que o duto se apoia, é necessário garantir um equilíbrio estável entre ele e o solo, o que é feito mediante um modelo representativo dessa interação, no qual deve-se definir o atrito e rigidez do leito marinho. No ABAQUS, pode-se relacionar a penetração e a força de reação do solo por meio de uma curva de rigidez axial, além de usar modelo anisotrópico para o atrito do solo, representando as diferenças entre os atritos nas direções longitudinal e transversal.

Após a descida do duto, têm-se os processos de alagamento e desalagamento, que acarreta mudanças no peso submerso do duto e, conseqüentemente, altera na sua configuração. Esses processos podem ser facilmente modelados por uma variação em uma carga vertical atuando no duto. Mas, um duto sujeito a essa variação de carga na condição alagada podem sofrer grandes deformações axiais devido à mudança em sua geometria, e assim o duto se deforma e assenta com outra configuração nos vãos-livres ao longo de sua rota.

Dessa maneira, é desejável que o modelo a ser estabelecido use um procedimento de análise que considere grandes deslocamentos e o efeito de alterações na área da seção do duto devido à alta tensão axial. Além disso, é interessante que o modelo do material seja capaz de representar o comportamento plástico da seção do duto.

A pressão hidrostática externa é um fator importante para a capacidade de resistência de duto em águas profundas. Como o modelo pode incluir uma estrutura tridimensional no fundo do

mar, a pressão externa pode ser uma função da profundidade da água. Já a pressão interna pode ser assumida como constante, mas a possibilidade de representar o efeito estático do conteúdo na extremidade pode ser incluído.

### 3.1.2 Procedimentos e etapas de carregamento na análise de elementos finitos

Um conceito base no ABAQUS é a divisão do histórico de carregamentos em etapas de carga. Para cada etapa, o usuário escolhe um método de análise. Dessa forma, é possível representar qualquer sequência e histórico de carregamento. Por exemplo, em um passo estático, o duto pode ser carregado com gás, no passo estático seguinte descarregado, e na terceira etapa, pode-se realizar uma análise exclusiva do duto vazio. Um histórico de carga de um modelo construído para uma análise estrutural é apresentado na Tabela 1.

Tabela 1 – Sequência típica de carregamentos em uma análise de dutos no ABAQUS.

Passo	Ação	Análise
1	Aplicação de peso próprio e empuxo do duto	Estática
2	Aplicação de pressão externa hidrostática	Estática
3	Aplicação da tração de lançamento;	Estática
4	Assentamento do duto no fundo do mar (ver Figura 5)	Estática
5	Remoção dos elementos de guincho	Estática
6	Modificando condições de contorno para a condição de instalação	Estática
7	Enchimento de água para condições alagadas	Estática
8	Aplicação de pressão do teste hidrostático.	Estática
9	Remoção da pressão do teste hidrostático	Estática
10	Enchimento de gás	Estática
11	Aplicação de pressão de operação	Estática
12	Aplicação de temperatura de operação para a condição operacional	Estática
13	Remoção de pressão e temperatura para condição de recarga	Estática
14	Aplicação de carga de onda e corrente	Dinâmica

Fonte: Bai e Bai (2014).

Após os passos referentes aos processos de assentamento do duto, é possível incluir um passo de perturbação linear para obtenção dos modos de vibração, que são essenciais na análise de fadiga.

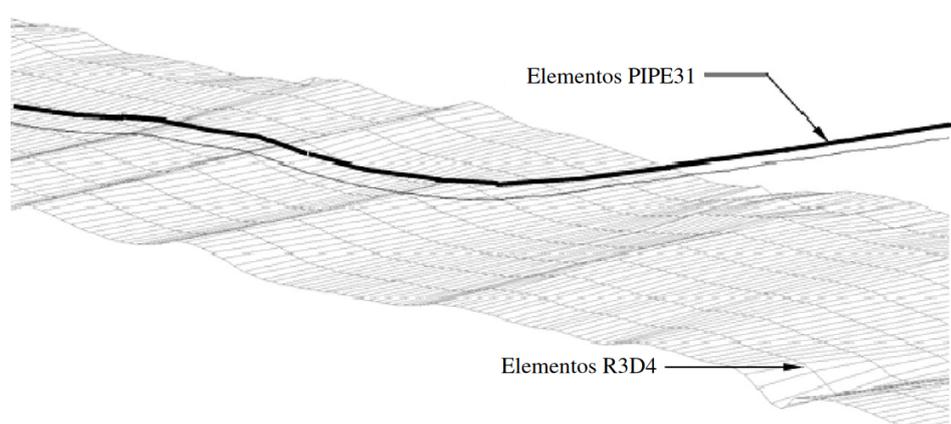
É válido de nota que a análise estática disponível no ABAQUS usada no modelo lida com respostas não lineares de efeitos de grandes deslocamentos, não-linearidade do material e não-linearidades de contorno, como contato, deslizamento e atrito (interação solo-duto). O ABAQUS usa o método de Newton para resolver as equações de equilíbrio não lineares. Portanto, a solução é obtida como uma série de incrementos com iterações para obter equilíbrio dentro de cada incremento (SIMULIA, 2018).

### 3.1.3 Tipos de elementos

O ABAQUS dispõe de alguns tipos de elementos a serem usados no modelo do sistema de solo-duto com elementos finitos, conforme a Figura 6:

- Para modelar o fundo do mar pode-se usar os elementos rígidos do tipo R3D4, ou superfícies analíticas rígidas;
- Os elementos do tipo PIPE31H usados para modelar o duto;
- Os elementos de molas usados para representar a continuidade do duto nas extremidades.

Figura 6 – Tipos de elementos usados no modelo.



Fonte: Bai e Bai (2014)

#### 3.1.3.1 Elementos para representação do duto

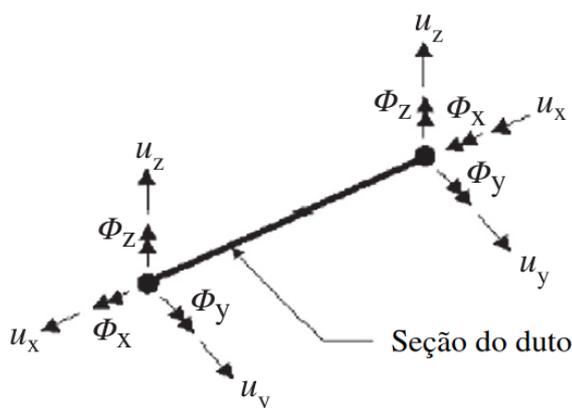
A Figura 7 mostra o elemento de duto finito 3D usado no modelo estabelecido, com 2 nós e 12 graus de liberdade. O elemento PIPE31 (padrão usado na ferramenta aqui apresentada) usa interpolação linear, e o elemento PIPE32 interpolação quadrática. A formulação híbrida torna o elemento adequado para casos com estruturas delgadas e problemas de contato, como um duto assentando sobre o fundo do mar.

Os elementos híbridos (PIPE31H/PIPE32H) são fornecidos para uso nos casos em que é numericamente difícil calcular as forças axiais e de cisalhamento pelo método de deslocamento próprio do Método dos Elementos Finitos. O problema nesses casos é que pequenas diferenças em posições nodais podem causar forças muito grandes em algumas partes do modelo, o que por sua vez, causar grandes deslocamentos em outras direções, a exemplo das deformações transversais provenientes de deformações axiais. Os elementos híbridos superam essa dificuldade usando uma formulação mais geral, onde as forças de cisalhamento axial e transversal nos elementos são incluídas, juntamente com os deslocamentos e rotações nodais, como variáveis

primárias. Embora essa formulação torne esses elementos mais custosos computacionalmente, eles geralmente convergem muito mais rapidamente quando as rotações dos dutos são grandes.

O elemento está disponível com uma seção circular vazada de paredes finas e suporta a possibilidade de o usuário especificar pressão externa ou interna. Elementos de paredes espessas também estão incluídos no ABAQUS. O elemento também pode considerar alterações na área da seção do duto devido ao alto nível de tensão axial do duto.

Figura 7 – Elemento PIPE31.



Fonte: Bai e Bai (2014)

### 3.1.3.2 Elementos para representação do fundo do mar

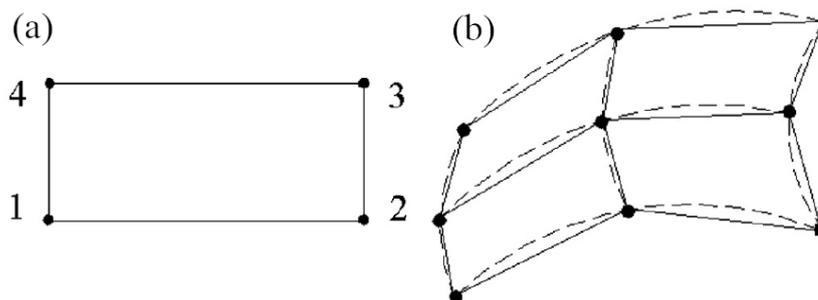
Para modelagem do leito marinho costuma-se utilizar elementos planos (R3D4) ou superfícies analíticas. A seguir, são apresentadas as características de cada uma dessas opções.

#### 3.1.3.2.1 Elemento R3D4

O elemento rígido R3D4 de quatro nós, como mostrado na Figura 8, possibilita modelar superfícies complexas com geometria arbitrária e é geralmente escolhido ao modelar a topografia do fundo do mar. Uma característica muito importante do ABAQUS ao modelar o fundo do mar tem sido a possibilidade de suavizar as superfícies geradas com os elementos rígidos, o que leva a uma representação muito melhor do fundo do mar do que a superfície facetada inicial.

A suavização é feita pela ABAQUS, criando superfícies de Bèzier com base na superfície facetada do fundo do mar, formada pelos elementos rígidos. As superfícies de Bèzier resultantes, diferentemente da superfície do elemento facetado, são lisas e têm uma superfície externa com direção normal contínua. As superfícies de Bèzier não correspondem exatamente à geometria facetada da superfície rígida, mas os nós dos elementos rígidos que definem o fundo do mar permanecem sempre na superfície de Bèzier. Além disso, o usuário pode especificar o grau de suavização para controlar a geometria da superfície suavizada.

Figura 8 – Elemento rígido R3D4 (a), e suavização da superfície de elementos (b).



Fonte: Bai e Bai (2014)

Um conjunto de elementos R3D4 que definem o fundo do mar é usado como a superfície principal chamada para aplicações de contato com os elementos do duto. Isso significa que um par de contatos (solo-duto) é definido e um modelo de interação é especificado. Esse modelo de interação geralmente consiste em uma definição de rigidez e atrito no fundo do mar.

### 3.1.3.2.2 Superfície analítica rígida

Outra forma de representar a batimetria do piso marinho é utilizar uma superfície analítica rígida, isto é, uma superfície geométrica com perfis que podem ser descritos com segmentos de linha reta ou curva, conforme Figura 9. Esses perfis podem ser varridos por um vetor gerador ou rotacionados em relação a um eixo para formar uma superfície tridimensional. Uma superfície analítica rígida está associada a um nó de referência de corpo rígido, cujo movimento governa toda a superfície. É importante destacar que este tipo de superfície possui apenas um lado disponível para contato, especificado de acordo com a orientação de eixos definida.

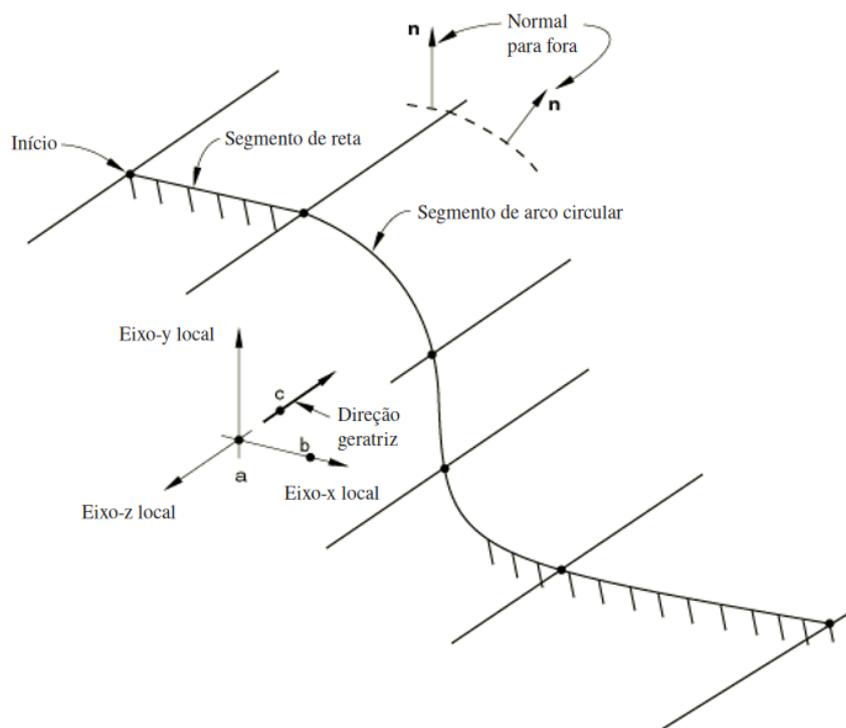
Em relação ao uso de superfícies formadas por elementos, o uso de superfícies analíticas rígidas apresenta duas vantagens:

- Superfícies geométricas curvas podem ser modeladas com precisão, uma vez que é possível parametrizá-las com segmentos de linhas curvas, o que tem como resultado uma superfície mais suave, fornecendo uma melhor aproximação à restrição de contato físico;
- Menor custo computacional decorrente do algoritmo de contato.

