



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA CIVIL
CURSO DE ENGENHARIA CIVIL

Gustavo André Gris Costa

Modelagem e análise iterativa da estrutura de *dolphins* de amarração

Florianópolis

2023

Gustavo André Gris Costa

Modelagem e análise iterativa da estrutura de *dolphins* de amarração

Trabalho de Conclusão de Curso submetido ao curso de Engenharia Civil do Centro Tecnológico da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Bacharel em Engenharia Civil.

Orientador: Prof. Rafael Holdorf Lopez, Dr.

Florianópolis

2023

**Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.**

Costa, Gustavo André Gris

Modelagem e análise iterativa da estrutura de dolphins de amarração / Gustavo André Gris Costa ; orientador, Rafael Holdorf Lopez, 2023.

76 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia Civil, Florianópolis, 2023.

Inclui referências.

1. Engenharia Civil. 2. Análise estrutural. 3. Portos. 4. Programação. 5. Automação. I. Lopez, Rafael Holdorf. II. Universidade Federal de Santa Catarina. Graduação em Engenharia Civil. III. Título.

Gustavo André Gris Costa

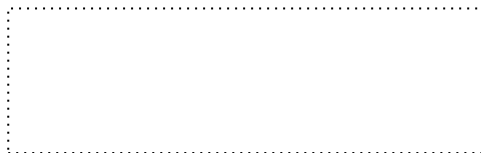
Modelagem e análise iterativa da estrutura de *dolphins* de amarração

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de “Bacharel em Engenharia Civil” e aprovado em sua forma final pelo Curso de Graduação em Engenharia Civil.

Florianópolis, 24 de novembro de 2023.

Prof^a. Liane Ramos da Silva, Dra.
Coordenadora do Curso de Engenharia Civil

Banca examinadora



Prof. Rafael Holdorf Lopez, Dr.
Orientador

Prof. Marcos Aurélio Marques Noronha, Dr.
Universidade Federal de Santa Catarina

Eng. Mateus Prado Lone
INFRAS Engenharia

Florianópolis, 2023.

AGRADECIMENTOS

Primeiramente, agradeço imensamente ao meu pai, que me puxou dos fundos, em meio à desmotivação e desistência, me colocando de volta nos trilhos. Sem a sua participação seria impossível realizar esse trabalho. Sua presença ao meu lado durante todo este processo, mesmo que remotamente, foi uma fonte de força e inspiração.

Agradeço também ao meu orientador, Professor Doutor Rafael Holdorf Lopez, pela sua paciência e empatia diante dos desafios enfrentados na elaboração do trabalho. Sua confiança em minha capacidade foi essencial para o sucesso desta pesquisa.

Além disso, sou grato ao meu antigo estágio na INFRAS Engenharia, que não apenas abriu meus olhos para esta fascinante área portuária, mas também forneceu dados valiosos e ensinamentos profundos sobre o tema. A experiência adquirida durante meu tempo lá foi inestimável e contribuiu significativamente para a realização deste trabalho.

A todos vocês, meu sincero obrigado.

*“Nossa maior fraqueza está em desistir.
O caminho mais certo de vencer é tentar mais uma
vez”.*

(Thomas Edison)

RESUMO

Este estudo aborda o uso do *OpenAPI* na automação do projeto de *dolphins* de amarração, estruturas robustas usadas em engenharia portuária para a amarração de navios. O principal objetivo foi o de avaliar a funcionalidade do *OpenAPI* na automação do processo de projeto de *dolphins* de amarração. Utilizando o *framework* .NET, desenvolveu-se um programa em C#, referenciando-se as ferramentas de *OpenAPI* do *SCIA Engineer*. Esse programa visa substituir os processos manuais e repetitivos, comuns ao se projetar *dolphins*, por uma abordagem automatizada. O trabalho demonstrou que o *OpenAPI* é capaz de automatizar de forma eficiente o processo de projeto das estruturas de *dolphins* de amarração, uma aplicação crítica na engenharia portuária e marítima. Os resultados obtidos realçam o potencial do *OpenAPI*, em conjunto com o *SCIA Engineer*, para otimizar o projeto e análise de estruturas complexas. No Brasil, onde o setor portuário é crucial para a economia, a adoção de tais metodologias pode levar a uma maior eficiência e economia em projetos de infraestruturas portuárias, contribuindo significativamente para a modernização e competitividade dessa indústria nacional.

Palavras-chave: *OpenAPI*, *SCIA Engineer*, automação de projeto, análise estrutural, otimização estrutural, *dolphin* de amarração, engenharia portuária, programação, C#, .NET.

ABSTRACT

This study addresses the use of OpenAPI for automating the design of mooring dolphins, robust structures employed in port engineering for ship mooring. The primary objective was to assess the functionality of OpenAPI in automating the design process of mooring dolphins. Utilizing the .NET framework, a program was developed in C#, referencing the OpenAPI tools of SCIA Engineer. This program aims to replace the manual and repetitive processes typically involved in designing dolphins with an automated approach. The work demonstrated that OpenAPI efficiently automates the design process of mooring dolphin structures, a critical application in port and maritime engineering. The obtained results highlight the potential of OpenAPI, in conjunction with SCIA Engineer, to optimize the design and analysis of complex structures. In Brazil, where the port sector is vital to the economy, the adoption of such methodologies could lead to greater efficiency and cost-effectiveness in port infrastructure projects, significantly contributing to the modernization and competitiveness of this national industry.

Keywords: OpenAPI, SCIA Engineer, project automation, structural analysis, structural optimization, mooring dolphins, port engineering, programming, C#, .NET.

LISTA DE FIGURAS

Figura 1 – Exemplo de <i>dolphin</i> rígido, com grupo de estacas inclinadas	18
Figura 2 – Tipos de estacas tipicamente utilizadas na construção de <i>dolphins</i>	19
Figura 3 – Layout típico de terminais marítimos	20
Figura 4 – Magnitudes básicas N, T (= V) e M dos elementos de barra	21
Figura 5 – Momentos fletores m_x e m_y dos elementos de placa.....	21
Figura 6 – Momento torsor m_{xy} dos elementos de placa ou casca.....	22
Figura 7 – Forças de cisalhamento q_x e q_y (= v_x e v_y) dos elementos de placa ou casca	22
Figura 8 – Forças de membrana n_x e n_y dos elementos de parede ou casca	22
Figura 9 – Força de cisalhamento q_{xy} (= n_{xy}) dos elementos de parede ou casca ..	23
Figura 10 – Propriedades do concreto C40.....	27
Figura 11 – Propriedades do aço da armadura CA-25.....	28
Figura 12 – Propriedades do aço A572 grau 50.....	28
Figura 13 – Propriedades da seção transversal da estaca tubular metálica	29
Figura 14 – Propriedades da seção transversal da estaca concretada.....	30
Figura 15 – Módulo de reação horizontal (K) variando com a profundidade	32
Figura 16 – Restrições do primeiro apoio de linha da estaca.....	33
Figura 17 – Restrições do apoio de ponto da extremidade da estaca.....	33
Figura 18 – Parâmetros geométricos do modelo, vista superior	38
Figura 19 – Parâmetros geométricos do modelo, vista lateral	38
Figura 20 – Parâmetros de altura do modelo	39
Figura 21 – Ângulo α , corte vertical passando pela projeção da estaca no plano XY	39
Figura 22 – Interface do Visual Studio – Utilização do <i>template SCIA Engineer</i>	41
Figura 23 – Configuração do <i>template SCIA Engineer</i>	42
Figura 24 – Referências às bibliotecas utilizadas no programa (.dll)	43
Figura 25 – <i>Assemblies</i> referenciadas	43
Figura 26 – Interface gráfica de cálculo do <i>SCIA Engineer</i> , com destaque para a funcionalidade de conectar elementos	48
Figura 27 – Caixa de diálogo de fim da análise.....	49
Figura 28 – Exemplo de envoltória ELU no relatório consolidado	52
Figura 29 – Estaca3C, ELU_4 do relatório resumido	52
Figura 30 – Estaca4C, ELU_3 do relatório resumido	53
Figura 31 – Estaca3C, ELU_4 do relatório completo	54

Figura 32 – Classe “Variáveis”, contendo os dezenove parâmetros	55
Figura 33 – Classe “Variáveis_iter”, contendo os parâmetros de iteração	55

LISTA DE GRÁFICOS

Gráfico 1 – ELU – Estacas – Forças internas – Normal – Mínimo	58
Gráfico 2 – ELU – Estacas – Forças internas – Normal – Máximo.....	59
Gráfico 3 – ELU – Estacas – Forças internas – Cortante – Mínimo	60
Gráfico 4 – ELU – Estacas – Forças internas – Cortante – Máximo	60
Gráfico 5 – ELU – Estacas – Forças internas – Momento Fletor – Mínimo.....	61
Gráfico 6 – ELU – Estacas – Forças internas – Momento Fletor – Máximo.....	61
Gráfico 7 – ELU – Estacas – Deformações – Deslocamentos	63
Gráfico 8 – ELU – Estacas – Deformações – Rotações.....	64
Gráfico 9 – ELU – Laje – Forças internas – Normal – Mínimo	66
Gráfico 10 – ELU – Laje – Forças internas – Normal – Máximo.....	67
Gráfico 11 – ELU – Laje – Forças internas – Cortante – Mínimo	68
Gráfico 12 – ELU – Laje – Forças internas – Cortante – Máximo	68
Gráfico 13 – ELU – Laje – Forças internas – Momentos (fletor e torsor) – Mínimo...	69
Gráfico 14 – ELU – Laje – Forças internas – Momentos (fletor e torsor) – Máximo..	69
Gráfico 15 – ELU – Laje – Deformações – Deslocamentos	71
Gráfico 16 – ELU – Laje – Deformações – Rotações.....	72

LISTA DE TABELAS

Tabela 1 – Combinações de Estado Limite Último (ELU)	35
Tabela 2 – Combinações de Estado Limite de Serviço Quase Permanentes (ELSQ)	36
Tabela 3 – Combinações de Estado Limite de Serviço Frequentes (ELSF).....	36
Tabela 4 – Combinações de Estado Limite de Serviço Raras (ELSR).....	36
Tabela 5 – Parâmetros geométricos do modelo	37
Tabela 6 – Demais parâmetros do modelo	39
Tabela 7 – Valores calculados a partir dos parâmetros	40
Tabela 8 – Valores de variáveis de iteração utilizados no exemplo	56
Tabela 9 – Valores das demais variáveis utilizadas no exemplo.....	56

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ADM	Analysis Data Model
API	Application Programming Interface
BIM	Building Information Modelling
BS	British Standard
ELSF	Estado Limite de Serviço Frequentes
ELSQ	Estado Limite de Serviço Quase Permanentes
ELSR	Estado Limite de Serviço Raras
ELU	Estado Limite Último
GUI	Graphical User Interface
IBS	International Building Code

SUMÁRIO

1	INTRODUÇÃO	16
1.1	OBJETIVOS	16
1.1.1	Objetivo geral	16
1.1.2	Objetivos específicos	16
2	REVISÃO BIBLIOGRÁFICA	17
3	SCIA ENGINEER	23
3.1	INTERFACE GRÁFICA.....	23
3.2	OPENAPI	24
3.2.1	Analysis data model	24
3.2.2	Results api	25
3.2.3	Esa automation	25
4	MODELO	25
4.1	ELEMENTOS	26
4.1.1	Materiais	26
4.1.2	Seções transversais	29
4.1.3	Nós	30
4.1.4	Estacas	30
4.1.5	Laje	31
4.1.6	Cabeços de amarração	31
4.1.7	Apoios	31
4.2	AÇÕES	33
4.2.1	Peso próprio	34
4.2.2	Sobrecarga	34
4.2.3	Corrente	34
4.2.4	Amarração	34
4.3	COMBINAÇÕES DE AÇÕES.....	35
4.3.1	Estado limite último	35
4.3.2	Estado limite de serviço	35
4.3.2.1	<i>Quase permanentes</i>	35
4.3.2.2	<i>Frequentes</i>	36
4.3.2.3	<i>Raras</i>	36
5	PARÂMETROS DO MODELO	37

5.1	PARÂMETROS GEOMÉTRICOS	37
5.2	DEMAIS PARÂMETROS	39
5.3	VALORES CALCULADOS	40
6	PROGRAMA E SCIA OPENAPI.....	41
7	AUTOMAÇÃO DO MODELO	54
8	CÁLCULOS E RESULTADOS	57
8.1	ESTACAS	57
8.1.1	Forças internas	57
8.1.2	Deformações	62
8.2	LAJE	65
8.2.1	Forças internas	65
8.2.2	Deformações	70
9	CONCLUSÃO	73
	REFERÊNCIAS	75

1 INTRODUÇÃO

Os *dolphins* são estruturas portuárias utilizadas para atracação e amarração de navios. Possuem flexibilidade na sua concepção, podendo ser de diversos tamanhos, formatos e materiais. Com a gradativa conscientização da população sobre as vantagens que o transporte aquaviário oferece, a demanda por essas estruturas verá um aumento significativo.

Os esforços de atracação e amarração de navios ocorrem predominantemente na direção horizontal e com grandes valores, sendo uma situação de carregamento incomum na engenharia civil. Além disso, a estrutura dos *dolphins* é executada offshore e existem muitos detalhes construtivos a serem cumpridos, como o posicionamento, ângulo de inclinação e orientação das estacas, necessitando de mão de obra especializada para sua construção.

Juntando os fatores de aumento de demanda e de complexidade construtiva dos *dolphins*, o presente trabalho tem como objetivo facilitar o processo de projetar essas estruturas, visto que, nos dias de hoje, esse processo é feito por modelagem manual repetitiva. Para isso, a ferramenta de programação *OpenAPI*, do *software* de análise estrutural *SCIA Engineer*, terá sua funcionalidade testada para a implementação de automação nos projetos de *dolphins*.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Desenvolver um programa capaz de acelerar o processo manual e repetitivo de modelagem estrutural comumente utilizado na concepção de *dolphins* de amarração, estruturas portuárias *offshore*, utilizadas para a amarração de navios.