Por outro lado, como desvantagens do uso deste tipo de superfície:

- Uma superfície analítica rígida sempre age como superfície *master* em uma interação de contato, impossibilitando que se modele o contato entre duas superfícies rígidas analíticas;
- Forças de contato e pressões não podem ser plotadas em uma superfície rígida analítica, apenas na superfície *slave*;

Figura 9 – Exemplo de superfície analítica rígida.



Fonte: SIMULIA (2018)

- É necessário um número muito grande de segmentos, na ordem de milhares, para definir uma superfície rígida analítica pode diminuir o desempenho, sendo mais recomendável o uso de superfície baseada em elementos.

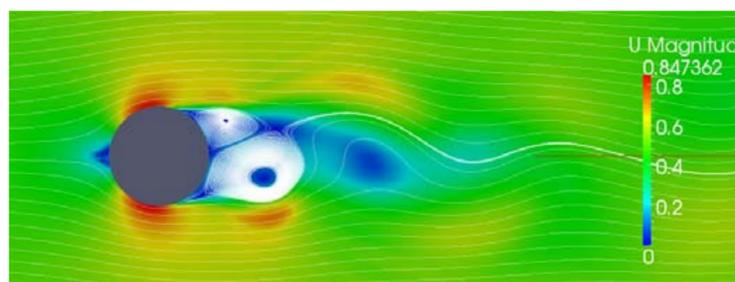
Para os casos estudados, utilizou-se uma superfície cilíndrica rígida tridimensional.

### 3.2 Vibração induzida por vórtices em vãos-livres

Quando um fluido de baixa viscosidade encontra um obstáculo, forma-se uma camada limite. Essa fina camada de fluido está sujeita aos efeitos das forças viscosas. Nesta camada, a velocidade do fluxo varia rapidamente, ficando cada vez mais lenta, formando um escoamento rotacional dentro da camada limite. Para determinadas velocidades de escoamento, a camada limite se desprende do obstáculo, formando uma esteira de vórtices, conhecida como esteira de von Kármán (CURRIE, 2012), conforme visto na Figura 10. Como consequência direta do desprendimento de vórtices, surge uma força oscilatória transversal ao fluxo, que age sobre o obstáculo, resultando em oscilações verticais e horizontais (NIELSEN; SØREIDE; KVARME, 2002).

A frequência do desprendimento de vórtices causado por um fluxo normal ao obstáculo (o duto em vão-livre, no caso em questão), é governado pelo número de Strouhal, diâmetro externo e velocidade de fluxo (MØRK et al., 2003). O número de Strouhal pode ser obtido pela

Figura 10 – Esteira de Von Kármán.



Fonte: Van den Abeele, Boël e Hill (2013)

expressão

$$S_t = (fL)/V \quad (3.1)$$

onde  $f$  é a frequência de vórtices,  $L$  é o comprimento característico e  $V$  é a velocidade do fluxo. Quando a velocidade do fluxo alcança uma das frequências naturais da estrutura, ela começa a vibrar e estas duas vibrações se correlacionam, causando vibrações de grande amplitude e grande dano (*lock-in*) (MØRK et al., 2003).

Como os dutos são geralmente modelados como cilindros, é importante entender como funciona o comportamento do fluxo de fluido ao redor dessa estrutura. Segundo Sumer e Fredsoe (1995 apud BATCHELOR, 1967), ao estudar vibrações de cilindros em corrente constante, inicia-se o desprendimento de vórtices quando o número de Reynolds,

$$Re = (UD)/\nu \quad (3.2)$$

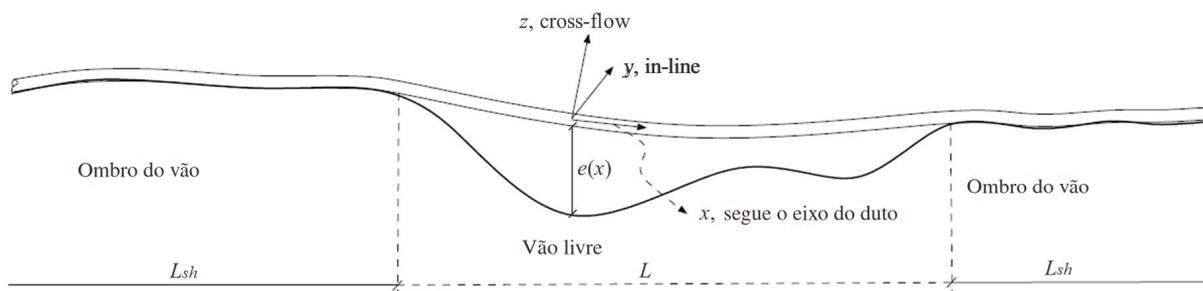
é maior que 40, onde  $U$  é a velocidade do fluxo,  $D$  é o diâmetro do cilindro e  $\nu$  é a viscosidade cinemática (SUMER; FREDSOE, 1995).

O desprendimento de vórtices induz uma variação cíclica de forças no cilindro. Assim, enquanto uma força de sustentação (*lift force*) oscila à mesma frequência do desprendimento de vórtices, a força de arrasto (*drag force*) oscila à duas vezes esta mesma frequência (SUMER; FREDSOE, 1995). Essas forças oscilatórias, os vórtices, podem induzir vibrações na direção ortogonal ao fluxo, *cross-flow* (CF), e na direção do fluxo, *in-line* (IL), denominadas vibrações induzidas por vórtices (VIV).

Com o duto exposto às ondas e correntes, a que encontra-se em vão livre estará suscetível à VIV (Figura 11). Caso a frequência de desprendimento alcance uma das frequências naturais do duto, esse poderá entrar em ressonância. As excitações dinâmicas podem causar danos por fadiga, sendo importante identificar os corretos procedimentos de intervenção, seja no duto ou no leito marinho.

A DNVGL-RP-F105 é uma referência técnica adotada mundialmente no estudo de dutos em vão-livre. Nele está presente uma metodologia baseada em modelos de resposta a fim de avaliar a fadiga causada por VIV. Estes modelos representam relações empíricas

Figura 11 – Duto em Vão-livre e direções das oscilações.



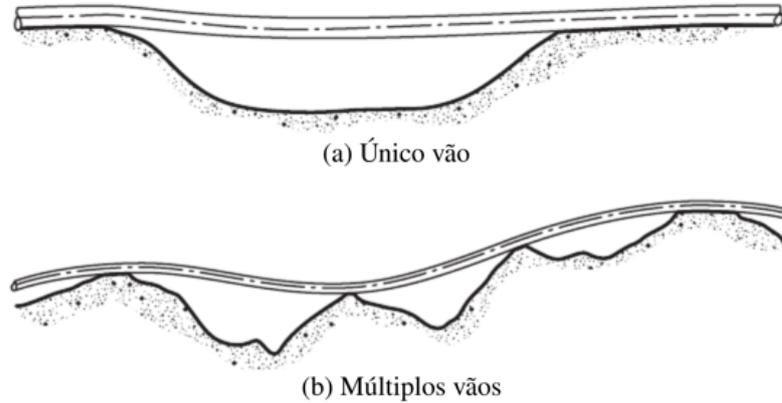
Fonte: Adaptado de Det Norske Veritas (2017)

entre a velocidade reduzida (Equação 3.4) e a amplitude de resposta adimensional, utilizadas para prever as amplitudes de vibração nas direções *in-line* e *cross-flow* (MØRK et al., 2003; Det Norske Veritas, 2017). Além desta, a recomendação prática sugere também um método baseado no coeficiente de sustentação e nas curvas do coeficiente de massa adicional como função da amplitude de resposta adimensional e da frequência de vibração adimensional (Det Norske Veritas, 2017). Como terceira opção, a DNVGL-RP-F105 indica o uso de fluidodinâmica computacional (CFD, na sigla em inglês) para escoamento turbulento ao redor dos dutos para avaliação do VIV.

A DNVGL-RP-F105 considera dois modelos para estimar a resposta dinâmica em um vão-livre: Modelo de Resposta (*Response Model*–RM) e Modelo de Força (*Force Model*–FM). A escolha do modelo, segundo Tura et al. (1994), depende: (i) do comportamento dos carregamentos ambientais, isto é, quando há ressonância induzida por vórtice, aplica-se RM; e quando o comportamento do vão-livre é afetado por carregamentos periódicos com pouca ou nenhuma amplificação dinâmica, aplica-se FM; (ii) da direção e tipo de fluxo, RM é aplicável na direção *in-line* para corrente contínua e na direção *cross-flow* para qualquer padrão de fluxo; o FM é aplicado na direção *in-line* para carregamentos de onda direto.

A DNVGL-RP-F105 pode ser aplicada para vãos únicos e múltiplos onde um modo de vibração é predominante (unimodal), conforme a Figura 12. Porém, a combinação de vãos de grande extensão e altas correntes, ou ainda vãos múltiplos, faz com que não apenas os modos fundamentais sejam ativados, mas também diversos outros modos de ordem mais alta (multimodal).

Figura 12 – Configurações típicas para vãos.



Fonte: Adaptado de Bai e Bai (2014)

### 3.2.1 Modelos de respostas para VIV

Os modelos de resposta relacionam a velocidade do fluxo do fluido externo e a amplitude de vibração do duto em casa direção (*in-line* e *cross-flow*). Estes modelos podem ser construído através do conjunto de equações que dependem principalmente da velocidade reduzida  $V_R$  e do ângulo do fluxo  $\theta_{rel}$ .

Segundo a DNVGL-RP-F105, o fluxo pode ser dividido em duas zonas: (i) uma zona exterior, distante do solo marinho, onde velocidade de corrente média e a turbulência variam muito pouco na direção horizontal; e (ii) uma zona interior, onde a velocidade de corrente média e a turbulência têm variações consideráveis na direção horizontal. Uma vez que as medições da corrente são realizadas na zona exterior, fora da camada limite, a velocidade de corrente  $U_c$  incidente no duto pode ser aproximada a partir da equação

$$U_c = R_c U(z_r) \frac{\ln(e + D/2) - \ln(z_0)}{\ln(z_r) - \ln(z_0)}, \quad (3.3)$$

em que  $U(z_r)$  é a velocidade da corrente na altura  $z_r$  em relação ao leito marinho,  $e$  é distância o duto e solo (*gap*),  $z_0$  é o parâmetro de rugosidade do leito marinho (conforme a tabela 3-1 da Det Norske Veritas (2017)), e  $D$  é o diâmetro externo do duto.

Uma vez encontrada a velocidade da corrente na zona interior, isto é, próxima do solo, a velocidade reduzida  $V_R$  pode ser calculada por

$$V_R = \frac{U_c + U_w}{f_n D}, \quad (3.4)$$

onde  $U_w$  é a velocidade de fluxo induzida por onda e  $f_n$  é a frequência natural de amplitude.

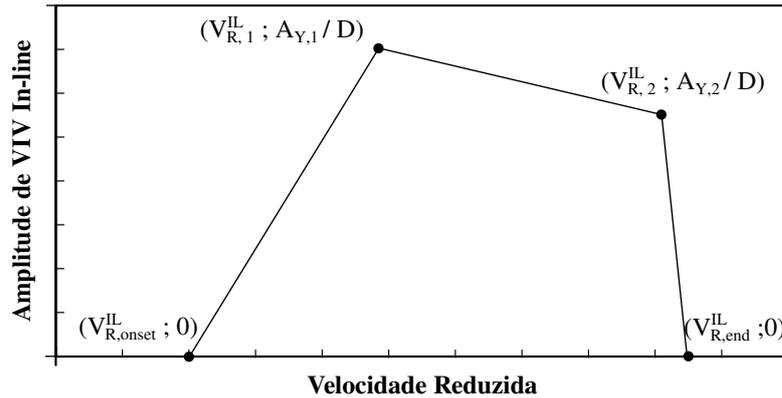
O parâmetro de estabilidade  $K_S$ , representa o amortecimento para uma dada forma modal, sendo obtido a partir da equação

$$K_S = \frac{4\pi m_e \zeta_T}{\rho_w D^2}, \quad (3.5)$$

em que  $\rho_w$  é a densidade da água e  $\zeta_T$  é a taxa de amortecimento modal total.

Com esses valores calculados, e pode-se construir a curva que relaciona velocidade reduzida e amplitude de vibração *in-line* adimensionalizada pelo diâmetro, semelhante a Figura 13, onde as coordenadas dos pontos são encontradas nas equações do item 4.6.4 da DNVGL-RP-F105.

Figura 13 – Curva de modelo de resposta *in-line*.



Fonte: Adaptado de Det Norske Veritas (2017)

De acordo com a DNVGL-RP-F105, a resposta de amplitude de um duto vibrando na direção *in-line*, contempla regiões com velocidade de corrente entre 1,0 e 4,5. Tem-se que a resposta na direção longitudinal depende dos parâmetros de velocidade de corrente, estabilidade, intensidade de turbulência e do ângulo entre a corrente e o duto.