1.1.2 Objetivos específicos

Os objetivos deste trabalho são:

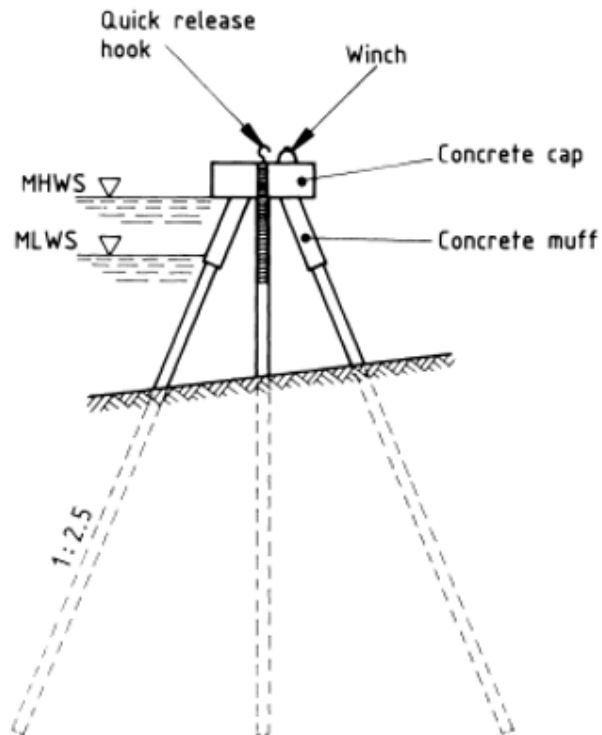
- a) realizar a modelagem e a análise estrutural de um *dolphin* de amarração automaticamente, utilizando parâmetros e o *OpenAPI* do *software SCIA Engineer*;
- b) implementar a iteratividade no programa, possibilitando a criação de vários *dolphins* com diferentes combinações dos principais parâmetros de formação da estrutura;
- c) Utilizar o programa desenvolvido para obter resultados de análise estrutural de diversos *dolphins*;
- d) Avaliar os resultados para entender a influência da variação dos parâmetros iterados.

2 REVISÃO BIBLIOGRÁFICA

Dolphins são estruturas solitárias, utilizadas para absorver o impacto durante a atracação de navios (*dolphin* de atracação), ou como pontos de amarração para navios (*dolphin* de amarração). A diversidade das estruturas de *dolphins* varia desde simples aglomerados de estacas de madeira até estruturas metálicas em águas profundas (GAYTHWAITE, 2004, p. 235). De acordo com Tsinker (1996, p. 898), *dolphins* de amarração, geralmente posicionados atrás da linha de atracação, possuem estruturas similares aos de atracação, com a principal diferença sendo a sua função exclusiva de segurar as linhas de amarração dos navios.

A norma britânica BS 6349-2 (1988, p. 93) define que as estruturas de *dolphins* podem ser rígidas ou flexíveis. De acordo com a na norma britânica BS 6349-1 (2000, p. 138), estruturas rígidas são aquelas que suportam cargas horizontais principalmente por meio de compressão ou tração diretas. No presente trabalho, serão abordados *dolphins* rígidos, mais especificamente, com grupo de estacas inclinadas, como ilustra a figura 1.




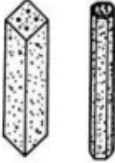

Figura 1 – Exemplo de *dolphin* rígido, com grupo de estacas inclinadas



Fonte: Bertlin & Partners, 1969 apud BS 6349-2, 1988, p. 96

A figura 2 ilustra os principais tipos de estacas utilizadas na construção de *dolphins*, sendo a seção tubular metálica concretada a opção selecionada para a elaboração do presente trabalho.

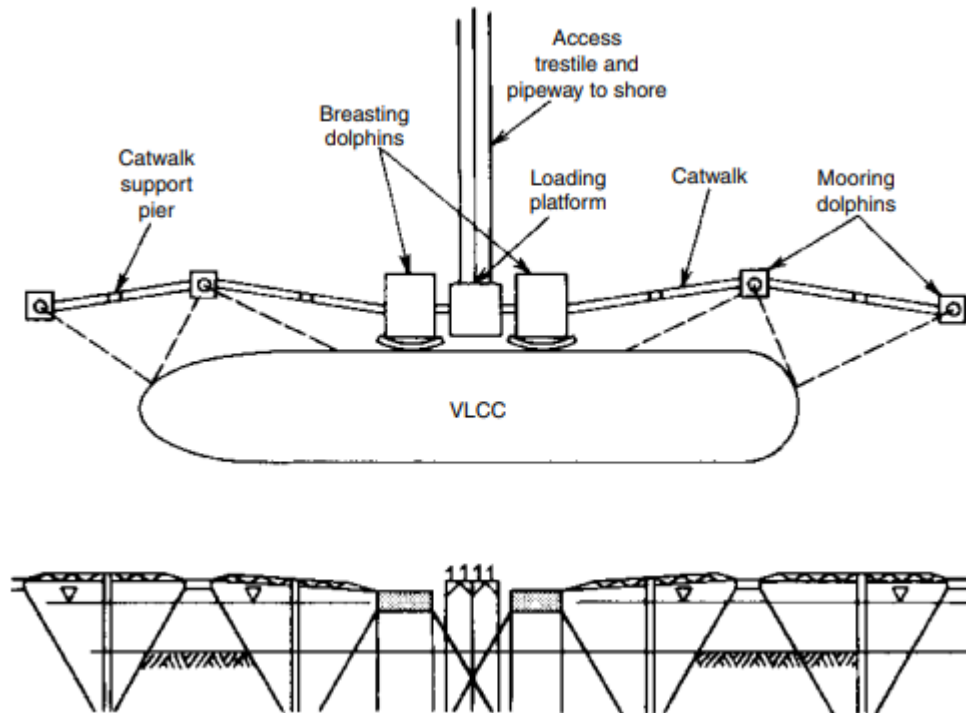
Figura 2 – Tipos de estacas tipicamente utilizadas na construção de *dolphins*

PILE TYPE	TYPICAL DIMENSIONS AND LENGTHS	USUAL RANGE OF CAPACITIES AND COLUMN LENGTHS	REMARKS
Timber 	6"-8" Tip diameter 12"-20" Butt diameter Douglas fir to 80' Southern yellow pine to 65' lengths Special order up to 125'	Typically limited to between 15 and 20 tons for all column lengths	Either Southern yellow pine or Douglas fir with ready available lengths of up to 60 feet. Piles are usually pressure treated with creosote or CCA. Greenheart piles are usually untreated.
Steel H-Pile 	Section depth 8" to 14" Unspliced- 60'-80' lengths Splices- unlimited length	40 to 120 ton capacity with effective lengths of up to 60'.	Low displacement, able to penetrate through some obstructions, and easily spliced. The pile is vulnerable to corrosion and may be damaged or deflected when encountering obstructions. Favored for end bearing on rock.
Concrete Filled Steel Pipe 	8" to 48" ϕ 5/16" to 3/4 wall thickness. Unspliced- 60'-80' lengths Splices- unlimited lengths	40 to 200 tons with effective lengths up to 100'	Displacement type piles may be driven either open or close ended, easily spliced, and provides good bending resistance
Precast Concrete 	12" to 24" round, octagonal or square 60' to 120' lengths unspliced	20 to 120 tons with effective lengths up to 80'	High displacement pile may be provided with good corrosion resistance, tolerable of hard driving stresses and vulnerable to large handling stresses.
Concrete Cylinder 	30" to 54" diameter 150'-200' + lengths	120 to 240 tons with effective lengths up to 250'	Prestressing allows for large handling stresses and capable of tolerating high bending stress induced by lateral loading and long unsupported lengths.

Fonte: Gaythwaite, 2016, p. 327

Conforme Gerwick Junior (2007, p. 416), o terminal marítimo típico consiste em uma plataforma de carregamento, dois (ou quatro) grandes *dolphins* de atracação e quatro *dolphins* de amarração, comumente interligados por passarelas metálicas, conforme ilustra a figura abaixo.

Figura 3 – Layout típico de terminais marítimos

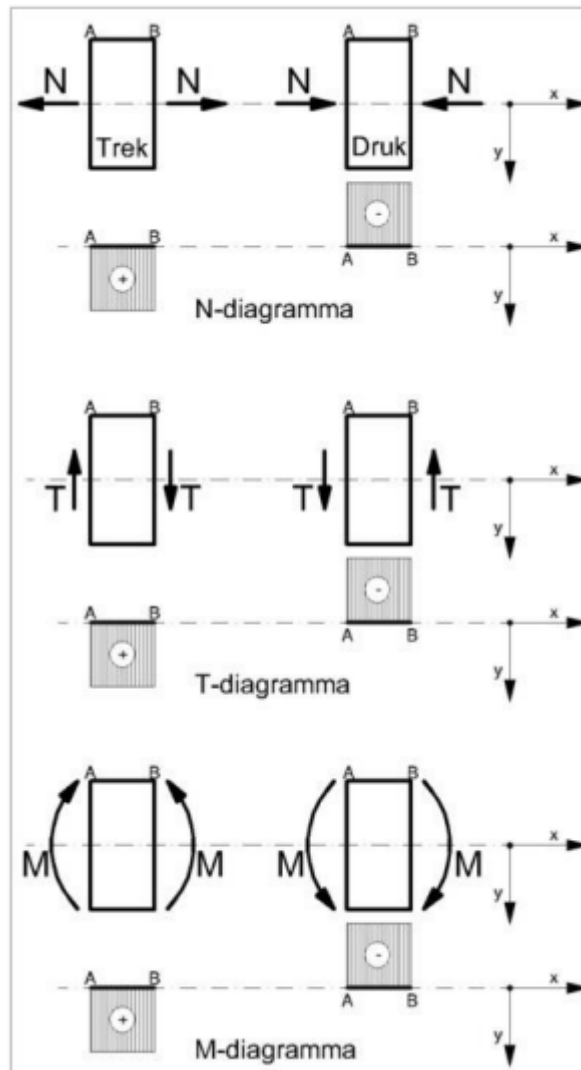


Fonte: Gerwick Junior, 2007, p. 417

A partir desse *layout*, é possível perceber que existe, nos *dolphins* intermediários, a amarração de dois cabos em apenas uma estrutura estaqueada. Por ser a situação mais crítica dentre os *dolphins* de amarração, o modelo criado e analisado neste trabalho será desenvolvido com dois cabeços de amarração suportando duas ações de amarração simultâneas.

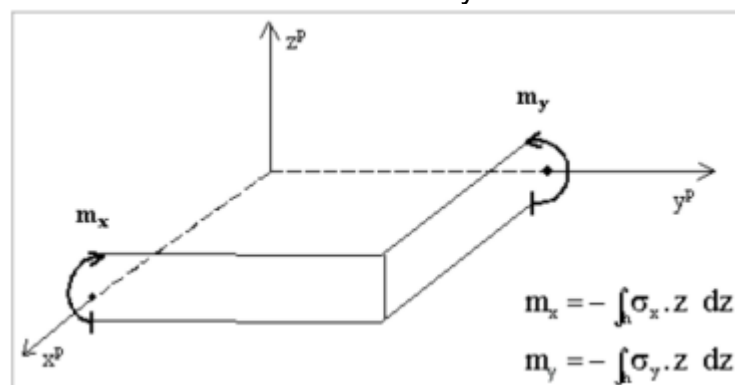
A análise estrutural dos *dolphins* será feita através do *software SCIA Engineer*. Da figura 4 até a figura 9 mostram-se os diagramas das magnitudes básicas resultantes do cálculo da malha de elementos finitos dos elementos de barra, placa, parede e casca.

Figura 4 – Magnitudes básicas N, T (= V) e M dos elementos de barra



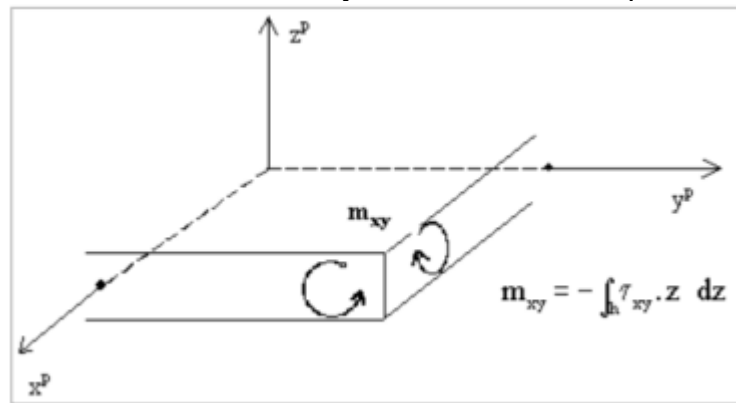
Fonte: SCIA ENGINEER. Basic Concept Training

Figura 5 – Momentos fletores m_x e m_y dos elementos de placa



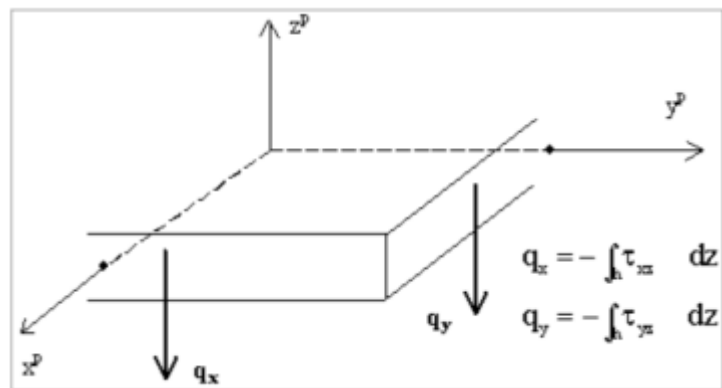
Fonte: SCIA ENGINEER. Basic Concept Training