Para o modelo de resposta *cross-flow* também é necessário calcular alguns outros parâmetros. Dessa vez, inicia-se com o cálculo do fator de correção para considerar a proximidade do duto com o solo

$$\Psi_{proxi,onset} = \begin{cases} \frac{1}{5} (4 + 1,25 \frac{e}{D}) & \text{para } \frac{e}{D} < 0,8 \\ 1 & \text{caso contrário} \end{cases} \quad (3.6)$$

Caso o duto esteja localizado próximo ou em trincheiras é necessário considerar o fator de correção específico

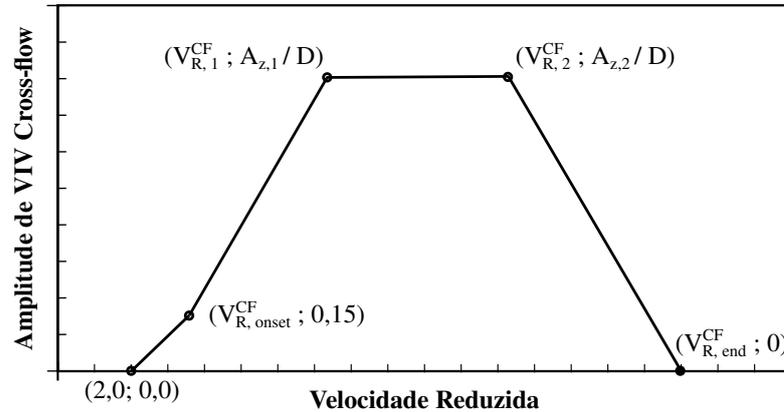
$$\Psi_{trench,onset} = 1 + 0,5 \frac{\Delta}{D} \quad (3.7)$$

onde  $\Delta$  é a profundidade da trincheira.

Com esses valores calculados, pode-se construir a curva que relaciona velocidade reduzida e amplitude de vibração adimensionalizada pelo diâmetro, semelhante a Figura 14, onde as coordenadas dos pontos são encontradas nas equações do item 4.6.4 da DNVGL-RP-F105.

### 3.2.2 Resposta multimodal

A resposta do vão-livre pode ser dada em função de uma coordenada  $x$  ao longo da direção longitudinal do duto. Para cada combinação relevante de estado de mar e velocidade de

Figura 14 – Curva de modelo de resposta *cross-flow*.

Fonte: Adaptado de Det Norske Veritas (2017)

corrente, um número de modos pode ser excitado simultaneamente na mesma direção, dando origem a uma resposta multimodal. Todavia, o número de modos que responderão e o quanto cada modo contribuirá para o dano por fadiga dependerá da velocidade do fluxo, da posição no eixo  $x$  e da competição com outros modos.

A DNVGL-RP-F105 define três diferentes tipos de modos:

**Modos ativos** são aqueles que podem ser excitados por VIV. Com base no item 2.3.3 da DNVGL-RP-F105, os critérios de definição para que um modo *in-line* com frequência  $f_{IL,j}$ , ou *cross-flow* com frequência  $f_{CF,j}$ , seja considerado ativo é:

$$\begin{aligned} f_{IL,j} &\leq \frac{U_{extreme} \gamma_{f,IL}}{V_{R,onset}^{IL} D} \\ f_{CF,j} &\leq \frac{U_{extreme} \gamma_{f,CF}}{2D} \end{aligned} \quad (3.8)$$

sendo  $\gamma_{f,IL}$  e  $\gamma_{f,CF}$  coeficientes de segurança, variando de 1 a 1,3 a depender da classe de segurança e nível de definição do vão-livre (item 2.7.2 da DNVGL-RP-F105). Já  $U_{extreme}$  é a velocidade do fluxo perpendicular ao duto que considera velocidades de corrente e onda, e pode considerar combinações de períodos de retorno de 1, 10 e 100 anos. Um modo que não é passível de ativação pode ser desconsiderado nas análises em todos os pontos e velocidades de fluxo;

**Modos participantes** são modos ativos com amplitude de tensão relevante em um, ou ambos os lados, de um ponto na coordenada  $x$  ao longo do duto. Para que um modo  $j$  seja considerado participante no vão, é necessário que a seguinte condição (presente no item 4.3.3) seja atendida:

$$|A_{IL/CF,j}(x)| \geq \frac{A_{IL/CF}^{\max}}{10} \text{ para algum } x \in (x_{\text{start},j}, x_{\text{end},j})$$

sendo

$$A_{IL/CF,j}(x) = (1 + CSF)D\kappa_j(x)Er$$

onde

- $CSF$  fator de rigidez do concreto
- $\kappa_j(x)$  curvatura do modo na posição  $x$
- $E$  módulo de elasticidade
- $r$  coordenada radial da seção transversal do duto
- $(x_{start,j}, x_{end,j})$  intervalo de influência do modo

**Modos contribuintes** são modos participantes que deve satisfizer um dos seguintes critérios:

- direção *cross-flow*:  $(A_Z/D)_j \geq 0,1(A_Z/D)_{max}$ ;
- direção *in-line*:  $S_{IL,j}^P(x) \geq 0,1S_{IL}^{max}(x)$ .

onde  $(A_Z/D)_j$  é a amplitude VIV normalizada para o j-ésimo modo,  $(A_Z/D)_{max}$  é a amplitude VIV normalizada para o modo *cross-flow* dominante,  $S_{IL,j}^P(x)$  é a amplitude de tensões de reposta preliminar para o j-ésimo modo *in-line* e  $S_{IL}^{max}(x)$  é a amplitude de tensões de resposta associadas ao modo *in-line* dominante.

Baseado nos modelos de resposta unimodal, pode-se calcular as amplitudes do VIV para todos os modos. Assim, é necessário calcular VIV *cross-flow* e *in-line* para cada velocidade de corrente, estado de mar e em cada ponto com se os seguintes procedimentos:

- VIV *cross-flow*
  1. Identifica-se todos os modos participantes (*single* ou *multi location*)
  2. Com o modelo de resposta *cross-flow*:
    - a) Calcula-se a amplitude VIV normalizada para cada modo  $(A_Z/D)_j$ ;
    - b) Identifica-se o modo dominante, isto é,  $(A_Z/D)_{max}$ ;
    - c) Identificam-se os modos fracos  $0,1(A_Z/D)_{max} \leq (A_Z/D)_j \leq (A_Z/D)_{max}$ ;
    - d) Desconsidera-se os modos irrelevantes:  $(A_Z/D)_j < 0,1(A_Z/D)_{max}$ .
  3. Usando o modelo de resposta para baixos valores de Keulegan-Carpenter (*low Keulegan Carpenter flow regime*–LKCR), calcula-se  $(A_Z/D)_j$  para cada modo;
  4. Determina-se a resposta de tensão combinada:

$$S_{comb,CF} = \max(S_{comb,CF}^{RM}, S_{comb,CF}^{LKCR});$$

5. Determina-se a frequência de contagem de ciclos.

- VIV *in-line*

1. Identifica-se todos os modos participantes (*single* ou *multi location*);
2. Com o modelo de resposta *in-line*:
  - a) Calcula-se a amplitude VIV normalizada para cada modo  $(A_Y/D)_j$ ;
  - b) Identifica-se o modo dominante, isto é, o modo com  $S_{IL}^{\max}(x)$ ;
  - c) Identificam-se potenciais modos fracos:  $0, 1S_{IL}^{\max}(x) \leq S_{IL,j}^P(x) \leq S_{IL}^{\max}(x)$ ;
  - d) Desconsideram-se os modos irrelevantes:  $S_{IL,j}^P(x) < 0, 1S_{IL}^{\max}(x)$ .
3. Reduzir os modos fracos. Para VIV *in-line*, dois modos adjacentes podem competir se suas frequências forem próximas, ou agir de forma independente se estiverem distantes. A DNVGL-RP-F105 define que os modos competem se a razão entre as frequências é menor que 2, isto é,  $\frac{f_{n+1}}{f_n} < 2$ . Em modos que estão competindo, considera-se que apenas o “vencedor” da competição (maior valor de  $S_{IL}^P(x)$ ) pode ter máxima amplificação, enquanto a amplificação do modo “perdedor” é reduzida à metade. É interessante ressaltar que modos que não competem não têm redução;
4. Calcular o intervalo de tensões *in-line* excitados pelo modo *cross-flow* dominante  $S_{cross-flow-IL}(x)$ .

Para cada ponto e cada modo, calcula-se o intervalo de tensões induzido por VIV *in-line* para os modos contribuintes:

$$S_{IL,j}^{RM}(x) = S_{IL,j}^P \cdot 0,5^{\beta_j(x)};$$

Assume-se que apenas o modo *cross-flow* dominante é capaz de contribuir para o movimento *in-line* induzido pelo modo transversal. Desta forma, o modo *in-line* participante cuja frequência natural é próxima a duas vezes a resposta *cross-flow* dominante é escolhido como candidato a VIV *in-line* induzido por *cross-flow*. A amplitude de tensões *in-line* excitados pelo modo *cross-flow* dominante é dado por:

$$S_{CF-IL}(x) = 0,8 \cdot A_{IL,k}(x) \cdot \left(\frac{A_z}{D}\right)_{\max} \cdot R_k \cdot \gamma_s;$$

5. Escolher o maior  $S_{IL}^{RM}(x)$  e  $S_{CF-IL}(x)$  para cada modo;
6. Determinar a faixa de resposta de tensão combinada e a frequência de contagem de ciclos.

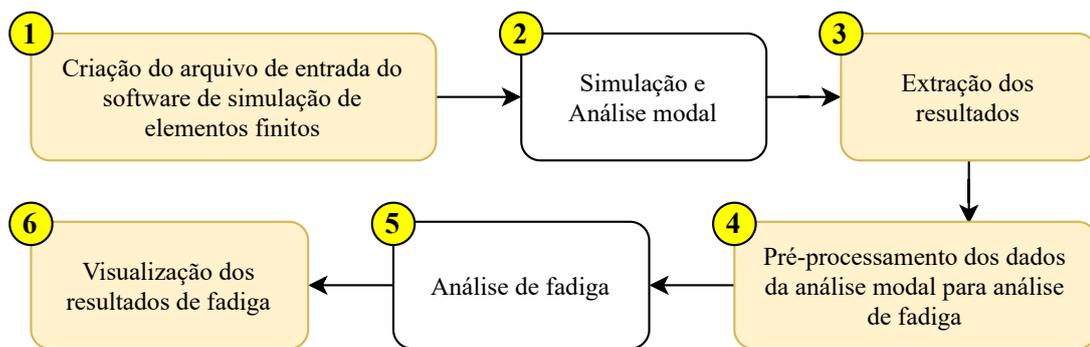
## 4 DESENVOLVIMENTO DA FERRAMENTA

Neste capítulo apresenta-se o fluxo de trabalho para análise de fadiga de um duto em vão livre com o uso da ferramenta desenvolvida e, em seguida, apresenta-se os detalhes da sua implementação.

### 4.1 Fluxo de avaliação de vida à fadiga com uso da ferramenta

Baseada no fluxograma apresentado na Figura 1, a ferramenta proposta tem como requisito atender o fluxo apresentado na seção 1.1, automatizando certas etapas da análise de vida à fadiga. Os itens 1, 3, 4 e 6 são etapas cobertas pela implementação ferramenta, e os itens 2 e 5 são realizados em aplicações externas.

Figura 15 – Fluxo de operação proposto para a ferramenta.



De forma mais detalhada, a ferramenta deve:

- (1) A partir de um arquivo de entrada com informações do modelo, criar arquivos de entrada (.inp, no caso do ABAQUS) que reproduza todo o processo de simulação do comportamento do duto apresentado (seção 3.1);
- (2) Submeter o arquivo gerado para análise. No caso de haver um passo de carga para a colocação de suportes, a simulação é executada em duas partes:
  - a) a primeira com os passos de carga anteriores a colocação dos suportes, e
  - b) a segunda com o passo da colocação dos suportes e os passos de carga seguintes.
- (3) Processar os arquivos de saída (.odb, no caso do ABAQUS) extraindo as informações relevantes como a configuração deformada, modos de vibração, etc., gerando arquivos em outros formatos de fácil leitura para pós-processamento, tanto por este quanto por outros softwares;
- (4) Pós-processar as informações gerando gráficos e relatórios relevantes para as tomadas de decisão do usuário. Esse é o requisito mais crítico, uma vez que é fundamental o

entendimento sobre a análise de duto em vão-livre. Entre as tarefas que fazem parte deste item está a automação da escolha dos modos de vibração ativos e relevantes e para cada vão de interesse (a qual deve ser norteadada pelos aspectos discutidos na subseção 3.2.2) e a manipulação da FatFree;

- (5) Iniciar o processo de cálculo de fadiga. No caso da FatFree, esse cálculo é realizado por uma rotina VBA ativada por um botão no no arquivo da planilha;
- (6) Capturar os resultados no arquivo que contém os resultados do cálculo de vida à fadiga, e apresentá-los na forma de gráficos e relatórios.

## 4.2 Implementação computacional

A seguir são apresentados os mais aspectos mais computacionais relacionados a implementação da ferramenta, como a escolha da linguagem de programação e a estrutura de do código em termos de módulos e classes.

### 4.2.1 Linguagem de programação

Python<sup>1</sup> foi a linguagem de programação adotada para implementação da ferramenta. Além de ser uma linguagem interpretada de alto nível Orientada a Objeto — que permite um alto índice de reaproveitamento de código — e da sintaxe simples. Rao (2018) apresenta algumas das principais vantagens que destaca a linguagem para este tipo de aplicação:

- Disponibilidade de bibliotecas para aplicações científicas contemplando manipulação de matrizes (Numpy), funções matemáticas (SciPy), manipulação de dados em forma tabular (Pandas), criação de gráficos interativos (Matplotlib e Bokeh);
- Suporte para automação de tarefas. Os recursos de *script* internos do Python e vários pacotes têm um forte suporte à automação de tarefas. A automação de tarefas repetitivas e a realização do registro de dados são fáceis e requerem pouco esforço. O ABAQUS, por exemplo, permite modelagem e acesso a informações em arquivos de saída via Python. A biblioteca xlwings permite manipulação de planilhas Excel, a exemplo da FatFree;
- Pacotes Python como Django e Flask tornam possível desenvolver e usar o Python para criar uma API<sup>2</sup> com um *front-end* da web. Essa funcionalidade é particularmente útil para reaproveitamento da ferramenta em outras aplicações.

<sup>1</sup> <https://www.python.org>

<sup>2</sup> Na programação de computadores, uma Interface de Programação de Aplicativos (*Application Programming Interface* — API) é um conjunto de definições de sub-rotinas e ferramentas para a criação de software. Em termos gerais, é um conjunto de métodos de comunicação claramente definidos entre vários componentes.

## 4.2.2 Pacotes e classes

Para implementação do fluxo de trabalho proposto para a ferramenta, fez-se a implementação de módulos para lidar com cada contexto específico. Em Python, módulos podem ser quaisquer arquivos com extensão `.py`. Estes módulos podem ser agrupados em pacotes, que são pastas que, além dos módulos, contém um arquivo `__init__.py`. Na ferramenta, têm-se alguns pacotes que agrupam módulos em torno do contexto de um problema que o *software* resolve.

### 4.2.2.1 Pacote `analysis`

É o pacote principal responsável por orquestrar o fluxo de trabalho da ferramenta desde o processamento dos dados de entrada, geração dos arquivos para o ABAQUS e os pós-processamentos. As funções de pré-processamento de dados estão aqui. Este pacote tem dois módulos (`models.py` e `inp.py`) que contém duas classes principais, respectivamente:

- `Model`: classe que contém as informações do modelo do problema. A classe armazena todas as informações para construção dos arquivos `.inp`, isto é, dados de batimetria, material, geometria do duto, coeficientes de segurança, entre outros. A principal forma de criação da instâncias dessa classe é pelo método estático `load_json`, que recebe um arquivo principal de entrada (em formato JSON, ver Apêndice A), e realiza o pré-processamento dos dados contidos nele;
- `Inp`: lida com a escrita modularizada de arquivos de entrada para o ABAQUS. Tem-se um arquivo `.inp` principal que conterá informações para a localização de outros arquivos acessórios que, por sua vez, terão as informações específicas de cada aspecto da modelagem: batimetria, passos de carga, etc. Isso facilita o reaproveitamento dos arquivos acessórios, sem a necessidade de alterações no arquivo principal.

Na Figura 16 vê-se um diagrama UML<sup>3</sup> com uma visão geral do pacote `analysis`.

### 4.2.2.2 Pacote `odb_handler`

Neste pacote está o módulo responsável por lidar com os arquivos de saída do ABAQUS (`odb`) no sentido de acessar e guardar os dados relevantes em arquivos com formatos de fácil manipulação por outros softwares (CSV e JSON, por exemplo, que possui módulos para leitura e escrita nativos em Python).

Na Figura 17 vê-se um diagrama UML com uma visão geral do pacote `odb_handler`.

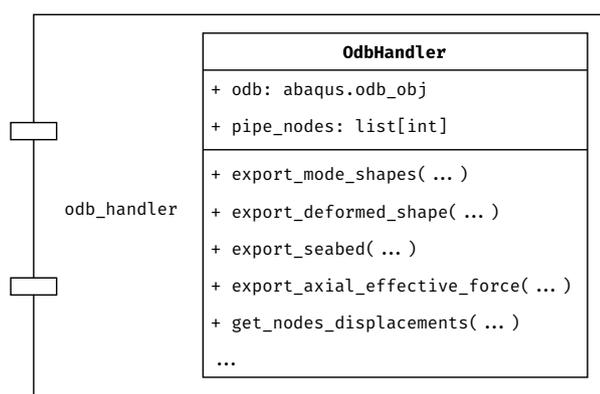
<sup>3</sup> Sigla para *Unified Modeling Language*: Linguagem Unificada de Modelagem é uma linguagem padrão para modelagem orientada a objetos (MARTINEZ, 2020)

Figura 16 – Digrama UML do pacote analysis.



Fonte: Autor (2020)

Figura 17 – Digrama UML do pacote odb\_handler.



Fonte: Autor (2020)

#### 4.2.2.3 Pacote mode\_selector

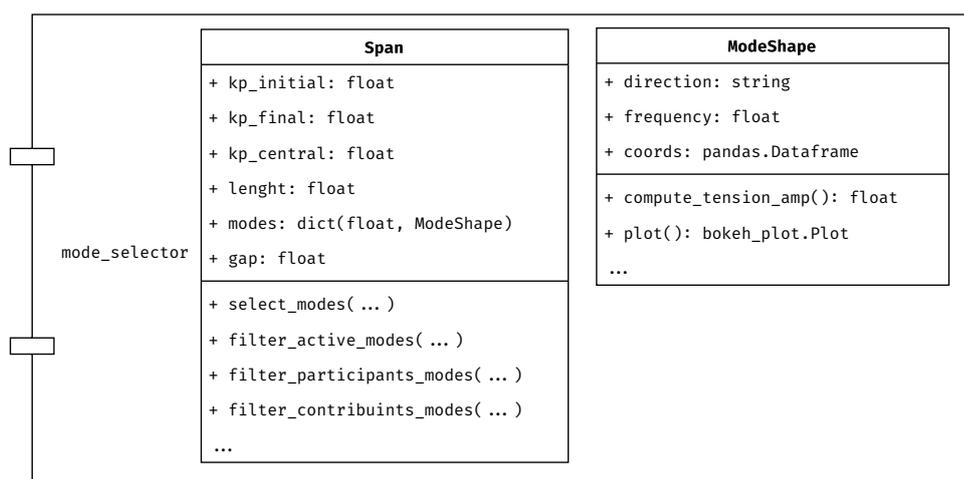
Neste pacote estão implementados os métodos responsáveis pela estratégia de seleção automática de modos de vibração (que depende da definição dos vãos), bem como a manipulação dos dados associados aos vãos e seus respectivos modos. As abstrações dos conceitos de vão e modo de vibração estão implementadas nas seguintes classes:

- *Span*: classe que representa um vão do duto. Uma vez que análise de fadiga é feita por vão, é nessa classe que são implementados os métodos responsáveis pela seleção dos modos de vibração, que são ligados à classe por um dos seus atributos;
- *ModeShape*: classe que representa um modo de vibração (*mode shape*). A principal forma de criação de objetos dessa classe é por meio da função `load_json` do mó-

dulo `mode_shapes.py`, que carrega os dados do arquivo gerado com utilização do `odb_handler`, e retorna uma lista de objetos desta classe.

Um diagrama UML com uma visão geral do pacote `mode_selector` é apresentado na Figura 18.

Figura 18 – Digrama UML do pacote `mode_selector`.



Fonte: Autor (2020)

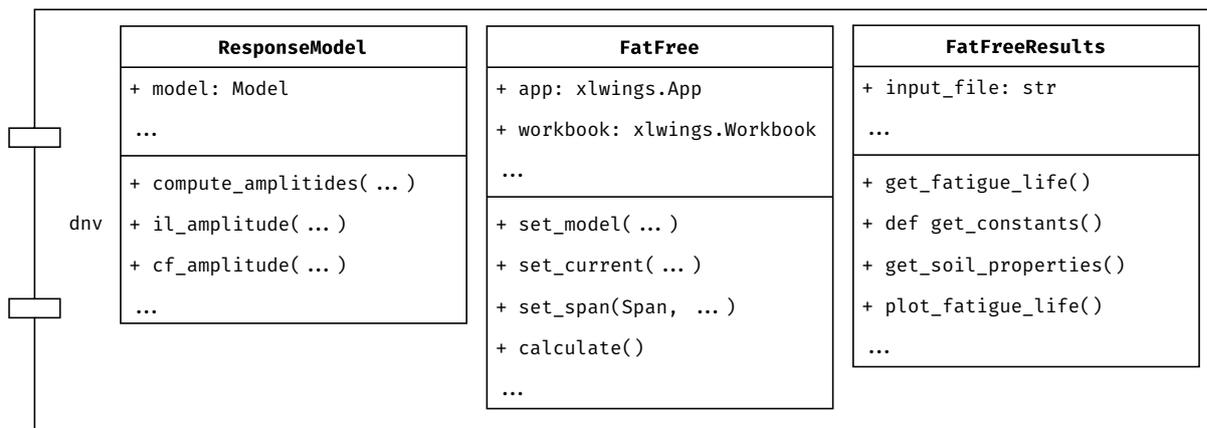
#### 4.2.2.4 Pacote `dnv`

Neste pacote são agrupadas módulos referentes à DNVGL-RP-F105, como cálculos dos modelos de resposta da DNVGL-RP-F105, que alimentam o algoritmo de seleção de modos de vibração, e entrada de saída de dados da planilha FatFree. As principais classes contidas nesses pacotes são:

- `ResponseModel`: implementa a formulação dos modelos de resposta apresentados na seção 3.2;
- `FatFree`: responsável por fazer a entrada dos dados na planilha e invocar a execução dos cálculos. Essa classe faz uso da biblioteca `Xlwings`, que consegue se comunicar com o Microsoft Excel e manipular os componentes (caixas de seleção, botões, etc) que formam a interface da FatFree;
- `FatFreeResults`: esta classe permite o acesso facilitado aos dados contidos em um determinado arquivo da FatFree.

Na Figura 19 vê-se um diagrama UML com uma visão geral do pacote `dnv`.

Figura 19 – Digrama UML do pacote dnv.



Fonte: Autor (2020)

#### 4.2.2.5 Pacote plots

Pacote responsável por agregar as funções de geração de gráficos dos resultados. Disponibilizar essas funções como parte da ferramenta permite que os gráficos com os resultados das análises sejam gerados automaticamente, além de padronizar o visual e a forma de apresentação. As principais classes neste pacote são:

- Plot: é construída em cima da classe Figure da biblioteca Bokeh<sup>4</sup>, que permite a criação de gráficos interativos em HTML. Os gráficos gerados por essa biblioteca possuem controles que permitem, por exemplo, que o usuário dê zoom ou translate os dados exibidos, ative/desative os itens mostrados por meio de clique nos respectivos itens nas legendas, salve o gráfico em formato de imagem estática (png). Na Figura 20 vê-se um exemplo desses gráficos, destacando os elementos interativos em vermelho;

Figura 20 – Exemplo de gráfico customizado criado com a biblioteca Bokeh.



Fonte: Autor (2020)

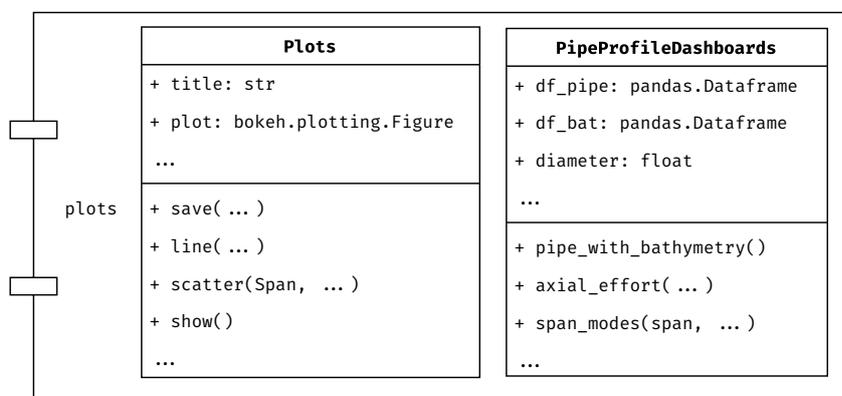
<sup>4</sup> [www.bokeh.org](http://www.bokeh.org)

O objetivo da classe `Plot` é simplificar a criação dos gráficos devido a implementação de métodos que facilitam desde alterar os componentes mais comuns das figuras, como títulos dos eixos e legendas, até combinar gráficos e formar *dashboards* com gráficos integrados. A combinação de gráficos pode ser feita por meio dos operadores matemáticos:

- $a + b$ : para sobrepor os gráficos  $a$  e  $b$ ;
  - $a | b$ : para posicionar dois gráficos, com  $a$  à esquerda de  $b$ ;
  - $a / b$ : para posicionar dois gráficos, com  $a$  acima de  $b$ ;
- `PipeProfileDashboards`: esta classe se utiliza da classe `Plot` para facilitar a criação de um tipo *dashboard* recorrente no processo de análise: dois gráficos empilhados verticalmente, um com dados que podem ser associados a posição ao longo do duto (como esforço axial, modos de vibração, enterramento, vida à fadiga, etc), e o outro abaixo dele com o perfil do duto. Nesse tipo de arranjo é interessante que os eixos das abcissas estejam atrelados, permitindo ao usuário movimentar os dois eixos em sincronia.

Na Figura 21 vê-se um diagrama UML com uma visão geral do pacote `plots`.