Figura 6 – Momento torsor m_{xy} dos elementos de placa ou casca



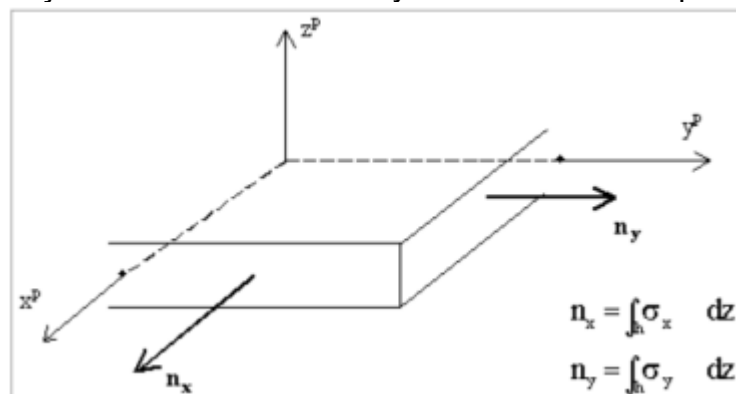
Fonte: SCIA ENGINEER. Basic Concept Training

Figura 7 – Forças de cisalhamento q_x e q_y ($= v_x$ e v_y) dos elementos de placa ou casca



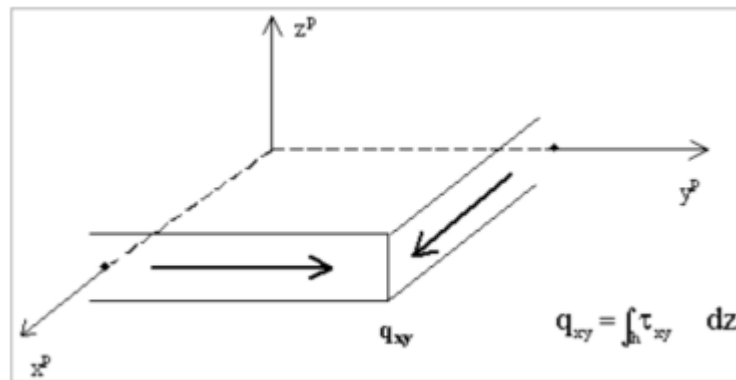
Fonte: SCIA ENGINEER. Basic Concept Training

Figura 8 – Forças de membrana n_x e n_y dos elementos de parede ou casca



Fonte: SCIA ENGINEER. Basic Concept Training

Figura 9 – Força de cisalhamento q_{xy} ($= nxy$) dos elementos de parede ou casca



Fonte: SCIA ENGINEER. Basic Concept Training.

3 SCIA ENGINEER

O *SCIA Engineer* é um *software* avançado de análise e projeto estrutural desenvolvido pela Nemetschek SCIA, amplamente utilizado na engenharia civil. Ele oferece uma plataforma integrada para modelagem, análise e design de estruturas, suportando diversos materiais como aço, concreto, madeira, alumínio e materiais compostos. O *software* é conhecido por sua interface amigável e sua capacidade de se integrar com outros *softwares* de Modelagem de Informações da Construção (BIM), o que o torna popular entre engenheiros estruturais, empresas de projeto e instituições acadêmicas.

Dentre suas principais funcionalidades, o *SCIA Engineer* utiliza o método dos elementos finitos, permitindo análises estruturais complexas. Além disso, o *software* inclui recursos avançados como análise não linear, análise dinâmica, avaliação de flambagem e análise de fadiga. É utilizado em uma variedade de aplicações, incluindo o projeto de edifícios, pontes e estruturas industriais e especiais. Para a definição de materiais, seções transversais e combinações de cargas o *SCIA Engineer* possui compatibilidade com uma vasta biblioteca de normas nacionais e internacionais, como por exemplo a Associação Brasileira de Normas Técnicas (ABNT), a *International Building Code* (IBC) e as *EN Eurocodes*.

3.1 INTERFACE GRÁFICA

A interface gráfica do usuário (GUI) do *SCIA Engineer* é fundamental para a experiência e eficiência dos usuários no *software* de análise e projeto estrutural. Sua

interface, inspirada no estilo da Microsoft Office, apresenta uma faixa de opções que organiza ferramentas e funções de maneira lógica, além de navegação por abas que facilita a alternância entre diferentes fases do projeto. Esta GUI oferece um ambiente de modelagem 3D interativo, equipado com ferramentas avançadas para modelagem precisa. Ela também inclui uma estrutura de árvore de projeto para organizar componentes, e ferramentas para filtragem e seleção, otimizando a gestão de modelos complexos. A visualização de resultados é clara e personalizável, permitindo aos engenheiros interpretar facilmente dados como diagramas de força e deformações.

A GUI do *SCIA Engineer* suporta integralmente a integração com ferramentas BIM, facilitando a importação e exportação de modelos para diferentes *softwares*, e promove a colaboração eficiente entre equipes. A personalização da interface é possível, permitindo aos usuários adaptá-la às suas preferências, e a extensibilidade através do *OpenAPI* oferece oportunidades para integrar funcionalidades personalizadas e automação. Em resumo, a interface do *SCIA Engineer* é projetada para maximizar a produtividade, combinando navegação intuitiva, poderosas ferramentas de modelagem, visualizações avançadas de resultados e recursos de colaboração, tornando-se uma plataforma robusta para o enfrentamento de desafios de projeto em um ambiente digital e colaborativo.

3.2 OPENAPI

O *OpenAPI* do *SCIA Engineer* é uma *Application Programming Interface* (API) poderosa que permite a interação entre programas desenvolvidos pelo usuário e o *SCIA Engineer*, possibilitando tarefas como a automação de rotinas e o projeto iterativo para otimização estrutural. Essa ferramenta é separada em 3 (três) grupos que englobam diversas funcionalidades do *SCIA Engineer*. Estes grupos estão descritos nos itens a seguir.

3.2.1 Analysis data model

O *Analysis Data Model* (ADM) é uma parte crucial do *OpenAPI* do *SCIA Engineer*. É o pacote de códigos voltado para a modelagem estrutural, isto é, criação

de elementos como material, seção transversal, nó estrutural, barra 1D, entre outros. Além disso, é possível encontrar elementos já existentes no modelo e modificá-los.

Esta vertente do *OpenAPI* possui uma versão simplificada, contendo os elementos de modelagem mais comumente utilizados, sendo a versão cujo presente trabalho utiliza, e uma versão complexa, na qual o desenvolvedor tem a possibilidade de trabalhar com uma gama maior de funcionalidades dentro do *SCIA Engineer*.

3.2.2 Results api

A *Results API* (RAPI) é focada na manipulação e análise dos resultados gerados pelo *SCIA Engineer*. Permite extrair dados completos de resultados, como deformações, tensões, forças internas, etc. Além disso, os dados podem ser extraídos de um elemento da malha de elementos finitos específico, proporcionando uma grande liberdade para o desenvolvedor ao analisar resultados pela API de Resultados.

3.2.3 Esa automation

EsaAutomation é uma plataforma dentro do *OpenAPI* do *SCIA Engineer* que permite o controle remoto do ambiente do *SCIA Engineer*. Permite automatizar uma série de tarefas, como iniciar o *SCIA Engineer* em segundo plano, abrir um modelo de projeto, calcular um projeto, importar dados de XML, entre outros. Apesar de ser o menor dos grupos, possui grande importância por proporcionar a criação e o descarte do ambiente no qual o *SCIA Engineer* é executado, servindo como uma ferramenta intermediária obrigatória para o devido funcionamento do *OpenAPI* em geral.

4 MODELO

O modelo do presente trabalho será composto de três grupos, os elementos, as ações e as combinações de ações. O objetivo da criação desses grupos é de simular o funcionamento de uma estrutura portuária *offshore*, mais especificamente, um *dolphin* de amarração, como descrito no item 2.

4.1 ELEMENTOS


Elementos dentro do *SCIA Engineer* representam as partes físicas do modelo. Os itens abaixo descrevem os elementos utilizados para o modelo do *dolphin* de amarração.

4.1.1 Materiais

Os materiais no *SCIA Engineer* são definidos por suas propriedades físicas e mecânicas, como módulo de elasticidade e resistência à compressão. A escolha do material é crucial para determinar o comportamento da estrutura.


No modelo, foram utilizados os materiais C40 para o concreto, CA-25 para a armadura do concreto e A572 grau 50 para o aço. Da figura 10 até a figura 12 estão definidas as propriedades desses materiais dentro do *SCIA Engineer*, seguindo às normas ABNT NBR 6118:2023 e NBR 8800:2008.

Figura 10 – Propriedades do concreto C40

Name	C40
Code independent	
Material type	Concrete
Thermal expansion [m/mK]	0,00
Unit mass [kg/m ³]	2500,0
Density in fresh state [kg/m ³]	2600,0
Ecs modulus [MPa]	3,1876e+04
Poisson coeff.	0,2
Independent G modulus	<input type="checkbox"/>
G modulus [MPa]	1,3282e+04
Log. decrement (non-uniform dampin	0,062
Colour	
Specific heat [J/g°C]	6,0000e-01
Thermal conductivity [W/mK]	4,5000e+01
Order in code	5
Price per unit [€/m ³]	1,00
ABNT NBR 6118	
Characteristic compressive cylinder str	40,00
Calculated depended values	<input checked="" type="checkbox"/>
Mean tensile strength fctm(28) [MPa]	3,51
fctk,inf [MPa]	2,46
fctk,sup [MPa]	4,56
Design compressive strength - regular	28,57
Design compressive strength - special	33,33
Design compressive strength - exceptic	33,33
Strain at reaching maximum strength	20,0
Ultimate strain eps cu [1e-4]	35,0
Stone diameter (dg) [mm]	30
Type of aggregate	Granite

Fonte: SCIA Engineer e ABNT NBR 6118:2023

Figura 11 – Propriedades do aço da armadura CA-25

Name CA-25	
Code independent	
Material type	Reinforcement steel
Thermal expansion [m/mK]	0,00
Unit mass [kg/m ³]	7850,0
E modulus [MPa]	2,1000e+05
Poisson coeff.	0,2
Independent G modulus	<input type="checkbox"/>
G modulus [MPa]	8,7500e+04
Log. decrement (non-uniform dampin	0,062
Colour	
Specific heat [J/gK]	6,0000e-01
Thermal conductivity [W/mK]	4,5000e+02
Bar surface	Smooth
Order in code	1
Price per unit [€/kg]	1,00
ABNT NBR 6118	
Characteristic yield strength fyk [MPa]	250,0
Calculated depended values	<input type="checkbox"/>
Design yield strength - regular (fyd = fy)	217,4
Design yield strength - specials (fyd = f	217,4
Design yield strength - exceptional (fyc	250,0
Maximum elongation eps uk [1e-4]	100,0

Fonte: SCIA Engineer e ABNT NBR 6118:2023

Figura 12 – Propriedades do aço A572 grau 50

Name A572 grade 50	
Code independent	
Material type	Steel
Thermal expansion [m/mK]	0,00
Unit mass [kg/m ³]	7850,0
E modulus [MPa]	2,0000e+05
Poisson coeff.	0,3
Independent G modulus	<input checked="" type="checkbox"/>
G modulus [MPa]	7,6923e+04
Log. decrement (non-uniform dampin	0,15
Colour	
Thermal expansion (for fire resistance	0,00
Specific heat [J/g°C]	0,0000e+00
Thermal conductivity [W/mK]	0,0000e+00
Price per unit [€/kg]	1,00
ABNT NBR 8800	
Ultimate strength [MPa]	448,0
Yield strength [MPa]	345,0

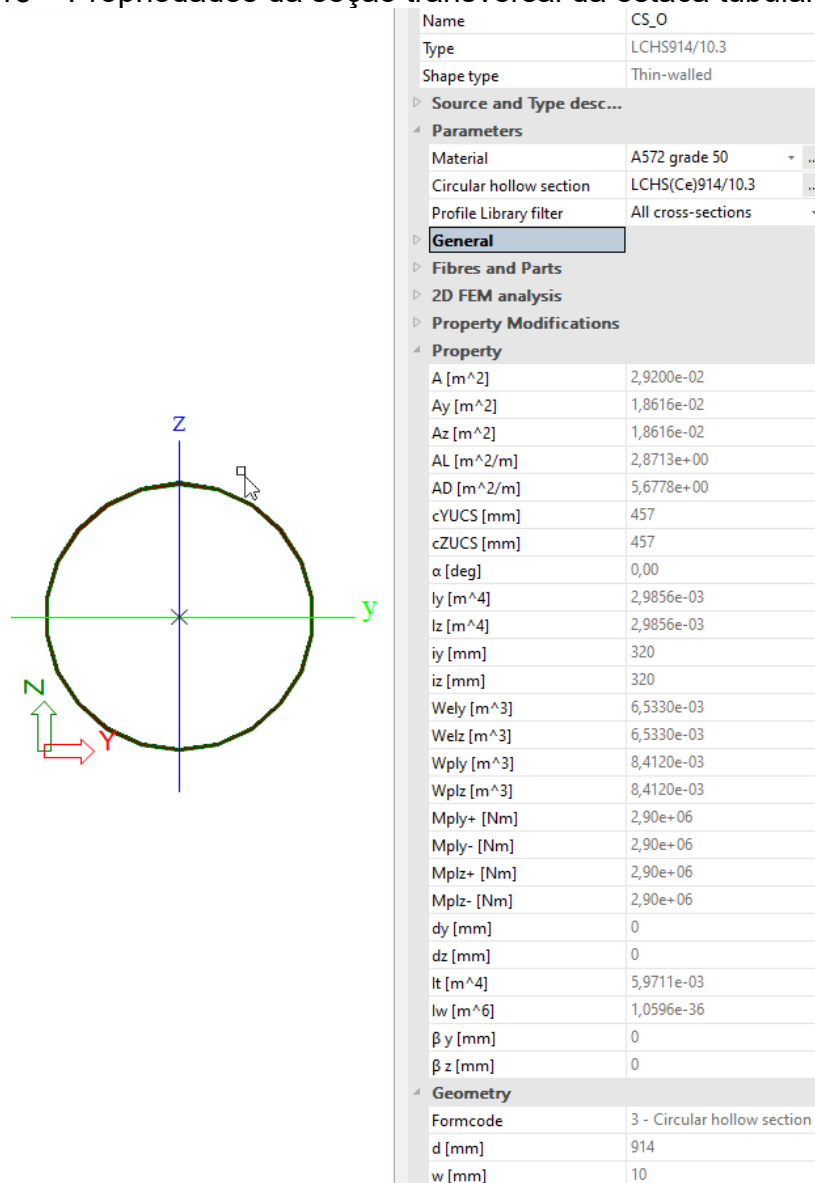
Fonte: SCIA Engineer e ABNT NBR 8800:2008

4.1.2 Seções transversais

As seções transversais definem a forma geométrica e o material da seção de um elemento estrutural. No *SCIA Engineer*, as seções podem ser selecionadas de uma biblioteca predefinida ou definidas de forma customizada, sendo essenciais para a análise de resistência e estabilidade dos elementos estruturais.