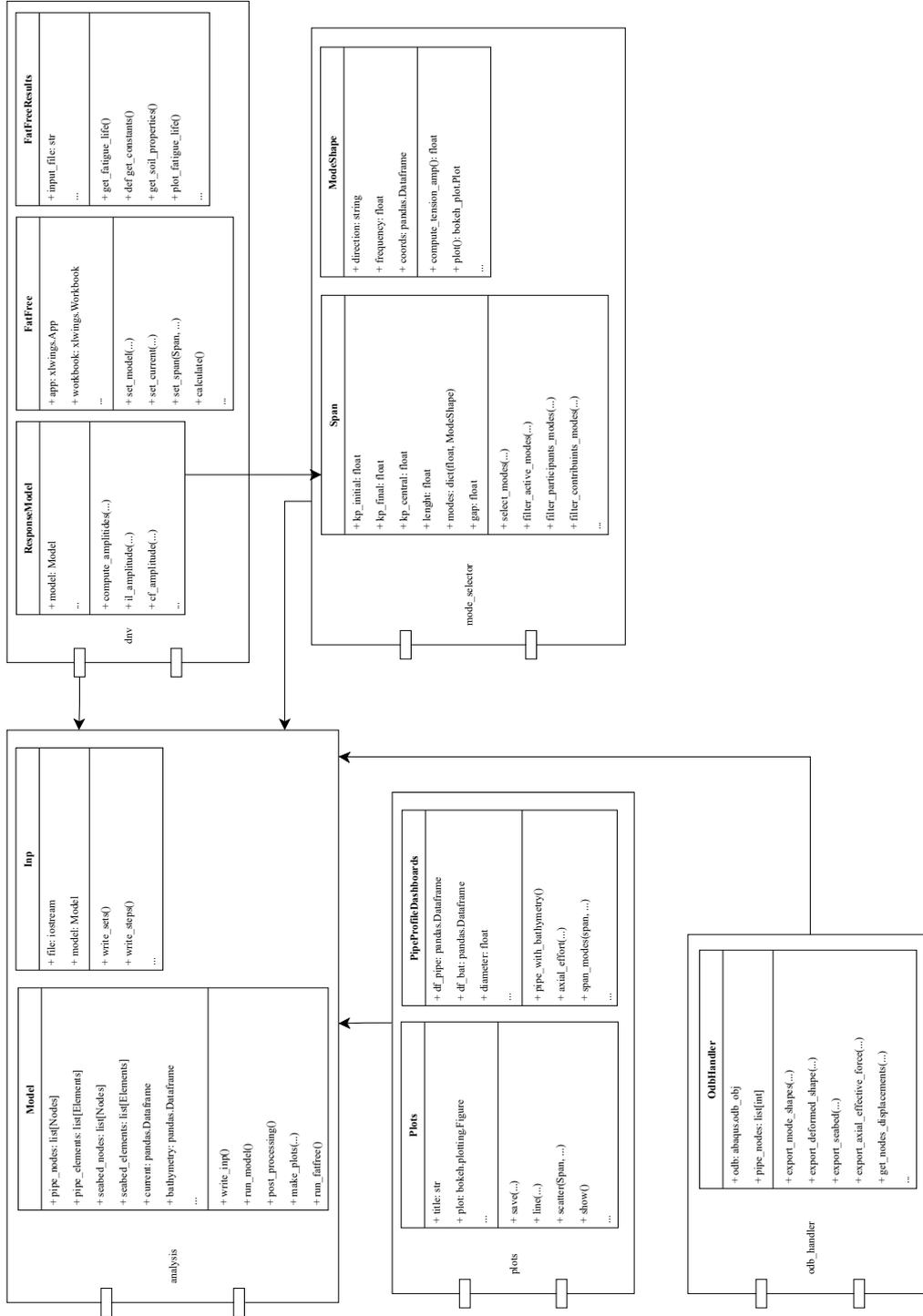
Figura 21 – Diagrama UML do pacote `plots`.



Fonte: Autor (2020)

A relação entre esses pacotes, módulos e classes pode ser vista de forma geral no diagrama apresentado na Figura 22.

Figura 22 – Digrama UML global



Fonte: Autor (2021)

## 5 APLICAÇÕES E RESULTADOS

Este capítulo apresenta um exemplo básico de aplicação da ferramenta, apresentando as funções principais para geração de modelos (.inp) e para simulação no ABAQUS, chamada do ABAQUS, pós-processamento e visualização dos resultados. Serão apresentados tanto os resultados obtidos com a simulação quanto os códigos básicos utilizados para geração desses resultados, bem como trechos de código demonstrando a forma de utilização das principais funcionalidades implementadas.

Embora não haja restrições da ferramenta quanto ao perfil da batimetria, este exemplo será um caso de uma batimetria simples, uma vez que o intuito é apresentar modelo cujos resultados (mais especificamente, os valores das frequências obtidas pela análise modal no ABAQUS) podem ser validados com os resultados previstos (calculados pela FatFree). Vale salientar que os dados utilizados são de domínio público ou fictícios, mas não representam um caso real, até porque esse tipo de dado é mantido sob sigilo devido a aspectos relacionados à segurança, competitividade tecnológica e propriedade intelectual das empresas do setor.

### 5.1 Exemplo: análise de vida à fadiga de um pequeno trecho de duto

O modelo consiste um pequeno trecho de duto com comprimento total de 1000 m com um vão de 17 m posicionado no centro. A grande extensão do duto a direita e à esquerda do vão é usada simplesmente para reduzir a influência das condições de contorno nas extremidades.

Para iniciar as análises, todos esses dados devem estar devidamente estruturados num arquivo de entrada no formato JSON. Para carregar esses dados e criar uma instância da classe `Model`, pode-se usar a função `load_json`, como mostrado na Figura 23.

Antes de iniciar a simulação, que pode ser um processo custoso, é importante revisar se o modelo está minimamente de acordo com o desejado. Pode-se, por exemplo, visualizar o perfil da batimetria tomando partido de uma das funções gráficas implementadas no módulo `plots`. O código para geração desse gráfico está na Figura 24, cujo resultado é mostrado na Figura 25.

A simulação realizada tem a seguinte sequência de passos de carga:

1. Aplica-se o peso do duto vazio;
2. Aplica-se a pressão externa;

Figura 23 – Código para carregamento dos dados de entrada.

```

1 from integrispan.model_generator.model_generator import load_json
2 model = load_json("inputs.json")

```

Figura 24 – Código para geração de Figura 25 com o perfil da batimetria

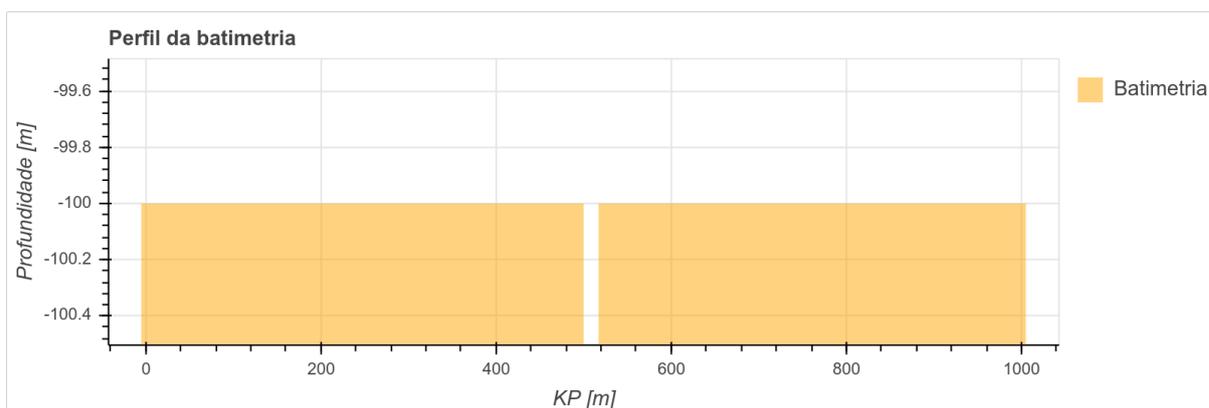
```

1 from integrispan.plots import plots
2 bathymetry_plot = plots.bathymetry(model.bat)
3 bathymetry_plot.save("perfil_batimetria.html")

```

Fonte: Autor (2020)

Figura 25 – Perfil do modelo.



Fonte: Autor (2020)

3. Aplica-se a tração de lançamento;
4. Aplica-se o deslocamento vertical e assenta-se o duto;
5. Restaura-se o atrito axial;
6. Ativa-se as molas;
7. Remove-se a tração de lançamento;
8. Aplica-se a pressão do teste hidrostático;
9. Remove-se a pressão do teste hidrostático;
10. Aplica-se a pressão operacional;
11. Obtêm-se os modos de vibração (análise modal).

A geração dos arquivos para simulação no ABAQUS contendo as instruções para toda essa sequência de passo é realizada usando o método `write_inps:model.write_inps()`. Isso deve gerar dois arquivos dentro de um diretório chamado `exemplo`: o arquivo principal `exemplo.inp`, e `bt_exemplo.inp`, com as coordenadas que define o perfil de batimetria.

O método `run_abaqus` do objeto `model` é responsável por executar a chamada do ABAQUS de maneira programática para iniciar a simulação. Nesse método, ocorre a leitura do arquivo de registro da simulação e o seu conteúdo é exibido na tela do console a cada 5 s.

Uma vez terminada a simulação, a extração de dados do arquivo `odb` gerado pelo ABAQUS e o respectivo pós-processamento podem ser realizados com a chamada do método `post_processing` do objeto `model`. Há uma função para representação gráfica de cada um dos principais tipos de dados extraídos. A seguir, serão apresentadas algumas dessas funções e as respectivas figuras geradas por eles.

Em geral, a primeira forma de validação é uma inspeção visual da configuração deformada do duto sobre a batimetria. Desta forma é possível ver se a simulação consegue reproduzir a situação *in-loco*, especialmente os vãos. Uma vez que a etapa anterior tenha gerado os arquivos `deformed_shape.csv` (com as coordenadas da configuração deformada do duto) e `seabed.csv`, (com as coordenadas da batimetria), pode-se gerar um gráfico com o perfil do duto. O código para geração do gráfico desejado é semelhante ao exposto na Figura 26, e o gráfico resultante na Figura 27. Este e outros gráficos mais comuns são gerados automaticamente com uma chamada do método `make_plots` do objeto `model`: `model.make_plots()`.

Figura 26 – Código para geração do gráfico do perfil da configuração deformada do duto sobre a batimetria.

```

1 deformed_shape = load_csv(model.final_results_folder / "deformed_shape.csv")
2 pipe_plot = plots.pipe_profile(deformed_shape, legend="Eixo do duto")
3 seabed = load_csv(model.final_results_folder / "seabed.csv")
4 pipe_plot.title = "Configuração deformada"
5 seabed_plot = plots.bathymetry(seabed)
6 seabed_pipe_plot = pipe_plot + seabed_plot
7 seabed_pipe_plot.set_range(y=(-0.06, 0.06), x=(400, 600))
8 seabed_pipe_plot.save("configuração_deformada.html")

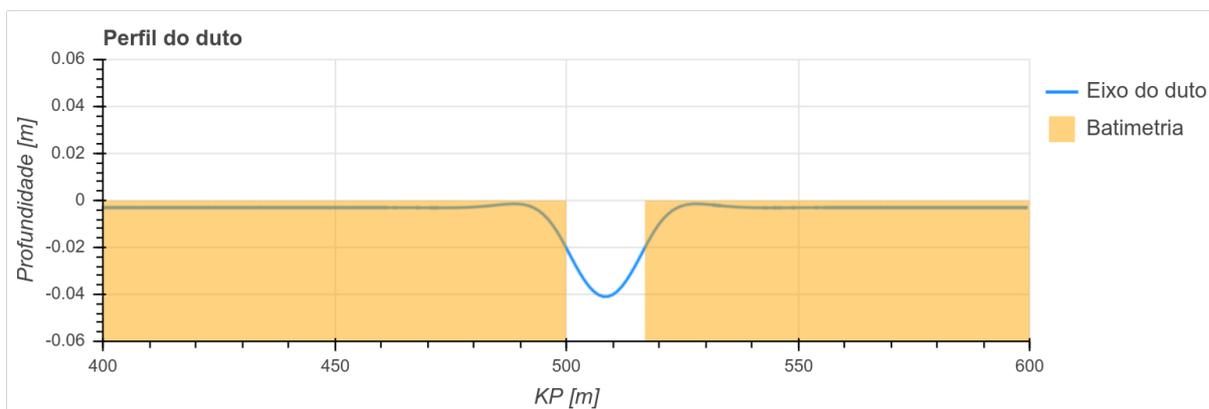
```

Fonte: Autor (2020)

Vale destacar o uso do operador de adição (+) na linha 6. Com ele, é possível combinar dois gráficos (instâncias da classe `Plot`), sobrepondo-os em um mesmo gráficos.

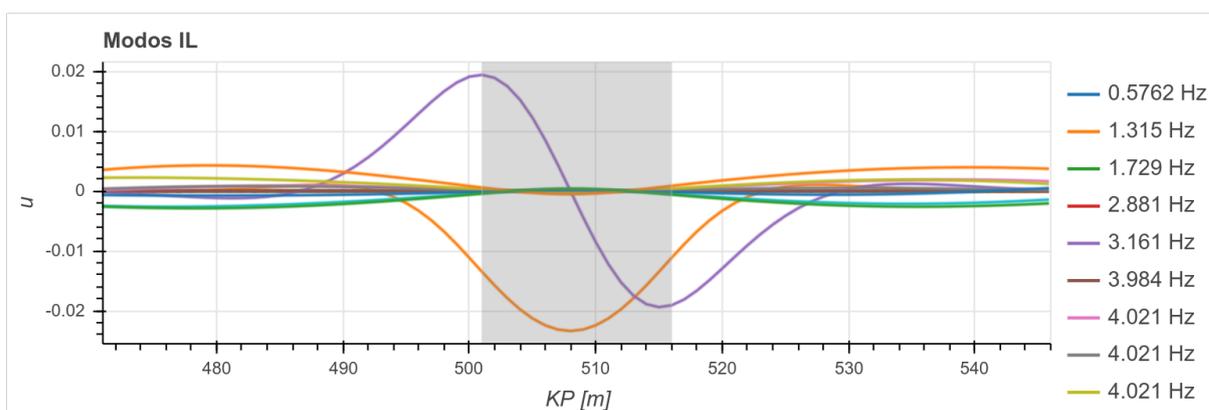
O próximo passo é a visualização dos modos de vibração com o objetivo de selecionar os que irão ser utilizados para o cálculo de fadiga. As Figuras 28 e 29 apresentam todos os modos de vibração computados pela análise modal do ABAQUS, separando-os em *in-line* e *cross-flow*, onde a região que compreende a extensão do vão está destacada com o fundo cinza.

Figura 27 – Configuração deformada do duto após a simulação.



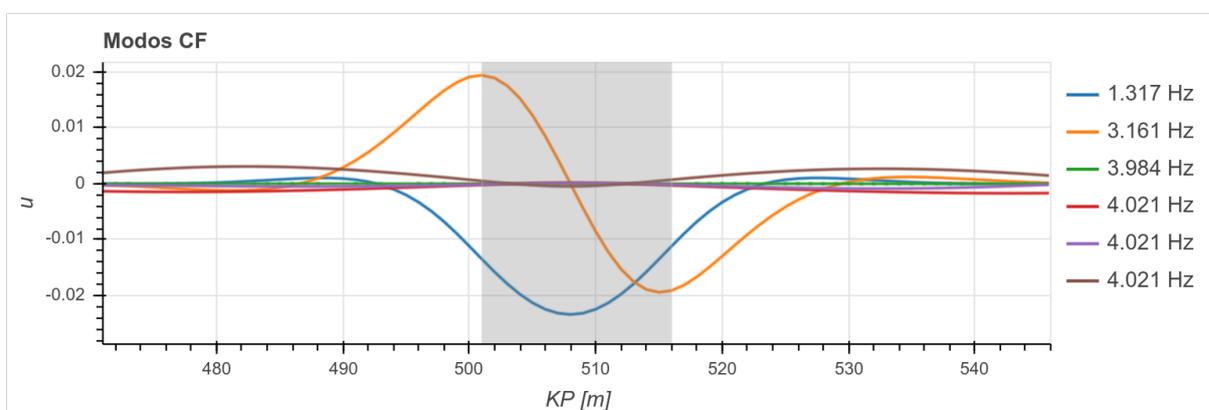
Fonte: Autor (2020)

Figura 28 – Modos de vibração *in-line* geradas na análise modal.



Fonte: Autor (2020)

Figura 29 – Modos de vibração *cross-flow* geradas na análise modal.

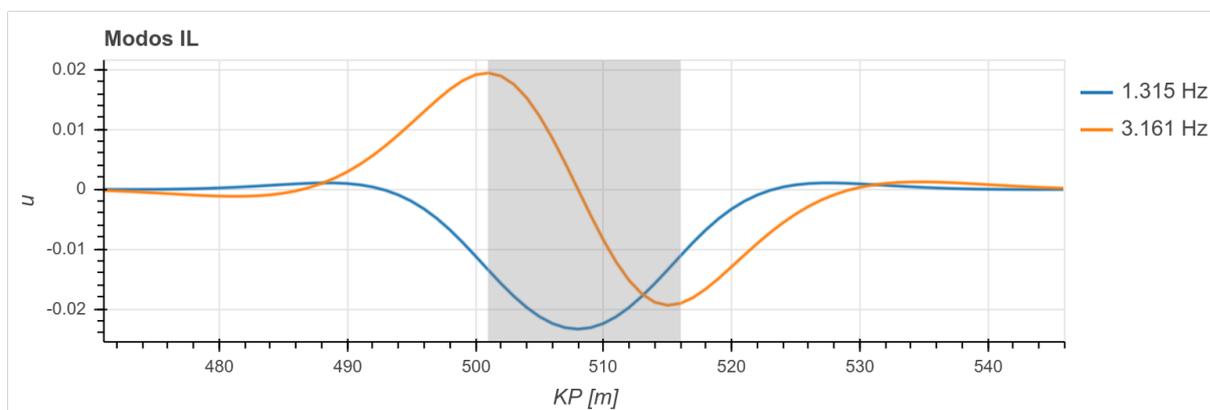


Fonte: Autor (2020)

Como é possível ver nas Figuras 28 e 29, há muitos modos espúrios, os quais devem ser desconsiderados da análise de fadiga. Esse é o papel do método `select_modes` da classe

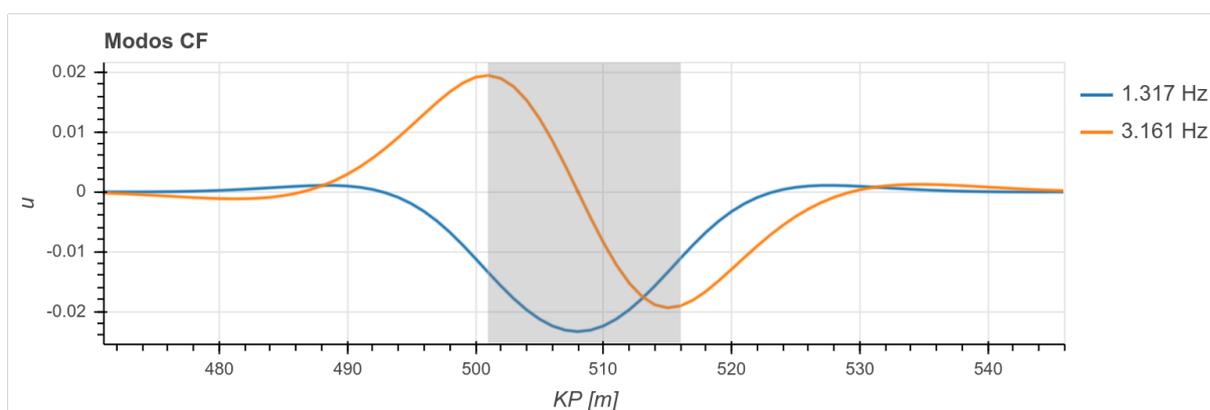
Span. Esse método consiste na implementação dos processos de seleção de modos recomendados pela DNVGL-RP-F105, apresentados na subseção 3.2.2. Para o exemplo aqui ilustrado, os resultados são os modos exibidos nas Figuras 30 e 31.

Figura 30 – Modos de vibração *in-line* selecionados pelo algoritmo implementado.



Fonte: Autor (2020)

Figura 31 – Modos de vibração *cross-flow* selecionados pelo algoritmo implementado.



Fonte: Autor (2020)

Estando satisfeito com os modos selecionados na etapa anterior, o usuário pode usar o método `run_fatfree`, para que seja feita o preenchimento de uma instância da planilha com os dados específicos do vão, modos, e dados mais gerais, como condições ambientais e coeficientes de segurança. Um exemplo da massiva entrada de dados necessária nesse processo acontece na aba *Multimode* da FatFree, onde se faz a entrada das coordenadas dos modos de vibração (colunas em azul), conforme é possível ver na Figura 32.

Com essa instância da planilha, tem-se outro ponto para validação dos resultados. Segundo o item 6.7.4 da DNVGL-RP-F105, a análise de elementos finitos para um único vão com  $L/D_s \approx 60$ , as frequências naturais de *in-line* e *cross-flow* e as faixas de tensão devem mostrar valores semelhantes com margem de  $\pm 5\%$ , considerando esforço axial efetivo nulo ( $S_{eff} = 0$ ). Para o modelo em questão, a comparação entre os valores da primeira frequência obtidos pela análise modal do ABAQUS e a FatFree é apresentada na Tabela 2.

Figura 32 – Dados inseridos na aba *Multimode* da FatFree.

Span and modal input									
Location [m]	IN-LINE				CROSS-FLOW				
	IL Mode 1		IL Mode 2		CF Mode 1		CF Mode 2		
	<input type="checkbox"/> Deactivate	f							
	A IL	153.54663	A IL	341.66245	A CF	153.496154	A CF	341.6502	
	Mode Shape	Stress Amplitude							
500	-0.47752	0.001	0.984205	0.001	-0.4777062	0.001	0.984133	0.001	
501	-0.57727	0.001	0.999995	0.001	-0.5774294	0.001	0.999924	0.001	
502	-0.6746	51.331743	0.973812	341.66245	-0.674744	51.3549717	0.973744	341.6471	
503	-0.76514	80.854772	0.902425	337.3906	-0.7652559	80.8635004	0.902362	337.3761	
504	-0.84496	106.10885	0.786383	306.4768	-0.8450419	106.100902	0.786327	306.4685	
505	-0.91069	126.45517	0.629768	252.20913	-0.9107467	126.434109	0.62972	252.204	
506	-0.95963	141.37592	0.439759	179.15955	-0.9596672	141.341064	0.43972	179.1602	
507	-0.98981	150.48487	0.226026	92.961151	-0.9898215	150.439804	0.225996	92.96488	
508	-1	153.54663	-1.6E-05	0.0059713	-1	153.496154	-3.8E-05	0.000986	
509	-0.9898	150.48378	-0.22606	92.972565	-0.9897965	150.430393	-0.22607	92.96348	
510	-0.95963	141.37219	-0.43979	179.16974	-0.9596185	141.319852	-0.4398	179.1592	
511	-0.91068	126.45113	-0.62979	252.2166	-0.9106771	126.405428	-0.6298	252.2052	
512	-0.84494	106.10213	-0.7864	306.48325	-0.8449551	106.064105	-0.7864	306.4703	
513	-0.76513	80.847702	-0.90244	337.39105	-0.7651569	80.8234916	-0.90244	337.3786	
514	-0.67458	51.322707	-0.97382	341.66131	-0.674638	51.310631	-0.97382	341.6502	
515	-0.57725	18.261118	-1	317.43829	-0.5773224	18.2644031	-1	317.4294	
516	-0.47751	0.001	-0.98421	0.001	-0.4776038	0.001	-0.98421	0.001	
517	-0.38012	0.001	-0.9335	0.001	-0.3802401	0.001	-0.93351	0.001	

Fonte: Autor 2020

Tabela 2 – Comparação entre valores para a primeira frequência.

Direção	ABAQUS	FatFree	Diferença relativa
<i>In-line</i>	1,315	1,351	-2,66%
<i>Cross-flow</i>	1,317	1,357	-2,95%

Uma vez que se tenha atingido sucesso na tarefa de modelar o problema de modo a condizentes, pode-se agora usar a classe `FatFreeResults` para extrair os resultados de uma dada instância da `FatFree`. Por exemplo, pode-se construir um gráfico da vida à fadiga ao longo do vão com perfil do duto com a batimetria abaixo dele (ver Figura 27), como mostra a Figura 33.

Figura 33 – Código para geração dos gráficos de vida à fadiga.

```

1 from integrispan.dnv.fatfree_results import FatFreeResults
2 fatfree_results = FatFreeResults("FatFree_kp_508.5.xls", ".")
3 fatigue_life_plot = fatfree_results.plot_fatigue_life(baseline=20)
4 dashboard = fatigue_life_plot / seabed_pipe_plot
5 dashboard.save("fatigue_pipe_dashboard.html")

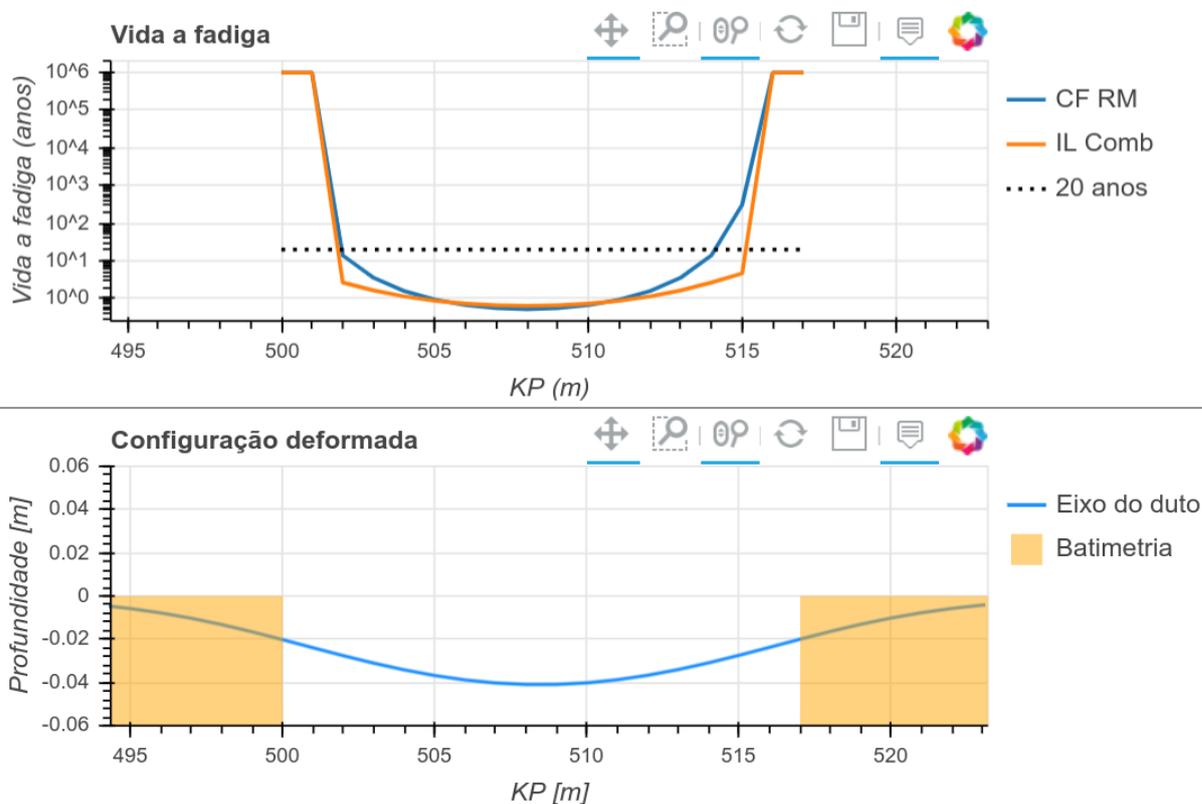
```

Fonte: Autor (2020)

Vale destacar o uso do operador de divisão (`/`) na linha 4. Com ele, é possível combinar dois gráficos (instâncias da classe `Plot`) posicionando-os um sobre o outro em um mesmo gráfico.