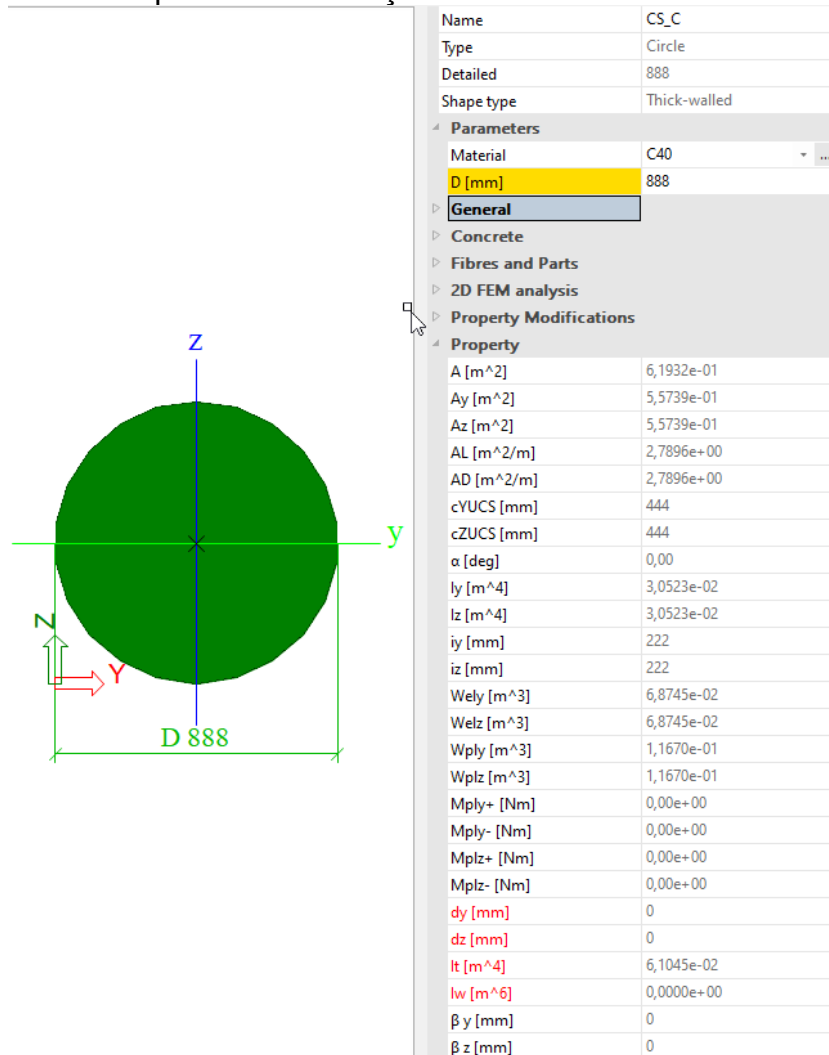
No modelo, foram utilizadas as seções transversais de estaca tubular metálica e de estaca concretada, com suas respectivas propriedades evidenciadas nas figura 13 e na figura 14.

Figura 13 – Propriedades da seção transversal da estaca tubular metálica



Fonte: *SCIA Engineer*

Figura 14 – Propriedades da seção transversal da estaca concretada



Fonte: SCIA Engineer

4.1.3 Nós

Os nós são elementos cruciais em qualquer modelo estrutural no SCIA Engineer, atuando como pontos de conexão para barras, lajes e outros elementos estruturais. Eles são os locais onde as cargas e restrições são aplicadas, desempenhando um papel fundamental na definição da geometria e na transferência de forças dentro da estrutura. Cada nó é definido a partir de suas coordenadas X, Y e Z.

4.1.4 Estacas

As barras, ou vigas, são elementos estruturais lineares que resistem a cargas principalmente através de flexão. No *SCIA Engineer*, são modeladas conectando-se dois nós e atribuindo-se propriedades específicas como material, seção transversal e condições de contorno, sendo essenciais para a análise de esforços internos como momentos e forças cortantes.

Para o atual modelo, serão definidas quatro estacas simétricas que conectam a laje ao solo, funcionando como fundações profundas. Para cada estaca será necessário criar duas barras dentro do modelo, sendo uma com a seção transversal tubular metálica e a outra com a concretada, totalizando oito barras. Posteriormente, essas barras serão conectadas durante a execução do cálculo da estrutura.

4.1.5 Laje

Lajes são elementos estruturais planos que resistem a cargas em seu próprio plano. No *software*, são modeladas definindo-se sua geometria, espessura, material e condições de apoio. As lajes são analisadas para distribuição de cargas, momentos, forças cortantes e deformações.

Para o modelo foi definido uma laje, composta do concreto C40, com armaduras de aço CA-25.

4.1.6 Cabeços de amarração

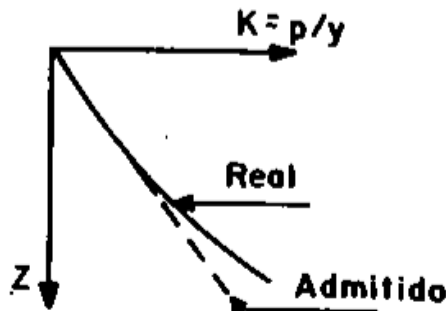
Os cabeços de amarração são equipamentos instalados na laje que permitem a amarração do navio atracado. Para simular a aplicação da força de amarração dentro do modelo serão criadas duas barras de seção transversal de concreto sobre a laje.

4.1.7 Apoios

Os apoios no *SCIA Engineer* podem ser de linha ou de ponto. Apoios de linha são usados para modelar apoios ao longo de uma linha, enquanto apoios de ponto são aplicados em nós específicos. Eles podem variar entre fixos, móveis ou elásticos, dependendo das condições de contorno estabelecidas.

A estaca tubular metálica possui uma parcela cravada. Devido ao comportamento estrutural dos *dolphins* de amarração, evidenciado no item 2, as reações do solo de maior importância serão nas direções horizontais. Para simular essas reações horizontais do solo, utiliza-se o modelo proposto por Alonso (1989, p. 70-71), que admite o comportamento flexível, com a variação do módulo de reação horizontal (K) crescendo linearmente com a profundidade.

Figura 15 – Módulo de reação horizontal (K) variando com a profundidade



Fonte: Alonso (1989, p. 70)

O programa não possui ferramentas para aplicar este modelo de variação linear de rigidez em um apoio de linha. Portanto, para simular este comportamento, dividiu-se a parcela cravada da estaca (em contato com o solo) em oito apoios de linha de mesmo comprimento. Por conta dessa adaptação, o valor fornecido de rigidez, em N/m, deve ser ajustado, dividindo-se pelo comprimento do apoio de linha, sendo, assim, um módulo de reação horizontal em N/m². Portanto, cada apoio de linha terá rigidez constante, cujo valor aumenta de um apoio para o seguinte, visto que esse apoio seguinte se encontra em uma profundidade maior. Para a extremidade, é utilizado um apoio de ponto, cuja rigidez se dá em coeficiente de mola, na unidade N/m.

Todos os apoios possuem rotação livre, restrições horizontais flexíveis e translação livre no eixo Z, com exceção do apoio da extremidade, cuja translação no eixo Z é restringida. A figura 16 mostra as propriedades do primeiro apoio de linha e a figura 17 mostra as propriedades do apoio de ponto da extremidade.

Figura 16 – Restrições do primeiro apoio de linha da estaca

Name	ApoioE0_1
Type	Line
Constraint	Custom
X	Flexible
Stiffness X [MN/m ²]	66,7875
Y	Flexible
Stiffness Y [MN/m ²]	66,7875
Z	Free
Rx	Free
Ry	Free
Rz	Free

Fonte: SCIA Engineer

Figura 17 – Restrições do apoio de ponto da extremidade da estaca

Name	ApoioE0_0
Type	Standard
Angle [deg]	
Constraint	Custom
X	Flexible
Stiffness X [MN/m]	75,4050
Y	Flexible
Stiffness Y [MN/m]	75,4050
Z	Rigid
Rx	Free
Ry	Free
Rz	Free

Fonte: SCIA Engineer

4.2 AÇÕES

O *SCIA Engineer* permite a aplicação de diversos tipos de ações, incluindo o peso próprio da estrutura, ações distribuídas em superfícies (como lajes) e ações pontuais, que são concentradas e aplicadas em pontos específicos.

Casos de ações representam diferentes cenários de carregamento, divididos entre variáveis, permanentes e de temperatura. No *SCIA Engineer*, diferentes casos

de ações são definidos para simular os efeitos variados das cargas ao longo da vida útil da estrutura.

Para o modelo, será considerada a ação permanente de peso próprio e as ações variáveis de sobrecarga, corrente e amarração.

4.2.1 Peso próprio

Representa o peso da própria estrutura e é calculado automaticamente pelo *software* com base nas propriedades dos materiais.

4.2.2 Sobrecarga

Consiste em uma ação de superfície aplicada sobre a laje. De acordo com a NBR 9782:1987, as sobrecargas são ações de natureza não definida que podem ocorrer durante a vida útil da estrutura. Ainda, segundo a tabela 1 da norma, para estruturas fixas sem sobrecarga definida deve-se utilizar o valor de 10 kN/m².

Apesar de Thoresen (2014, p. 173), como uma referência internacional, indicar que o valor de sobrecarga para *dolphins* de amarração deve ser de 3 kN/m², os modelos gerados neste trabalho utilizarão 10 kN/m². Mesmo que a NBR 9782:1987 tenha sido cancelada, trata-se da única norma Brasileira que possui essa recomendação, e, complementando, é um valor de sobrecarga maior, portanto, a favor da segurança.

4.2.3 Corrente

Ação de linha aplicada na porção submersa da estaca. Para o modelo foram definidos quatro casos, sendo eles: direção longitudinal e transversal, assim como seus respectivos sentidos positivos e negativos.

4.2.4 Amarração

A ação de amarração corresponde à ação realizada pela corda que amarra o navio ao *dolphin*. A maior porção dessa força se encontra no sentido horizontal,

impedindo a movimentação do navio ao ser perturbado pelas variações de corrente, onda e vento. Portanto, serão criadas ações de ponto horizontais aplicadas na extremidade superior do cabeço de amarração. Para o modelo foram considerados dez possíveis ângulos de atuação da força em cada cabeço.

4.3 COMBINAÇÕES DE AÇÕES

Combinações de ações são combinações específicas de diferentes casos de ações, usadas para avaliar a resposta estrutural sob múltiplas condições de carregamento simultâneas.

Para o modelo foram geradas as combinações de Estado Limite Último e de Estado Limite de Serviço. Da tabela 1 até a tabela 4 especificam-se os coeficientes aplicados em cada caso de ação, cujos valores foram definidos de acordo com as tabelas 11.1 a 11.4 da NBR 6118:2023.

4.3.1 Estado limite último

A tabela 1 contém as dez possíveis combinações para o Estado Limite Último.

Tabela 1 – Combinações de Estado Limite Último (ELU)

Combinações	Peso Próprio	Sobrecarga	Corrente	Cabeço de amarração 1	Cabeço de amarração 2
ELU 1	1,4	-	-	-	-
ELU 2	1,0	-	-	-	-
ELU 3	1,4	1,05	1,05	1,50	1,05
ELU 4	1,4	1,05	1,05	1,05	1,50
ELU 5	1,4	1,05	1,50	1,05	1,05
ELU 6	1,4	1,50	1,05	1,05	1,05
ELU 7	1,0	1,05	1,05	1,50	1,05
ELU 8	1,0	1,05	1,05	1,05	1,50
ELU 9	1,0	1,05	1,50	1,05	1,05
ELU 10	1,0	1,50	1,05	1,05	1,05

Fonte: Elaborado pelo autor.

4.3.2 Estado limite de serviço

4.3.2.1 Quase permanentes

A tabela 2 contém as duas possíveis combinações para o Estado Limite de Serviço Quase Permanentes.

Tabela 2 – Combinações de Estado Limite de Serviço Quase Permanentes (ELSQ)

Combinações	Peso Próprio	Sobrecarga	Corrente	Cabeço de amarração 1	Cabeço de amarração 2
ELSQ 1	1,0	-	-	-	-
ELSQ 2	1,0	0,4	0,4	0,4	0,4

Fonte: Elaborado pelo autor.

4.3.2.2 *Frequentes*

A tabela 3 contém as cinco possíveis combinações para o Estado Limite de Serviço Frequentes.

Tabela 3 – Combinações de Estado Limite de Serviço Frequentes (ELSF)

Combinações	Peso Próprio	Sobrecarga	Corrente	Cabeço de amarração 1	Cabeço de amarração 2
ELSF 1	1,0	-	-	-	-
ELSF 2	1,0	0,4	0,4	0,6	0,4
ELSF 3	1,0	0,4	0,4	0,4	0,6
ELSF 4	1,0	0,4	0,6	0,4	0,4
ELSF 5	1,0	0,6	0,4	0,4	0,4

Fonte: Elaborado pelo autor.

4.3.2.3 *Raras*

A tabela 4 contém as cinco possíveis envoltórias para o Estado Limite de Serviço Frequentes.

Tabela 4 – Combinações de Estado Limite de Serviço Raras (ELSR)

Combinações	Peso Próprio	Sobrecarga	Corrente	Cabeço de amarração 1	Cabeço de amarração 2
ELSR 1	1,0	-	-	-	-
ELSR 2	1,0	0,6	0,6	1,0	0,6
ELSR 3	1,0	0,6	0,6	0,6	1,0
ELSR 4	1,0	0,6	1,0	0,6	0,6
ELSR 5	1,0	1,0	0,6	0,6	0,6

Fonte: Elaborado pelo autor.

5 PARÂMETROS DO MODELO

Para construir o modelo é necessário definir dezenove parâmetros. Esses parâmetros foram divididos em duas categorias: parâmetros geométricos e os demais parâmetros.

5.1 PARÂMETROS GEOMÉTRICOS

A tabela 5 apresenta os treze parâmetros geométricos do modelo.

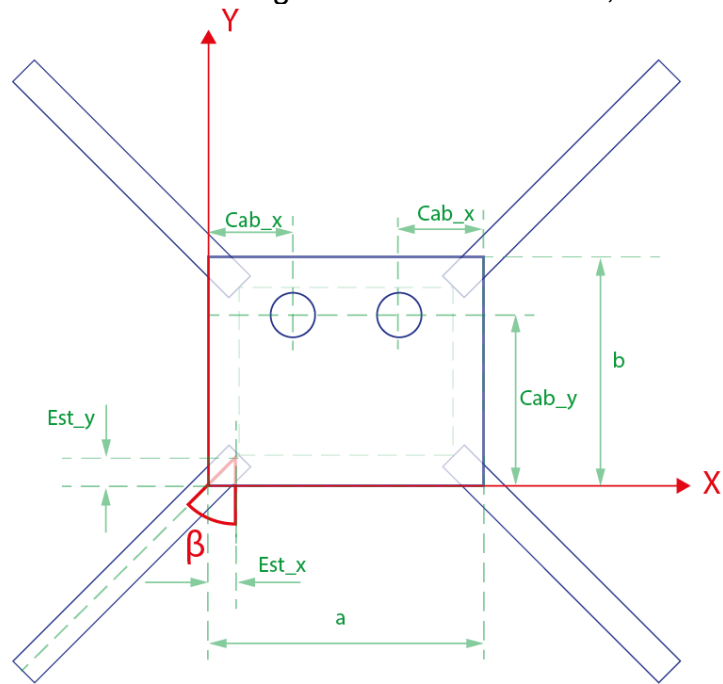
Tabela 5 – Parâmetros geométricos do modelo

Parâmetro	Descrição
a	Largura da laje em metros, apresentada no eixo X
b	Comprimento da laje em metros, apresentada no eixo Y
c	Espessura da laje em metros, apresentada no eixo Z
α	Ângulo de inclinação da estaca com relação à sua projeção no plano XY
β	Ângulo de rotação da projeção da estaca no plano XY em relação ao eixo X
Cab_x	Distância do eixo central de cada cabeço em relação às bordas paralelas ao eixo Y
Cab_y	Distância do eixo central de cada cabeço em relação ao eixo X ($b/2 \leq \text{Cab}_y < b$)
Est_x	Distância do centro da estaca no plano XY em relação às bordas paralelas ao eixo X
Est_y	Distância do centro da estaca no plano XY em relação às bordas paralelas ao eixo Y
h_{solo}	Profundidade de solo em que a extremidade da estaca se apoia
$h_{\text{laje_solo}}$	Altura da laje (plano XY) com relação ao solo
$h_{\text{nível_água}}$	Altura máxima do nível da água (utilizado para o cálculo das forças de corrente)
$h_{\text{concretada}}$	Altura da estaca a ser concretada

Fonte: Elaborado pelo autor

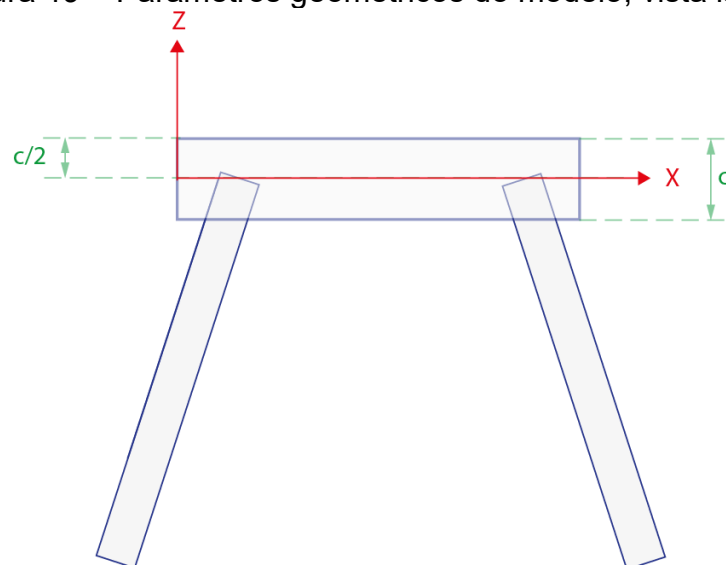
Para a melhor compreensão desses parâmetros, foram elaboradas a figura 18, a figura 19, a figura 20 e a figura 21.

Figura 18 – Parâmetros geométricos do modelo, vista superior



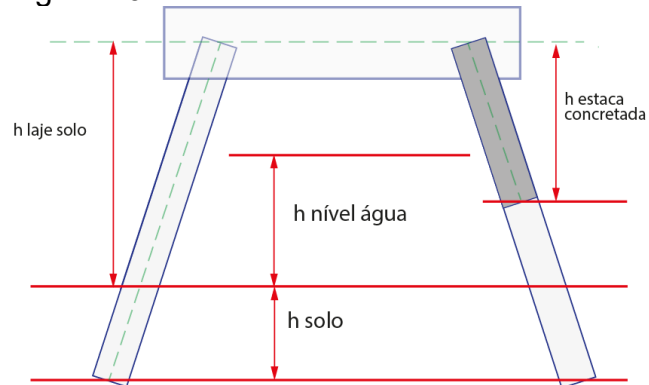
Fonte: Elaborado pelo autor

Figura 19 – Parâmetros geométricos do modelo, vista lateral

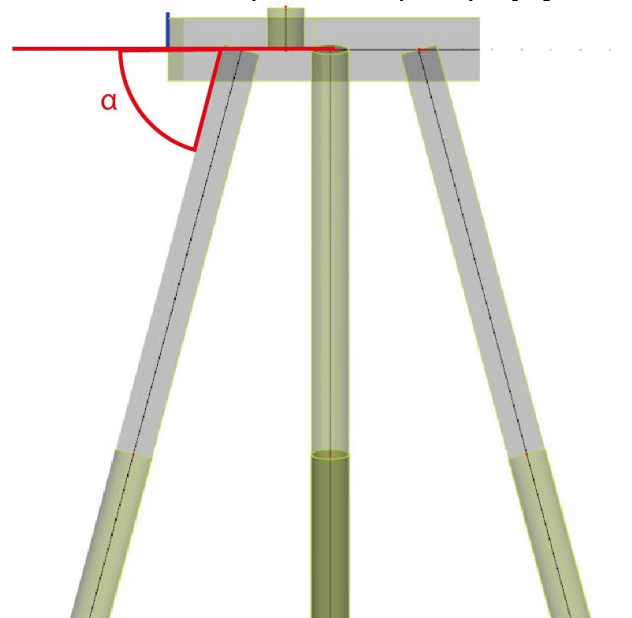


Fonte: Elaborado pelo autor

Figura 20 – Parâmetros de altura do modelo



Fonte: Elaborado pelo autor

Figura 21 – Ângulo α , corte vertical passando pela projeção da estaca no plano XY

Fonte: Elaborado pelo autor

5.2 DEMAIS PARÂMETROS

A tabela 6 apresenta os outros seis parâmetros do modelo.

Tabela 6 – Demais parâmetros do modelo

Parâmetro	Descrição
Rigidez _{inicial}	Valor inicial da rigidez do solo, em N/m
Rigidez _{final}	Valor final da rigidez do solo, em N/m
Sobrecarga	Valor da ação de sobrecarga, em N/m ² (10 kN/m ² conforme 4.2.2)
Força _{Corrente_transV}	Valor da ação de corrente na direção transversal (paralela ao eixo X), em N
Força _{Corrente_long}	Valor da ação de corrente na direção longitudinal (paralela ao eixo Y), em N
Força _{Cabeço}	Valor da ação de amarração do navio, em N

Fonte: Elaborado pelo autor

5.3 VALORES CALCULADOS

Os valores calculados são aqueles considerados relevantes para o modelo, decorrentes dos parâmetros definidos em 5.1 e 5.2.

Tabela 7 – Valores calculados a partir dos parâmetros

Valor calculado	Descrição
h_{total}	Altura total da estaca
$Comp_Estaca_{Total}$	Comprimento total da estaca
$Comp_Estaca_{ProjXY}$	Comprimento da projeção da estaca no plano XY
$Comp_Estaca_x$	Comprimento da projeção da estaca no eixo X
$Comp_Estaca_y$	Comprimento da projeção da estaca no eixo Y
$Comp_Estaca_z$	Comprimento da projeção da estaca no eixo Z
$Comp_Estaca_{Concretada}$	Comprimento da porção concretada da estaca
$Comp_Estaca_{Metálica}$	Comprimento da porção metálica da estaca
$Comp_Estaca_{Solo}$	Comprimento da estaca cravada no solo
$Comp_Estaca_{Submersa}$	Comprimento submerso da estaca
$Comp_Estaca_{Metálica_Submersa}$	Comprimento submerso da porção metálica da estaca
$Comp_Estaca_{Concretada_Submersa}$	Comprimento submerso da porção concretada da estaca

Fonte: Elaborado pelo autor

A seguir serão apresentadas as equações de definição desses valores calculados:

$$h_{total} = h_{solo} + h_{\acute{a}gua_plataforma} \quad (1)$$

$$Comp_Estaca_z = h_{total} \quad (2)$$

$$Comp_Estaca_z = Comp_Estaca_{Total} \times \sin \alpha \quad (3)$$

$$Comp_Estaca_{ProjXY} = Comp_Estaca_{Total} \times \cos \alpha \quad (4)$$

$$Comp_Estaca_x = Comp_Estaca_{ProjXY} \times \cos \beta \quad (5)$$

$$Comp_Estaca_y = Comp_Estaca_{ProjXY} \times \sin \beta \quad (6)$$

$$h_{concretada} = Comp_Estaca_{Concretada} \times \sin \beta \quad (7)$$

$$Comp_Estaca_{Metálica} = Comp_Estaca_{Total} - Comp_Estaca_{Concretada} \quad (8)$$

$$h_{solo} = Comp_Estaca_{Solo} \times \sin \beta \quad (9)$$

$$h_{nível_água} = Comp_Estaca_{Submersa} \times \sin \beta \quad (10)$$

$$Comp_Estaca_{Metálica_Submersa} = Comp_Estaca_{Metálica} - Comp_Estaca_{Solo} \quad (11)$$

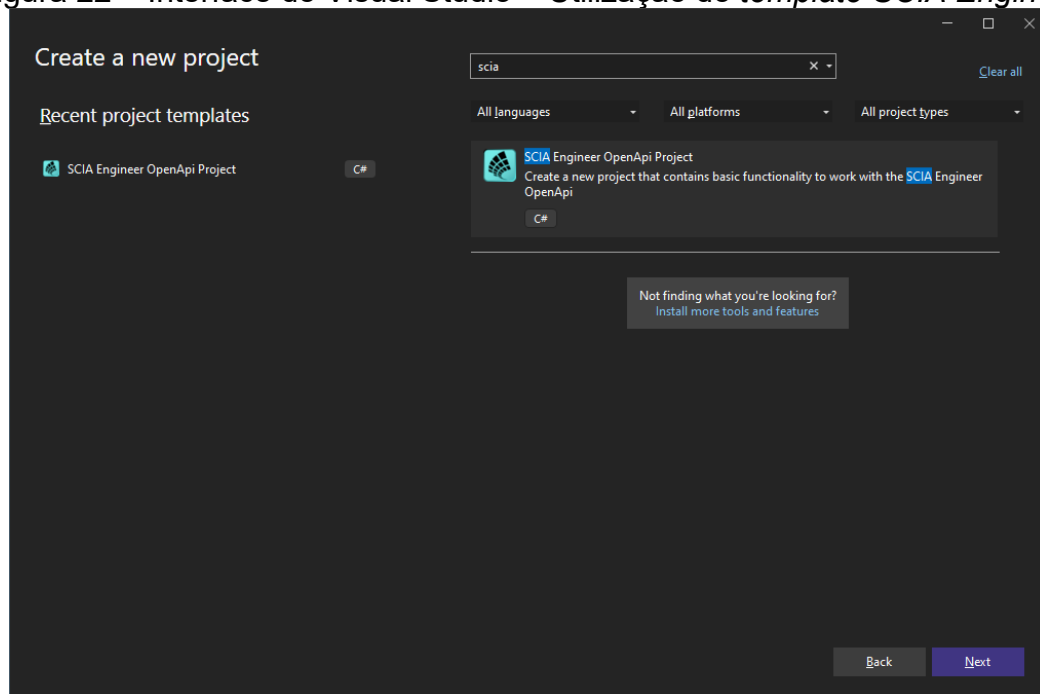
$$Comp_Estaca_{Concretada_Submersa} = Comp_Estaca_{Submersa} - Comp_Estaca_{Metálica_Submersa} \quad (12)$$

6 PROGRAMA E SCIA OPENAPI¹

No desenvolvimento desse programa foram utilizados os programas *Visual Studio 2022 – Community version*, utilizando a linguagem de programação C# (C sharp), bem como a biblioteca *OpenAPI* do programa *SCIA Engineer*, versão 2022.1.2011.

Para a definição dos parâmetros foi utilizado o *template* do *SCIA Engineer* para o Visual Studio 2022, nos termos da figura abaixo.