O gráfico resultante é exibido na Figura 34, onde se tem a vida à fadiga nas direções *in-line* (IL Comb) e *cross-flow* (CF RM) na região do vão, bem como uma linha de referência que ajuda a identificar os trechos cuja previsão de vida à fadiga está abaixo do esperado.

Figura 34 – Gráfico de vida à fadiga e perfil do duto.



Fonte: Autor (2020)

Por fim, vale destacar que a execução dos códigos pertencentes a ferramenta apresentadas neste capítulo levam não mais que poucos segundos cada. Isso representa uma redução de tempo significativa em relação ao cenário sem uso da ferramenta, onde esses processos poderiam levar de horas a dias. Os ganhos se concentram principalmente nas operações em que envolvem a maior quantidade de dados como a geração da representação da batimetria na modelagem de elementos finitos e no preenchimento da FatFree com os dados nos modos de vibração.

## 6 CONCLUSÕES

A tendência da indústria de óleo e gás de investimentos em ferramentas computacionais que aumentem a produtividade do profissional nos processos de análise motivou o desenvolvimento deste trabalho. A principal contribuição da ferramenta é a automatização de processos, tornando transparentes complexidades que não agregam valor à análise. Dessa maneira, o usuário pode explorar um maior número de combinações de parâmetros e modelos.

Ao abstrair certas complexidades da análise, a ferramenta também diminui a curva de aprendizagem para novos profissionais, tornando-os produtivos mais rapidamente. Para além do ganho em produtividade, o fato de a ferramenta incluir as funcionalidades de geração de gráficos permite aos usuários padronizar a elaboração e apresentação dos resultados das análises. Essa padronização facilita o entendimento destes resultados e favorece a adoção de aplicações construídas com base na ferramenta dentro de um grupo de usuários.

O ecossistema de bibliotecas e ferramentas existentes ao redor da linguagem Python se mostrou bastante útil no desenvolvimento da ferramenta, e deve facilitar também a sua utilização. Aqui pode-se destacar o ambiente Jupyter<sup>1</sup>, que permite a execução e visualização da saída de blocos de código de forma iterativa. O uso conjunto deste ambiente com o pacote `plots` da ferramenta deve facilitar o fluxo de análise exploratório dos dados e dos resultados.

A principal dificuldade no desenvolvimento da ferramenta se deu na integração com as ferramentas de terceiros (ABAQUS e FatFree). No caso da planilha FatFree, os dados devem ser inseridos nas células da planilha na própria aplicação. Se fosse possível a utilização de um arquivo de entrada com os dados, assim como o `.inp` utilizado pelo ABAQUS, facilitaria sua utilização de forma programática. O ABAQUS por sua vez tem um formato do arquivo de saída proprietário (`odb`), cujo acesso aos dados só é possível com o uso do próprio ABAQUS. Essas limitações diminuem consideravelmente a flexibilidades da implementação de um `_como` aqui desenvolvido.

### 6.1 Sugestões de trabalhos futuros

Aqui faz-se a apresentação de algumas sugestões de melhorias na ferramenta que podem melhorar e expandir suas funcionalidades. Alguns destes itens já estão em desenvolvimento no contexto do projeto IntegriSpan da PETROBRAS.

- Implementação de outros modelos de análise. Os arquivos de entrada para simulação gerados pela ferramenta representam o tipo de análise corriqueiro apresentado na seção 3.1, mas existem outros tipos de simulação com uma sequência de passos de carga ou modelagem diferentes;

---

<sup>1</sup> <https://jupyter.org/>

- Refatorar classes grandes para melhorar a reutilização de código na implementação de novos modelos. Os principais exemplos seriam as classes `Model` e `Inp`;
- Experimentar a utilização de uma linguagem de *templates* para a geração dos arquivos `.inp`, a exemplo da linguagem Jinja<sup>2</sup>. Isso pode simplificar significativamente o código da classe `Inp`, e até trazer ganho de performance;
- Estudar a utilização de programação paralela. As funcionalidades de pós-processamento podem se beneficiar grandemente do provável ganho de performance;
- Implementação dos cálculos de fadiga realizados pela FatFree dentro da ferramenta, eliminando a dependência da FatFree. Isso aumentaria a integração dessas funcionalidades na ferramenta;
- Elaborar uma documentação técnica, onde seja possível consultar informações sobre qualquer classe e seus respectivos atributos e métodos, facilitando o entendimento global da ferramenta para outros desenvolvedores.

---

<sup>2</sup> <https://jinja.palletsprojects.com>

## REFERÊNCIAS

- BAI, Q.; BAI, Y. *Subsea Pipeline Design, Analysis, and Installation*. [S.l.]: Elsevier Inc., 2014. ISBN 978-0-12-386888-6. Citado 7 vezes nas páginas 22, 23, 24, 25, 26, 27 e 31.
- BATCHELOR, G. K. *An Introduction to Fluid Dynamics*. [S.l.]: Cambridge University Press, 1967. (Cambridge Mathematical Library). ISSN 9780521663960. Citado na página 29.
- CURRIE, I. G. *Fundamental Mechanics of Fluids*. Fourth. [S.l.]: Taylor & Francis, 2012. (Civil and Mechanical Engineering). ISSN 9781439874608. Citado na página 28.
- Dassault Systèmes Simulia. *ABAQUS Unified FEA - SIMULIA by Dassault Systèmes®*. 2018. Disponível em: <<https://www.3ds.com/products-services/simulia/products/ABAQUS/>>. Acesso em: 17 set. 2019. Citado na página 18.
- Det Norske Veritas. *DNV-GL-RP-F105 Free Spanning Pipelines 2017*. [S.l.], 2017. Citado 4 vezes nas páginas 30, 31, 32 e 33.
- DNV-GL. *Sesam for pipelines - Subsea pipeline design software*. 2020. Disponível em: <<https://www.dnvgl.com/services/subsea-pipeline-design-software-sesam-for-pipelines-2479>>. Acesso em: 17 abr. 2019. Citado na página 18.
- FUGRO. *SAGE Profile*. 2020. Disponível em: <<http://www.sage-profile.com>>. Acesso em: 17 abr. 2020. Citado na página 17.
- GAMINO, M.; ABANKWA, S.; PASCALI, R. FSI methodology for analyzing VIV on subsea piping components with practical boundary conditions. *Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering - OMAE*, v. 7, 2013. Citado na página 14.
- MARTINEZ, M. *UML - Linguagem Unificada de Modelagem - Diagramas*. 2020. Disponível em: <<https://www.infoescola.com/engenharia-de-software/uml/>>. Acesso em: 23 de nov. de 2020. Citado na página 38.
- MITTAL, A.; SLAUGHTER, A.; BANSAL, V. *From bytes to barrels: The digital transformation in upstream oil and gas*. 2017. 1–28 p. Disponível em: <<https://www2.deloitte.com/us/en/insights/industry/oil-and-gas/digital-transformation-upstream-oil-and-gas.html>><<https://www2.deloitte.com/insights/us/en/industry/oil-and-gas/digital-transformation-upstream-oil-and-gas.html>><<https://dupress.deloitte.com/dup->>. Acesso em: 1 out. 2019. Citado na página 17.
- MØRK, K. et al. Assessment Of Viv Induced Fatigue In Long Free Spanning Pipelines. In: *Proceedings of OMAE 2003*. Cancun, Mexico: International Conference on Offshore Mechanics & Arctic Engineering, 2003. p. 1–7. Citado 3 vezes nas páginas 28, 29 e 30.
- NIELSEN, F. G.; SØREIDE, T. H.; KVARME, S. O. VIV Response of Long Free Spanning Pipelines. In: *21st International Conference on Offshore Mechanics and Arctic Engineering, Volume 1*. Oslo, Norway: ASME, 2002. p. 121–129. ISBN 978-0-7918-3611-8. Citado na página 28.
- RAO, V. R. *Here's Why You Should Use Python for Scientific Research*. 2018. <https://developer.ibm.com/dwblog/2018/use-python-for-scientific-research/>. Acesso em: 17 abr. 2020. Citado na página 37.

SIMULIA. *Nonlinear Solution Methods in ABAQUS/Standard*. 2018. [https://help.3ds.com/2018/English/DSSIMULIA\\_Established/SIMACAETHERefMap/simathec-nonlinearsol.htm?ContextScope=all&id=f6d8fd0953f64ef794cf56a8990a3493#Pg0](https://help.3ds.com/2018/English/DSSIMULIA_Established/SIMACAETHERefMap/simathec-nonlinearsol.htm?ContextScope=all&id=f6d8fd0953f64ef794cf56a8990a3493#Pg0). Acesso em: 17 set. 2019. Citado 2 vezes nas páginas 24 e 28.

SUMER, B. M.; FREDSOE, J. M. A Review On Vibrations Of Marine Pipelines. *International Journal of Offshore and Polar Engineering*, International Society of Offshore and Polar Engineers, 1995. Citado na página 29.

TURA, F. et al. Guidelines for free spanning pipelines: The GUDESP project. In: *13th International Conference on Offshore Mechanics and Arctic Engineering*. United States: American Society of Mechanical Engineers, New York, NY (United States), 1994. Citado na página 30.

Van den Abeele, F.; BOËL, F.; HILL, M. Fatigue Analysis of Free Spanning Pipelines Subjected to Vortex Induced Vibrations. In: *Volume 7: CFD and VIV*. Nantes, France: American Society of Mechanical Engineers, 2013. p. V007T08A039. ISBN 978-0-7918-5541-6. Citado 2 vezes nas páginas 15 e 29.

## **APÊNDICE**

## APÊNDICE A – DESCRIÇÃO DO FORMATO DO ARQUIVO DE ENTRADA

JSON é um formato de arquivo em texto puro que representa informações atribuindo um nome (ou rótulo) que descreve o seu significado e, a seguir, o seu valor. Esta sintaxe de representação é derivada da forma utilizada pelo JavaScript para representar objetos. Muito mais que um formato de arquivo, é um modelo para armazenamento e transmissão de informações no formato texto e que é bastante utilizado por aplicações web. A representação de informações utilizada em arquivos JSON é simples, e sua forma de estruturação é bem mais compacta do que a que normalmente é feita em arquivos XML, o que torna o processamento das informações muito mais rápido. A especificação completa do formato JSON pode ser encontrado em [www.json.org](http://www.json.org).

O arquivo foi feito pensando na facilidade do usuário em reconhecer os rótulos e preenchê-los de forma fácil e prática. O arquivo compreende informações que servirão tanto para a análise do ABAQUS (construindo um ou mais arquivos `.inp`), quanto para a FatFree. A seguir descreve-se brevemente a estruturação das chaves no arquivo de entrada, apresentado nas Figura 35 a Figura 48.

### Palavras-chave no arquivo de entrada

#### A.0.1 MODEL

Dados gerais do modelo, como:

- NAME: Nome do modelo. Usado como base para o nome dos arquivos e diretórios criados;
- BATIMETRIA: indicação do arquivo de batimetria;
- RESULTS\_FOLDER: indicação do caminho para armazenamento dos resultados;
- SPRING\_PIPE\_EXTREMITY: indicação da extremidade onde será colocada a mola, a mesma na qual foi aplicada a tração residual de lançamento.

Figura 35 – Exemplo de arquivo de entrada de dados: chave MODEL.

```
"MODEL": {  
  "NAME": "Duto_Piloto_X",  
  "BATIMETRIA": "C:/Batimetria/batimetria.csv",  
  "RESULTS_FOLDER": "C:/Resultados",  
  "SPRING_PIPE_EXTREMITY": 0  
}
```

Figura 36 – Exemplo de arquivo de entrada de dados: chave FILE\_BAT.

```
"FILE_BAT": {  
  "CPU": 12,  
  "GPU": 12,  
  "INTERACTIVE": 0  
},
```

### A.0.2 FILE\_BAT

Parâmetros para a execução da simulação no ABAQUS.

- CPU: número de núcleos disponíveis para simulação;
- GPU: número de núcleos da placa gráfica disponíveis para simulação;
- INTERACTIVE: garante que as simulações ocorram sequencialmente, e não simultaneamente.

### A.0.3 CONDITIONS

Configurações gerais do modelo:

- TYPE\_SEABED:
  - 0: Batimetria modelada com elementos do tipo R3D4;
  - 1: Batimetria modelada como superfície analítica.
- CURTAIN\_SPRINGS:
  - 0: A rigidez do solo é prescrita;
  - 1: A rigidez vertical é modelada como molas do tipo *spring*.
- RUN\_MODEL:
  - 0: Utilizar o programa apenas para pós-processamento;
  - 1: Rodar o modelo no ABAQUS.
- POST\_PROCESSING:
  - 0: Rodar apenas a simulação no ABAQUS;
  - 1: Após a simulação, iniciar diretamente o pós processamento.

Figura 37 – Exemplo de arquivo de entrada de dados: chave CONDITIONS.

```
"CONDITIONS": {
  "TYPE_SEABED": 1,
  "CURTAIN_SPRINGS": 0,
  "RUN_MODEL": 1,
  "POST_PROCESSING": 1,
  "ISIGHT": 0,
  "SUPPORTS": true,
  "DELETE_FOLDER": false
},
```

- ISIGHT: chama diretamente o iSight caso seja necessário um processo de otimização ou projeto de experimento;
- SUPPORTS: indica se os suportes devem ser inseridos no modelo;
- DELETE\_FOLDER: indica se o conteúdo da pasta de resultados deve ser apagado antes da nova simulação.

#### A.0.4 MODE\_SELECTOR

Parâmetros necessários para o módulo de seleção de frequências.