Figura 22 – Interface do Visual Studio – Utilização do *template SCIA Engineer*



Fonte: *Visual Studio 2022*

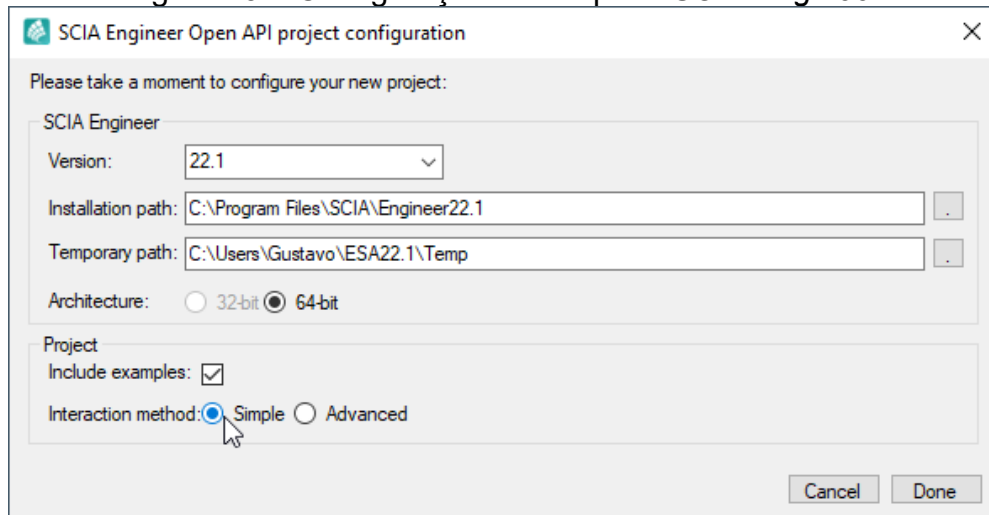
Na sequência da criação do modelo do *SCIA* no *Visual Studio*, é escolhido o método simples ou avançado. No método simples é utilizada a *OpenAPI*, como mencionado no item 3.2.1. Caso optasse pelo avançado, seriam referenciadas as bibliotecas do *Analysis Data Model (ADM)*². Nesse modelo avançado, as definições dos objetos envolvem uma complexidade maior, sendo necessário definir mais parâmetros para cada elemento do *OpenAPI*, adicionando dificuldade à elaboração e e à interpretação do programa.

¹ A íntegra do programa está disponível no *GitHub*: <https://github.com/AltTabo/Mooring-dolphin-SCIA-OpenAPI>

² Para a utilização do modelo ADM, convém utilizar a documentação constante no seguinte link: <https://docs.calatrava.scia.net/html/9900284a-b4b6-405e-b157-630e24129caf.htm>.

Para fins deste trabalho, o qual busca demonstrar a automação utilizando as bibliotecas do *OpenAPI*, considera-se suficiente a utilização do modelo simplificado. Na figura abaixo, apresenta-se a caixa de diálogo para a escolha do modelo simplificado.

Figura 23 – Configuração do template *SCIA Engineer*

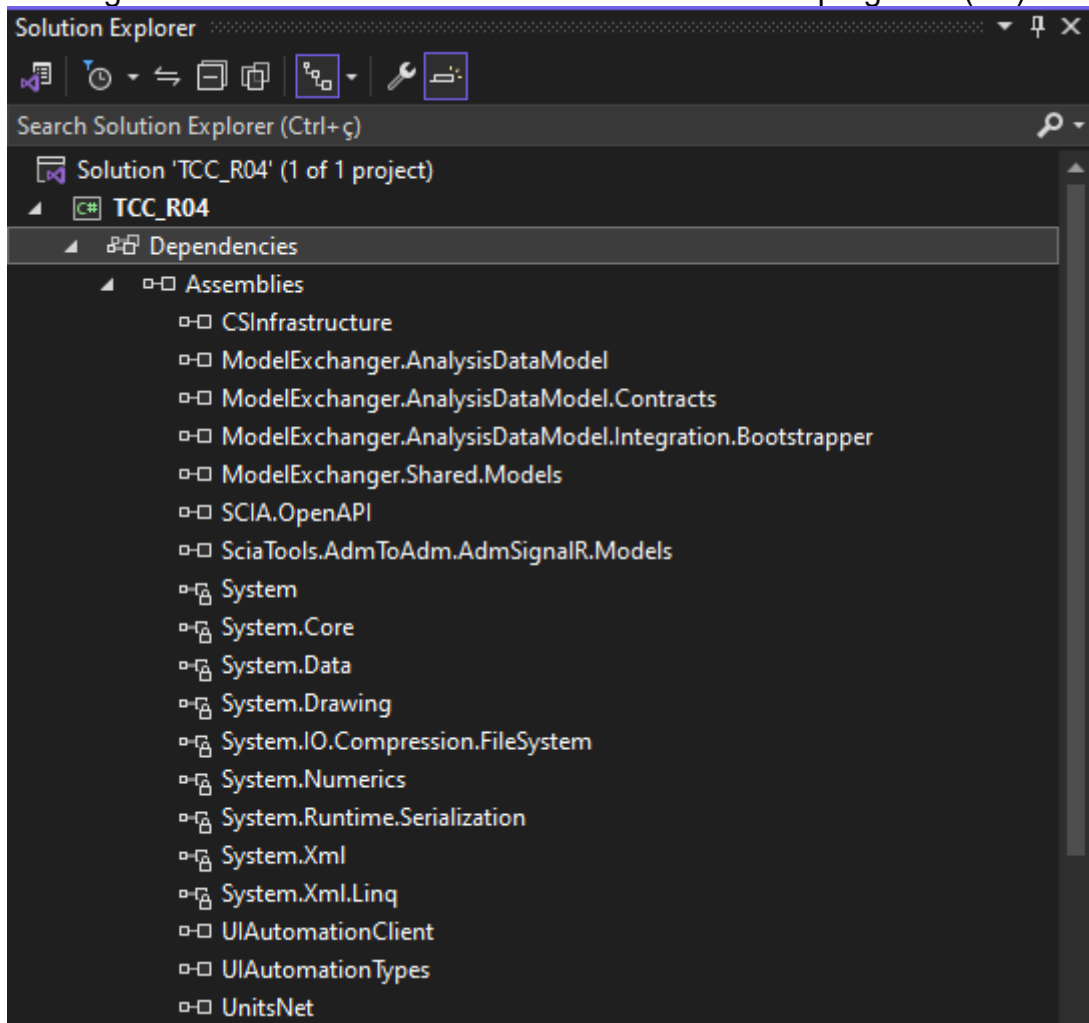


Fonte: *Visual Studio 2022*

O Visual Studio 2022, utilizando a linguagem C#, necessita acrescentar as referências às bibliotecas a serem utilizados pelo programa. Na figura abaixo verifica-se a biblioteca *SCIA.OpenAPI* referenciada pelo modelo simplificado, bem como as bibliotecas do modelo avançado ADM. Esse modelo avançado necessita de cinco bibliotecas para seu devido funcionamento, são elas:

- a) CSInfrastructure.dll;
- b) ModelExchanger.AnalysisDataModel.dll;
- c) ModelExchanger.AnalysisDataModel.Contracts.dll;
- d) ModelExchanger.AnalysisDataModel.Integration.Bootstrapper.dll;
- e) ModelExchanger.Shared.Models.dll.

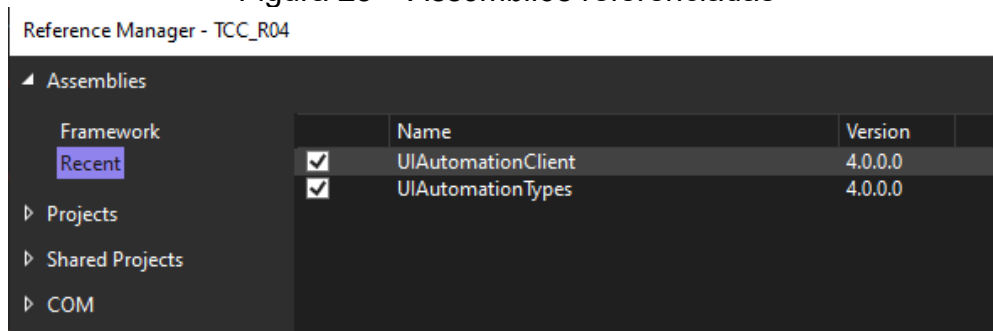
Figura 24 – Referências às bibliotecas utilizadas no programa (.dll)



Fonte: Visual Studio 2022

Além disso, para o funcionamento do programa desenvolvido, será necessário referenciar as seguintes *Assemblies*.

Figura 25 – *Assemblies* referenciadas



Fonte: Visual Studio 2022

Apresenta-se a seguir os comandos utilizados para definir os materiais do projeto. Também a forma de definição e criação dos nós, e a definição e criação da laje, a partir desses nós.

```

Material concreto = new Material(Guid.NewGuid(), "C40", 0, "C40");
Material aco = new Material(Guid.NewGuid(), "A572 grade 50", 1, "A572
grade 50");
foreach (var x in new List<Material> { concreto, aco }) {
model.CreateMaterial(x); }

StructNode node1 = new StructNode(Guid.NewGuid(), "Node1", 0, 0, 0);
StructNode node2 = new StructNode(Guid.NewGuid(), "Node2", var.A, 0, 0);
StructNode node3 = new StructNode(Guid.NewGuid(), "Node3", var.A,
var.B, 0);
StructNode node4 = new StructNode(Guid.NewGuid(), "Node4", 0, var.B, 0);
foreach (var x in new List<StructNode> { node1, node2, node3, node4 }) {
model.CreateNode(x); }

var slab = new Slab(Guid.NewGuid(), "Laje", 0, concreto.Id, var.C, new
Guid[4] { node1.Id, node2.Id, node3.Id, node4.Id });
model.CreateSlab(slab);

```

A seguir, é apresentado a definição das seções transversais da estaca tubular metálica e concretada, bem como suas respectivas criações. Também, mostra-se como exemplo a definição das barras, metálica e concretada, da primeira estaca, utilizando as seções transversais e os nós previamente definidos. Além de mostrar o comando de criação de todas as barras.

```

var cs_C = new CrossSectionParametric(Guid.NewGuid(), "CS_C",
concreto.Id, 0, diametro);
model.CreateCrossSection(cs_C);

var cs_O = new CrossSectionManufactured(Guid.NewGuid(), "CS_O",
aco.Id, "LCHS(Ce)914/10.3", 3, 103);
model.CreateCrossSection(cs_O);

Beam estaca1C = new Beam(Guid.NewGuid(), "Estaca1C", cs_C.Id, new
Guid[2] { nodeE1L.Id, nodeE1C.Id });
Beam estaca1O = new Beam(Guid.NewGuid(), "Estaca1O", cs_O.Id, new
Guid[2] { nodeE1C.Id, nodeE1.Id });
[...]
foreach (var x in new List<Beam> { estaca1C, estaca1O, estaca2C,
estaca2O, estaca3C, estaca3O, estaca4C, estaca4O }) {
model.CreateBeam(x); }

```

Apresenta-se na sequência o comando utilizado para definição dos apoios de linha na porção cravada das estacas. Essa porção é dividida em oito segmentos, nas quais possuem um valor constante no segmento e um valor variando linearmente

entre a rigidez inicial e a rigidez final, nos termos comentados no item 4.1.7. Esse comando se encontra dentro de um laço que percorre os oito segmentos, atribuindo os valores devidamente calculados para cada segmento (variável: rigidez). É elaborada uma matriz de apoio (ApoioE[i, j]) na qual o i representa cada uma das quatro estacas metálicas e o j cada um dos oito segmentos.

```
constraintz = eConstraintType.Free;
ApoioE[i, j] = new LineSupport(Guid.NewGuid(), nomeApoio,
estaca_apoio[i])
{
CoordinateDefinition = eCoordinateDefinition.Relative,
StartPoint = pontoinicial, EndPoint = pontofinal, Origin =
eLineOrigin.FromStart,
ConstraintX = eConstraintType.Flexible, StiffnessX = rigidez,
ConstraintY = eConstraintType.Flexible, StiffnessY = rigidez,
ConstraintZ = constraintz,
ConstraintRx = eConstraintType.Free,
ConstraintRy = eConstraintType.Free,
ConstraintRz = eConstraintType.Free
};
model.CreateLineSupport(ApoioE[i, j]);
```

Para o apoio de ponto da extremidade da estaca é utilizado o seguinte comando da biblioteca *OpenAPI*. É elaborada uma matriz de apoio (ApoioFinal[i]) na qual o i representa cada uma das quatro estacas metálicas. Esse apoio difere dos demais por ser rígido no eixo Z além de possuir a unidade N/m enquanto os demais apoios de linha são definidos em N/m².

```
constraintz = eConstraintType.Rigid;
ApoioFinal[i] = new PointSupport(Guid.NewGuid(), nomeApoio, no_apoio[i])
{
ConstraintX = eConstraintType.Flexible, StiffnessX = var.Rigidez_f,
ConstraintY = eConstraintType.Flexible, StiffnessY = var.Rigidez_f,
ConstraintZ = constraintz,
ConstraintRx = eConstraintType.Free,
ConstraintRy = eConstraintType.Free,
ConstraintRz = eConstraintType.Free
};
model.CreatePointSupport(ApoioFinal[i]);
```

Para definir as combinações de ações do modelo, primeiro definem-se os grupos de ações, exemplificado abaixo com a definição do grupo de ações permanentes LG1 e sua respectiva criação.

```
LoadGroup LG1 = new LoadGroup(Guid.NewGuid(), "LG1_PERMANENTE",
(int)eLoadGroup_Load.eLoadGroup_LoadPermanent);
```

```
[...]
foreach (var x in new List<LoadGroup> { LG1, LG2, LG3, LG4, LG5, LG6 }) {
model.CreateLoadGroup(x); }
```

Na sequência definem-se os casos de ação, como exemplificado abaixo para o caso do peso próprio (PP), inserido no grupo de ações permanentes. Em seguida, mostra-se o comando de criação de todos os casos, exceto para os 20 casos de amarração, que se encontram em um laço de repetição posteriormente.

```
LoadCase PP = new LoadCase(Guid.NewGuid(), "PP_PesoProprio",
(int)eLoadCase_ActionType.eLoadCase_ActionType_Permanent, LG1.Id,
0);
[...]
foreach (var x in new List<LoadCase> { PP, SC, RET, TEMP_MAIIS,
TEMP_MENOS, COR_LONG_MAIIS, COR_LONG_MENOS,
COR_TRANSV_MAIIS, COR_TRANSV_MENOS }) {
model.CreateLoadCase(x); }
```

Finalmente, definem-se os itens de combinação pertinentes para cada Estado Limite. Exemplifica-se abaixo o item de combinação do peso próprio para o Estado Limite Último (ELU). Todos esses itens são acrescentados a uma lista de combinações, que é utilizada para a criação da respectiva envoltória de ELU[k], nos termos da tabela 1.

```
combinationItems_ELU[0] = new CombinationItem(PP.Id, coef_perm);
[...]
ELU[k] = new Combination(Guid.NewGuid(), "ELU_" + p,
Lista_Combinacoes) { Category =
eLoadCaseCombinationCategory.AccordingNationalStandard,
NationalStandard = eLoadCaseCombinationStandard.IbclrfdUltimate };
model.CreateCombination(ELU[k]);
```

Para a criação da sobrecarga, mencionada no item 4.2.2, utilizaram-se os seguintes comandos.

```
SurfaceLoad sc = new SurfaceLoad(Guid.NewGuid(), "sobrecarga", -
var.Sobrecarga, SC.Id, slab.Id, 2);
model.CreateSurfaceLoad(sc);
```

Para a criação das ações de corrente, mencionadas no item 4.2.3, utilizaram-se os seguintes comandos, incluídos dentro de um laço. Com esses comandos em um laço, obtemos a matriz carga_corrente[i, j, k]. A dimensão i refere-se à estaca metálica ou estaca concretada. A dimensão j refere-se às quatro estacas. A dimensão

k refere-se às quatro combinações de direção (transversal e longitudinal) e sentido (positivo e negativo). Essas combinações geram um total de 32 (trinta e duas) ações de corrente.

```
carga_corrente[i, j, k] = new LineLoadOnBeam(Guid.NewGuid(), "COR_" + i
+ "_" + j + "_" + k)
{
    Member = estaca[i, j],
    LoadCase = corrente[k],
    Distribution = eLineLoadDistribution.Uniform,
    Value1 = forca[k],
    CoordinateDefinition = eCoordinateDefinition.Relative,
    StartPoint = 0,
    EndPoint = estaca_relativo[i],
    CoordinationSystem = eCoordinationSystem.GCS,
    Direction = direcao[k],
    Origin = origem[i],
    Location = eLineLoadLocation.Length,
    EccentricityEy = 0.0,
    EccentricityEz = 0.0
};
model.CreateLineLoad(carga_corrente[i, j, k]);
```

Para a criação das ações de amarração, mencionadas em 4.2.4, utilizaram-se os seguintes comandos. Esses comandos definiram a matriz carga_cabeco[i, j, k]. A dimensão i refere-se aos dois cabeços de amarração. A dimensão j refere-se aos dez ângulos estabelecidos (0°, 10°, 20°, 30°, 45°, 50°, 60°, 70°, 80° e 90°). A dimensão k refere-se às componentes X e Y da ação de amarração para cada respectivo ângulo, visto que havia a limitação do *OpenAPI* de gerar ações de ponto apenas na direção dos eixos principais.

```
carga_cabeco[i, j, 0] = new PointLoadInNode(Guid.NewGuid(), "FCABX_" + i
+ "_" + j, forca_x, CAB[i, j].Id, cabeco[i], (int)eDirection.X);
model.CreatePointLoadInNode(carga_cabeco[i, j, 0]);
[...]
carga_cabeco[i, j, 1] = new PointLoadInNode(Guid.NewGuid(), "FCABY_" + i
+ "_" + j, forca_y, CAB[i, j].Id, cabeco[i], (int)eDirection.Y);
model.CreatePointLoadInNode(carga_cabeco[i, j, 1]);
```

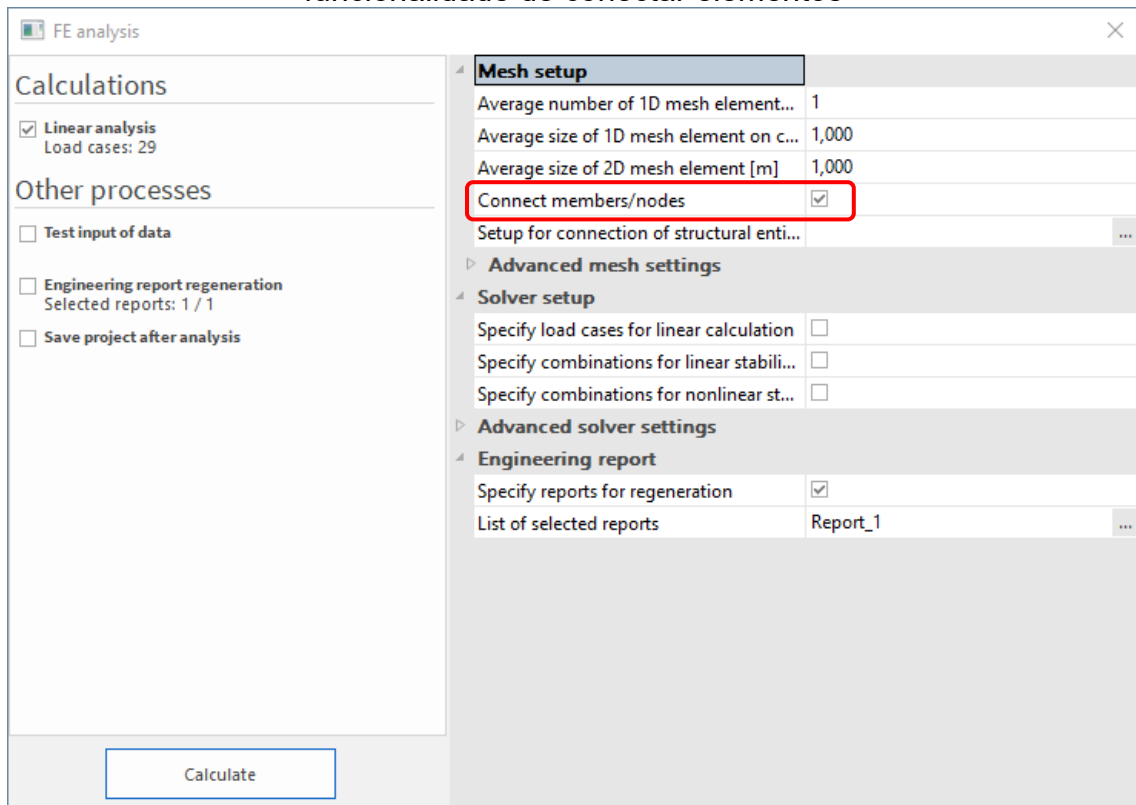
Após a definição dos elementos do modelo, das ações e das combinações de ações, é necessário executar o seguinte comando da *OpenAPI* para atualizar o modelo.

```
senData.Project.Model.RefreshModel_ToSCIAEngineer();
```


Agora o projeto está pronto para ser calculado. Verificou-se que para efetuar esse cálculo era necessário unir os elementos estaca e laje previamente. No entanto, a biblioteca *OpenAPI* não oferece recurso para efetuar tal procedimento. Aparentemente seria necessário redefinir todo o modelo utilizando o método de interação avançado das bibliotecas ADM.

Para contornar essa situação, percebeu-se que, ao efetuar o cálculo pela interface gráfica do *SCIA Engineer*, havia a opção de unir os elementos (vide figura 26).

Figura 26 – Interface gráfica de cálculo do *SCIA Engineer*, com destaque para a funcionalidade de conectar elementos

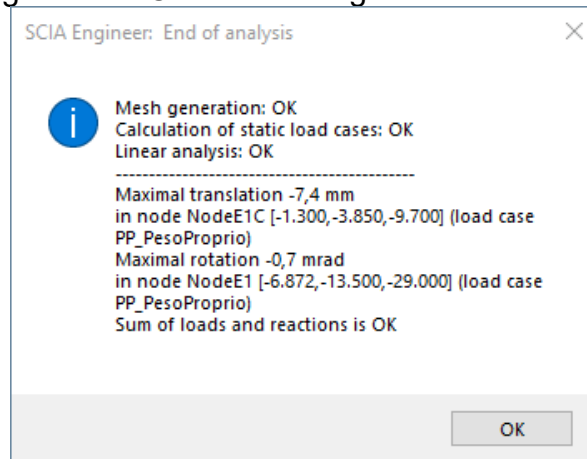


Fonte: *SCIA Engineer*

Assim, foi concebido um procedimento que, por meio de envios de comandos de teclado, permite ao programa desenvolvido em C# a abertura da caixa de diálogo de cálculo, a marcação da opção de conectar elementos e a realização do cálculo.

A seguir, é executado uma rotina em C# que, em processamento concorrente, detecta a abertura da seguinte caixa de diálogo.

Figura 27 – Caixa de diálogo de fim da análise



Fonte: *SCIA Engineer*

Ao detectar a caixa de diálogo, o programa encaminha a tecla “enter” para acionar o botão “OK”. Na sequência, é utilizado o seguinte comando para carregar para o programa C# as informações geradas no ambiente de interface gráfica do *SCIA Engineer*.

```
senData.Project.Model.RefreshModel_FromSCIAEngineer();
```

Para obter os resultados, são criadas três listas: a lista de envoltórias; a lista de combinações; e a lista de estacas. A lista de envoltórias refere-se às listas de combinações de cada tipo de Estados Limites (ELU, ELSF, ELSQ, ELSR). Para cada estado limite, por exemplo ELU, existem diversas combinações (ELU_1, ELU_2, ..., ELU_10). A lista de estacas refere-se às barras metálicas e às barras concretadas para cada uma das estacas. O laço a seguir percorre as três listas, produzindo o relatório que contém todas as informações para cada um desses casos. No final, a variável linha recebe esses dados que serão gravados em um arquivo texto.

Para o caso da laje, o laço percorrerá a lista de envoltórias e as suas respectivas combinações. Como há apenas uma laje, não há necessidade de percorrer uma terceira lista. Da mesma forma, as informações são armazenadas na variável linha, que é gravada no mesmo arquivo texto.

```
using (ResultsAPI resultsApi = senData.Project.Model.InitializeResultsAPI())
{
    [...]
    foreach(List<Combination> Envoltoria in Envoltorias)
    {
        foreach (Beam x in Lista_Estacas)
```

```

{
  foreach (Combination y in Envoltoria)
  {
    Result Results_InnerForces_estaca = new Result();
    ResultKey InnerForces_estaca = new ResultKey
    {
      EntityType = eDsElementType.eDsElementType_Beam,
      EntityName = x.Name,
      CaseType = eDsElementType.eDsElementType_Combination,
      Caseld = y.Id,
      Dimension = eDimension.eDim_1D,
      ResultType = eResultType.eFemBeamInnerForces,
      CoordSystem = eCoordSystem.eCoordSys_Local
    };
    Results_InnerForces_estaca =
    resultsApi.LoadResult(InnerForces_estaca);

    linha = Results_InnerForces_estaca.GetTextOutput();
    [...]
    Result Results_Deformations_estaca = new Result();
    ResultKey Deformations_estaca = new ResultKey
    {
      EntityType = eDsElementType.eDsElementType_Beam,
      EntityName = x.Name,
      CaseType = eDsElementType.eDsElementType_Combination,
      Caseld = y.Id,
      Dimension = eDimension.eDim_1D,
      ResultType = eResultType.eFemBeamDeformation,
      CoordSystem = eCoordSystem.eCoordSys_Local
    };
    Results_Deformations_estaca =
    resultsApi.LoadResult(Deformations_estaca);

    linha = Results_Deformations_estaca.GetTextOutput();
  }
}

foreach (Combination y in Envoltoria)
{
  Result Results_InnerForces_slab = new Result();
  ResultKey InnerForces_slab = new ResultKey
  {
    EntityType = eDsElementType.eDsElementType_Slab,
    EntityName = slab.Name,
    CaseType = eDsElementType.eDsElementType_Combination,
    Caseld = y.Id,
    Dimension = eDimension.eDim_2D,
    ResultType = eResultType.eFemInnerForces,
    CoordSystem = eCoordSystem.eCoordSys_Local
  };
  Results_InnerForces_slab =
  resultsApi.LoadResult(InnerForces_slab);

  linha = Results_InnerForces_slab.GetTextOutput();
  [...]
  Result Results_Deformations_slab = new Result();
  ResultKey Deformations_slab = new ResultKey
  {
    EntityType = eDsElementType.eDsElementType_Slab,

```

```

        EntityName = slab.Name,
        CaseType = eDsElementType.eDsElementType_Combination,
        Caseld = y.Id,
        Dimension = eDimension.eDim_2D,
        ResultType = eResultType.eFemDeformations,
        CoordSystem = eCoordSystem.eCoordSys_Local
    };
    Results_Deformations_slab =
    resultsApi.LoadResult(Deformations_slab);

    linha = Results_Deformations_slab.GetTextOutput();
}
}
}
}
}

```

Dentro desses laços também são realizadas consolidações das informações, que são gravadas em dois outros relatórios. O relatório resumido contém cada um dos elementos (8 barras e a laje); todas as combinações, por exemplo as da ELU (ELU_1, ELU_2, ..., ELU_10); os tipos (força interna e deformação); as magnitudes; e os valores mínimos, máximos e máximos absolutos de cada magnitude. As magnitudes para as forças internas são: N, Vy, Vz, Mx, My e Mz para as estacas e nx, ny, vx, vy, mx, my e mxy para a laje. E as magnitudes para as deformações são: Ux, Uy, Uz, Fix, Fiy Fiz e Uglob.

O relatório consolidado contém as envoltórias (ELU, ELSF, ELSQ, ELSR); o elemento (estaca e laje); o tipo (força interna e deformação); as magnitudes; e os valores mínimos e máximos de cada magnitude. Observe que esse relatório apresenta apenas o caso mais crítico de todas as estacas.