- NODESET: indica o *nodeset* referente ao duto;
- ELEMENTSET: indica o *elset* referente ao duto;
- SEABEDSET: indica o set referente à batimetria;
- SPANS: dados dos vãos que serão submetidos ao pós-processamento;
- N\_MODES\_IN\_LINE: indica o número de nós que serão extraídos na direção *in-line*;
- N\_MODES\_CROSS: indica o número de nós que serão extraídos na direção *cross-flow*.

#### A.0.5 PIPE\_GEOMETRY

Indica as informações relativas ao duto, como comprimento, altura de lançamento e tamanho do elemento.

- INITIAL\_KP: ponto de início do duto a ser modelado;
- END\_KP: ponto final do duto modelado;

Figura 38 – Exemplo de arquivo de entrada de dados: chave MODE\_SELECTOR.

```
"MODE_SELECTOR": {
  "NODESET": "PIPE",
  "ELEMENTSET": "PIPE",
  "SEABEDSET": "M_FUNDO",
  "SPANS": [
    {
      "kp_inicial": 9958.2,
      "kp_final": 9975.2,
      "azimute": 60.00
    }
  ],
  "N_MODES_IN_LINE": 4,
  "N_MODES_CROSS": 3
}
```

Figura 39 – Exemplo de arquivo de entrada de dados: chave PIPE\_GEOMETRY.

```
"PIPE_GEOMETRY": {
  "INITIAL_KP": 0,
  "END_KP": 1000,
  "LAUNCH_HEIGHT": -100,
  "LendTH_ELEMENT": 0.25
},
```

- LAUNCH\_HEIGHT: altura de lançamento do duto, sob o qual será modelado a superfície fictícia;
- LendTH\_ELEMENT: comprimento do elemento do tipo PIPE31 utilizado na modelagem do duto.

#### A.0.6 AUXILIARY\_NODE

Fornecer os dados para a criação de nós auxiliares.

- OFFSET\_NODE\_SPRING: define o *offset* para a alocação da mola;
- INCREASE\_FICTION\_PLAN\_X: aumento lateral do plano fictício na extremidade inicial;
- INCREASE\_FICTION\_PLAN\_Y: aumento lateral do plano fictício na extremidade final.

#### A.0.7 CURTAIN\_SPRINGS

Caso o modelo seja simulado com cortina de molas, é necessário passar os dados para sua modelagem.

Figura 40 – Exemplo de arquivo de entrada de dados: chave AUXILIARY\_NODE.

```
"AUXILIARY_NODE": {
  "OFFSET_NODE_SPRING": 3,
  "INCREASE_FICTION_PLAN_X": 100,
  "INCREASE_FICTION_PLAN_Y": 10
},
```

Figura 41 – Exemplo de arquivo de entrada de dados: chave CURTAIN\_SPRINGS.

```
"CURTAIN_SPRINGS": {
  "HEIGHT": -60,
  "STIFFNESS": [
    [
      -207904,
      -1
    ],
    [
      0,
      0
    ],
    [
      0,
      1
    ]
  ]
}
```

- HEIGHT: cota vertical inicial das molas;
- STIFFNESS: dados de rigidez das molas.

#### A.0.8 SPRING\_STIFFNESS

Fornece os dados da rigidez das molas nas extremidades do duto.

- INITIAL: dados da mola da extremidade inicial;
- END: dados da mola da extremidade final.

#### A.0.9 PIPE\_MATERIAL

Propriedades físicas e geométricas da sessão do duto.

- DENSITY: densidade;
- ELASTICITY\_MODULE: módulo de elasticidade;

Figura 42 – Exemplo de arquivo de entrada de dados: chave `SPRING_STIFFNESS`.

```
"SPRING_STIFFNESS": {
  "INITIAL": [
    [
      -1.0001e9,
      -10.0
    ],
    [
      1.0001e9,
      10.0
    ]
  ],
  "END": [
    [
      -1.0001e9,
      -10.0
    ],
    [
      1.0001e9,
      10.0
    ]
  ]
},
```

Figura 43 – Exemplo de arquivo de entrada de dados: chave `PIPE_MATERIAL`.

```
"PIPE_MATERIAL": {
  "DENSITY": 9850.0,
  "ELASTICITY_MODULE": 2.07e11,
  "POISSON": 0.3,
  "COEFFICIENT_EXPANSION": 1.17e-5,
  "YIELD_STRESS": 4.15e8,
  "PLASTIC_DEFORMATION": 0.00,
  "EXTERNAL_RADIUS": 0.200,
  "THICKNESS": 0.0150
},
```

- `POISSON`: coeficiente de *Poisson*;
- `COEFFICIENT_EXPANSION`: coeficiente de expansão térmica;
- `YIELD_STRESS`: limite de escoamento do aço;
- `PLASTIC_DEFORMATION`: limite de escoamento;
- `EXTERNAL_RADIUS`: raio externo do duto;
- `THICKNESS`: espessura da parede duto.

#### A.0.10 CONTACT\_PIPE\_SEABED

Propriedades do contato entre o duto e a superfície referente à batimetria.

Figura 44 – Exemplo de arquivo de entrada de dados: chave MODE\_SELECTOR.

```

"CONTACT_PIPE_SEABED": {
  "HCRT": 10000,
  "ELASTIC_SLIP": 0.001,
  "FRICTION_COEFF_1": 0.6,
  "FRICTION_COEFF_2": 0.8,
  "STABILIZE": 1e-8,
  "STIFFNESS": [
    [
      0,
      0
    ],
    [
      1000,
      1
    ]
  ]
},

```

- HCRT: define quanto a superfície *slave* pode penetrar na superfície *master* antes que o ABAQUS abandone o incremento atual e tente novamente com um incremento menor;
- ELASTIC\_SLIP: escoamento elástico permissível a ser usado no método de rigidez para aderência à fricção
- FRICTION\_COEFF\_1: coeficiente de atrito;
- FRICTION\_COEFF\_2: coeficiente de atrito;
- STABILIZE: parâmetro de estabilidade do contato;
- STIFFNESS: dados de rigidez do solo, para interação duto-solo.

#### A.0.11 CONTACT\_PIPE\_PLAN

Fornece as propriedades do contato entre o duto e a superfície fictícia.

#### A.0.12 STEPS\_DEFAULTS

Reúne as informações recorrentes em todos os passos de carga (*steps*) no arquivo \*.inp.

- MAXIMUM\_INCREMENT\_NUMBER: define o máximo de incrementos para o *step*;
- AUTOMATIC\_STABILIZATION: parâmetro de estabilização para casos onde há contato;
- INITIAL\_SIZE\_INCREMENT: valor inicial do incremento;
- TOTAL\_STEP\_TIME: tempo total no *step*;

Figura 45 – Exemplo de arquivo de entrada de dados: chave CONTACT\_PIPE\_PLAN.

```
"CONTACT_PIPE_PLAN": {
  "HCRT": 10000,
  "ELASTIC_SLIP": 0.001,
  "EXTENSION_ZONE": 0.2,
  "FRICTION_COEFF_1": 0,
  "FRICTION_COEFF_2": 0.8,
  "STABILIZE": 1e-8,
  "STIFFNESS": [
    [
      0,
      0
    ],
    [
      1000,
      0.1
    ]
  ]
},
```

- MINIMUM\_INCREMENT\_SIZE: número mínimo de incrementos para o *step*;
- MAXIMUM\_INCREMENT\_SIZE: número máximo de incrementos para o *step*.

STEP\_X: informação específica para cada *step* conforme os passos de carga anteriormente citados. Caso existam informações específicas do suporte, essas informações devem ser passadas respectivamente. Caso contrário, são utilizadas as informações do STEP\_DEFAULTS.

### A.0.13 SUPPORTS

Reúne as informações referentes à posição dos suportes estruturais.

- FILE\_DEFORMED\_IN\_LOCO: caminho para o arquivo de *topping*. O arquivo é utilizado como base de comparação para alocação dos suportes;
- SHIFT\_SURFACE:
- STEP: *step* a partir do qual serão alocados os suportes;
- LIST: lista de suportes de acordo com o tipo implementado. São passados o KP centro do suporte e o comprimento do mesmo:
  - No primeiro conjunto de colchetes são passados suportes do tipo *grout bag*, posicionados abaixo do duto;
  - Em seguida, suportes do tipo manta, posicionados acima do duto;
  - No terceiro colchete, suportes mecânicos do tipo pino (restrição de deslocamentos transversais ao eixo do duto e as rotações nodais).

Figura 46 – Exemplo de arquivo de entrada de dados: chave STEPS\_DEFAULTS.

```

TEPS_DEFAULTS": {
  "MAXIMUM_INCREMENT_NUMBER": 10000,
  "AUTOMATIC_STABILIZATION": 1.0e-8,
  "INITIAL_SIZE_INCREMENT": 0.0001,
  "TOTAL_STEP_TIME": 0.01,
  "MINIMUM_INCREMENT_SIZE": 1.0e-30,
  "MAXIMUM_INCREMENT_SIZE": 0.01
},
"STEP_1": {
  "EMPTY_SUBMERSE_WEIGHT": 1.4
},
"STEP_2": {
  "INITIAL_SIZE_INCREMENT": 0.1,
  "TOTAL_STEP_TIME": 2.1,
  "MAXIMUM_INCREMENT_SIZE": 2.1,
  "EXTERNAL_PRESSURE": 103542.25,
  "EFFECTIVE_DIAMETER": 0.4523
},
"STEP_3": {
  "INITIAL_SIZE_INCREMENT": 0.001,
  "RELEASE_TRACTION": 35000.00
},
"STEP_4": {
  "MAXIMUM_INCREMENT_NUMBER": 100000,
  "DISPLACEMENT_FICTITIOUS_PLANE": -200.00
},
"STEP_5": {},
"STEP_6": {},
"STEP_7": {},
"STEP_8": {
  "INITIAL_SIZE_INCREMENT": 0.1,
  "TOTAL_STEP_TIME": 24.93,
  "MAXIMUM_INCREMENT_SIZE": 5.00,
  "INTERNAL_PRESSURE": 3654453.0,
  "WEIGHT_SUBMERGED_FLOODED": 2.62
},
"STEP_9": {
  "INITIAL_SIZE_INCREMENT": 0.1,
  "TOTAL_STEP_TIME": 24.93,
  "MAXIMUM_INCREMENT_SIZE": 5
},
"STEP_10": {
  "INITIAL_SIZE_INCREMENT": 0.1,
  "TOTAL_STEP_TIME": 15.12,
  "MAXIMUM_INCREMENT_SIZE": 5,
  "WEIGHT_SUBMERGED_OPERATIONAL": 0.85,
  "OPERATION_PRESSURE": 1265413
},
"STEP_11": {
  "NUMBER_MODES": 100,
  "MAXIMUM_NUMBER_INTERACTIONS": 200
},

```

Figura 47 – Exemplo de arquivo de entrada de dados: chave SUPPORTS.

```
"SUPPORTS": {
  "FILE_DEFORMED_IN_LOCO": "D:/Batimetria/arquivo_survey.eff",
  "SHIFT_SURFACE": 10,
  "STEP": 7,
  "LIST": [
    [
      [
        845,
        1
      ]
    ],
    [],
    [],
    []
  ]
},
```

- Por fim, suportes mecânicos do tipo livre (restrição de deslocamento transversal ao eixo do duto)

#### A.0.14 STEP\_NODAL\_FIX

*Step* no qual serão aplicadas as restrições nodais, relacionadas aos diferentes tipos de suporte. Chave vazia refere-se aos valores passados na *keyword* STEPS\_DEFAULTS.

#### A.0.15 STEP\_SURFACE\_SUPPORTS

*Step* no qual as superfícies analíticas referentes aos suportes são posicionadas. Pode-se incluir as chaves presentes na chave STEPS\_DEFAULTS para modificar os valores.

#### A.0.16 FATFREE

Passa os dados necessários para preenchimento da FatFree via Xlwings.

- CURRENT\_FILE: indica caminho para o arquivo de corrente, retirado das ETs fornecidas pela na forma de histograma. A rotina realiza os cálculos necessários para incluir os dados na *sheet Current* da planilha de cálculo;
- ISIGHT\_FILE: indica o caminho onde será salvo o arquivo \*.txt com os dados que serão utilizados para o pós-processamento no iSight;
- Flag: permite o usuário escolher se os dados de rigidez (Kv, KL e Kv\_s) serão os passados pelo usuário acima ou extraídos do ABAQUS.

Figura 48 – Exemplo de arquivo de entrada de dados: chave FATFREE.

```
"FATFREE": {  
  "CURRENT_FILE": "D:\\Corrente\\arquivo_de_corrente.csv",  
  "ISIGHT_FILE": "D:\\Resultados\\DutoX_trecho3\\isight.txt",  
  "h": 165,  
  "L": 70,  
  "e": 0.88,  
  "d": 0,  
  "teta_pipe": 0,  
  "z_structure": 0.005,  
  "z_soil_in_line": 0.02,  
  "z_soil_cross_flow": 0.014,  
  "z_hRM": 0,  
  "Kv": 1.33e7,  
  "KL": 1e7,  
  "Kv_s": 2.5e5,  
  "SCF": 1,  
  "kc": 0,  
  "fcu": 42,  
  "k": 0.0033,  
  "p": 124.54,  
  "DT": 0.02062,  
  "Ds": 0.3238,  
  "t_concrete": 0,  
  "t_coating": 0.003,  
  "r_steel": 7850,  
  "r_concrete": 0,  
  "r_coating": 935,  
  "r_cont": 200,  
  "Turbulence_intensity_Ic": 0.04,  
  "Measurement_ref_Height_zr": 5,  
  "On_bottom_roughness_z0": 0.00004,  
  "Time_between_independent_current_events": 1.0,  
  "Flag": false  
}
```