O relatório consolidado, é composto de 118 (cento e dezoito) linhas. O relatório resumido, que relaciona a laje, todas as oito barras e todas as combinações, é composto de 5211 (cinco mil duzentos e onze) linhas. Enquanto que o relatório produzido pelo *OpenAPI*, que elenca, além dessas informações, todas as layers, é composto de incríveis 426.951 (quatrocentos e vinte e seis mil novecentos e cinquenta e uma) linhas.

A seguir, apresenta-se um exemplo do relatório consolidado, apenas para a envoltória ELU.

Figura 28 – Exemplo de envoltória ELU no relatório consolidado

ELU	Estaca	Deformação	Ux	0	0,00379
ELU	Estaca	Deformação	Uy	-0,01117	0,05345
ELU	Estaca	Deformação	Uz	-0,04281	0,04281
ELU	Estaca	Deformação	Fix	-0,007	0,007
ELU	Estaca	Deformação	Fiy	-0,00246	0,00246
ELU	Estaca	Deformação	Fiz	-0,00304	0,00067
ELU	Estaca	Deformação	Uglob	0	0,06052
ELU	Estaca	Força Interna	N	-1043496,96962	18574,86735
ELU	Estaca	Força Interna	Mx	0	0
ELU	Estaca	Força Interna	My	-269125,55307	269125,55306
ELU	Estaca	Força Interna	Mz	-323727,63287	226450,27162
ELU	Estaca	Força Interna	Vy	-51283,01775	28527,47811
ELU	Estaca	Força Interna	Vz	-42280,55654	42280,55654
ELU	Laje	Deformação	Ux	-0,04882	0,04882
ELU	Laje	Deformação	Uy	-0,01117	0,05568
ELU	Laje	Deformação	Uz	-0,00436	0,00002
ELU	Laje	Deformação	Fix	-0,00056	0,00009
ELU	Laje	Deformação	Fiy	-0,00035	0,00035
ELU	Laje	Deformação	Fiz	-0,007	0,007
ELU	Laje	Deformação	Uglob	0,00167	0,06309
ELU	Laje	Força Interna	mx	-160056,31543	85311,34624
ELU	Laje	Força Interna	my	-168098,26225	139735,72436
ELU	Laje	Força Interna	nx	-25100,27112	37637,14811
ELU	Laje	Força Interna	ny	-25296,42636	24574,05774
ELU	Laje	Força Interna	vx	-224384,79989	224384,79989
ELU	Laje	Força Interna	vy	-324614,95492	184149,45563
ELU	Laje	Força Interna	mxy	-96949,13305	96949,13554

Fonte: Elaborado pelo autor

Apesar dessa grande quantidade de dados, a navegação entre esses três relatórios demonstra-se, na prática, um processo fácil e ágil. Por exemplo, ao se desejar mais informações a respeito da seguinte linha, obtida da figura 28, basta fazer uma pesquisa no relatório resumido.

ELU Estaca Força Interna Mz -323727,63287 226450,27162

Ao realizar a pesquisa “localizar” no bloco de notas pelo valor -323727,63287, encontra-se imediatamente a seguinte informação.

Figura 29 – Estaca3C, ELU_4 do relatório resumido

```
#####
#Estaca: Estaca3C
#Combinação: ELU_4
#Forças Internas
#####
          Maximum inner force      Minimum inner force      Maximum absolute inner force
N          -317818,402335566        -984253,433128806        984253,433128806
Vy         24629,878287829          -2799,95809252074        24629,878287829
Vz         11461,6329296884         -9836,4668870699        11461,6329296884
Mx          0                        0                        0
My         126830,365151881         -146881,25757403        146881,25757403
Mz         32937,6579161409         -323727,632874141        323727,632874141
#####
#Deformações
#####
          Maximum absolute deformation
Ux          0,00378603967247023
Uy          0,0534508987277442
Uz          0,0266355608038117
Fix         0,00699830095571929
Fiy         0,00132408953481544
Fiz         0,0070160872300544
#####
```

Localizar

Localizar:

Diferenciar maiúsculas de minúsculas Acima Abaixo

Ao redor

Fonte: Elaborado pelo autor

Figura 30 – Estaca4C, ELU_3 do relatório resumido

```
#####
#Estaca: Estaca4C
#Combinação: ELU_3
#Forças Internas
#####
```

	Maximum inner force	Minimum inner force	Maximum absolute inner force
N	-317818,402335566	-984253,433128806	984253,433128806
Vy	24629,878287829	-2799,95809252074	24629,878287829
Vz	9836,4668870699	-11461,6329296884	11461,6329296884
Mx	0	0	0
My	146881,257584544	-126830,365162395	146881,257584544
Mz	32937,6579161409	-323727,632874141	323727,632874141

```
#####
#Deformações
#####
```

	Maximum absolute deformation
Ux	0,0037803967247023
Uy	0,0534508987277442
Uz	0,026635608038117
Fix	0,00699830095571929
Fiy	0,00132408953481544
Fiz	0,00270160823390544

Localizar

Localizar: Localizar Próxima

Diferenciar maiúsculas de minúsculas

Acima Abaixo

Ao redor

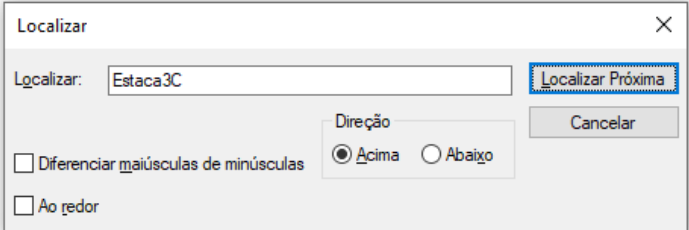
Cancelar

Fonte: Elaborado pelo autor

Assim verifica-se que tanto o caso ELU_4 - Estaca3C quanto o caso ELU_3 - Estaca4C possuem a situação crítica de menor momento fletor Mz dentro da envoltória de ELU, com o valor igual a -323,728 kNm. Além disso, pela figura 31, observa-se que a mesma pesquisa pode ser feita no relatório completo, permitindo obter mais detalhes sobre o comportamento da estrutura, para esse exemplo, no caso de momento fletor Mz mínimo.

Figura 31 – Estaca3C, ELU_4 do relatório completo

```
#Estaca: Estaca3C
#Combinação: ELU_4
#Forças Internas
#####
Member.Type = Beam
Member.ID = 5
Result type: FEM Beam Inner Forces
Case.Type = Combi
Case.ID = 7
Coord. system = LCS
Location of values = Centres
Magnitudes = 7
MeshElements = 18
```



elem	N	Vy	Vz	Mx	My	Mz
Layer 10: Magnitude = Mz, envelope = (Min):- - - - -						
0	-755787.25	21807.48	2964.62	0.00	-71230.07	-323727.63
1	-769538.22	21807.48	2964.62	0.00	-69312.94	-309625.46
2	-783289.28	21807.48	2964.62	0.00	-67395.86	-295523.30
3	-797040.22	22022.52	3179.66	0.00	-65409.90	-281352.30
4	-810791.15	22239.80	3396.94	0.00	-63283.46	-267040.84
5	-824542.12	22457.08	3614.22	0.00	-61016.52	-252588.84
6	-838293.05	22674.36	3831.50	0.00	-58609.08	-237996.33
7	-852043.98	22891.64	4048.78	0.00	-56061.12	-223263.33
8	-858919.38	23000.28	4157.42	0.00	-54734.49	-215844.38
9	-858919.82	23000.28	4157.42	0.00	-54734.40	-215843.92
10	-865795.05	23108.92	4266.06	0.00	-53372.65	-208389.82
11	-879546.02	23326.20	4483.34	0.00	-50543.68	-193375.79
12	-893296.95	23543.48	4700.62	0.00	-47574.20	-178221.28
13	-907047.88	23760.76	4917.90	0.00	-44464.21	-162926.22
14	-920798.81	23978.04	5135.18	0.00	-41213.71	-147490.69
15	-934549.92	24195.32	5352.46	0.00	-37822.71	-131914.63
16	-948300.85	24412.60	5569.74	0.00	-34291.20	-116198.07
17	-962051.78	24629.88	5787.02	0.00	-30619.18	-100341.00

Fonte: Elaborado pelo autor

A partir dessa busca, foi possível verificar os valores das demais magnitudes para o caso de Mz mínimo, no elemento de malha zero da *layer* dez, conforme destacado na figura 31. Além disso, o relatório completo apresenta a combinação de ações utilizada no cálculo dessa *layer*, que está de acordo com a ELU_4 da tabela 1 e com os casos de ações (*Load Cases* – LCs) definidas no programa.

$$1.050*LC2 + 1.050*LC6 + 1.050*LC8 + 1.500*LC20 + 1.050*LC11 + 1.500*LC21 + 1.050*LC12 + 1.500*LC22 + 1.050*LC13 + 1.500*LC23 + 1.050*LC14 + 1.500*LC24 + 1.050*LC15 + 1.500*LC25 + 1.050*LC16 + 1.500*LC26 + 1.050*LC17 + 1.500*LC27 + 1.050*LC18 + 1.500*LC28 + 1.050*LC19 + 1.500*LC29 + 1.400*LC1 + 1.400*LC3$$

7 AUTOMAÇÃO DO MODELO

O programa pode ser executado para produzir um resultado de um único modelo, informando os dezenove parâmetros mencionados no item 5. Isso produzirá

os três arquivos texto referentes aos relatórios completo, resumido e consolidado. Além disso, produzirá um arquivo do *SCIA Engineer* de extensão “.esad” com o modelo criado e calculado, o que permite estudos detalhados sobre o modelo gerado. Todos esses quatro arquivos terão, ao final do nome, o formato “_AAAAMMDD_HHmm”, onde AAAA é o ano com quatro dígitos, MM é o mês com dois dígitos, DD é o dia com dois dígitos, HH é a hora com dois dígitos e mm é o minuto com dois dígitos. Essa data e horário são obtidos no momento em que o programa em C# inicializa o *SCIA Engineer*, de forma que todos os arquivos terão o mesmo final.

Além de poder rodar um modelo por vez, é possível executar o programa gerando diversas variações do mesmo modelo. Para isso, é necessário definir valores para duas classes de variáveis, apresentadas na figura 32 e na figura 33.

Figura 32 – Classe “Variáveis”, contendo os dezenove parâmetros

```
Variaveis.Variaveis(double a, double b, double c, double estaca_alpha, double estaca_beta, double h_solo, double h_laje_solo, double h_agua, double h_conc,
double no_estaca_x, double no_estaca_y, double rigidez_i, double rigidez_f, double no_cabeco_x, double no_cabeco_y, double sobrecarga, double corrente_long,
double corrente_transv, double forca_cab)
```

Fonte: Visual Studio 2022

Figura 33 – Classe “Variáveis_iter”, contendo os parâmetros de iteração

```
Variaveis_iter.Variaveis_iter(double estaca_alpha_passo, double estaca_beta_passo, double no_estaca_x_passo, double no_estaca_y_passo, double estaca_alpha_iter,
double estaca_beta_iter, double no_estaca_x_iter, double no_estaca_y_iter)
```

Fonte: Visual Studio 2022

O *Visual Studio* facilita a apresentação dos parâmetros do procedimento. As variáveis *estaca_alpha*, *estaca_beta*, *no_estaca_x* e *no_estaca_y* podem ser iteradas. Os valores iniciais dessas variáveis constarão no procedimento *Variáveis*. Esses valores serão incrementados de um valor fixo definido respectivamente em *estaca_alpha_passo*, *estaca_beta_passo*, *no_estaca_x_passo* e *no_estaca_y_passo*. Além disso, serão realizados o número fixo de iterações definidos em *estaca_alpha_iter*, *estaca_beta_iter*, *no_estaca_x_iter* e *no_estaca_y_iter*.

Para demonstrar a funcionalidade do programa desenvolvido, efetuou-se a automação de um caso, alterando os ângulos alpha e beta. O ângulo alpha variou de 60° até 90° com um passo de 5° e o ângulo beta variou de 30° até 60° com um passo de 5°, gerando os valores de variáveis de iteração apresentados abaixo.

Tabela 8 – Valores de variáveis de iteração utilizados no exemplo

Parâmetro	Valor
estaca_alpha_passo	5°
estaca_beta_passo	5°
no_estaca_x_passo	0°
no_estaca_y_passo	0°
estaca_alpha_iter	7 iterações
estaca_beta_iter	7 iterações
no_estaca_x_iter	1 iteração
no_estaca_y_iter	1 iteração

Fonte: Elaborado pelo autor

Os valores das demais variáveis estão definidos na Tabela 9.

Tabela 9 – Valores das demais variáveis utilizadas no exemplo

Parâmetro	Valor
a	6 m
b	5 m
c	1,5 m
α	60° - 90°
β	30° - 60°
h _{solo}	7,7 m
h _{laje_solo}	21,3 m
h _{nível_água}	20 m
h _{concretada}	9,7 m
Est_x	1,5 m
Est_y	1 m
Rigidez _{inicial}	4,57 MN/m
Rigidez _{final}	75,405 MN/m
Cab_x	1,5 m
Cab_y	2,5 m
Sobrecarga	10 kN
Força _{Corrente_long}	320 N
Força _{Corrente_transV}	320 N
Força _{Cabeço}	5 kN

Fonte: Elaborado pelo autor

Dessa forma, mostram-se os comandos utilizados no programa para a definição das variáveis e das variáveis de iteração, respectivamente, conforme os valores apresentados na tabela 8 e na tabela 9.

```
Variaveis variaveis = new Variaveis(6, 5, 1.5, 60, 30, 7.7, 21.3, 20, 9.7, 1.5,
1, 4.57e6, 75.405e6, 1.5, 2.5, 10e3, 320, 320, 5e3);
```

```
Variaveis_iter var_iter = new Variaveis_iter(5, 5, 0, 0, 7, 7, 1, 1);  
CreateCustomModel(variaveis, var_iter);
```

Assim, o modelo terá sete variações do ângulo alpha e sete variações do ângulo beta, totalizando 49 (quarenta e nove) modelos distintos. Essas informações produziram 49 (quarenta e nove) relatórios consolidados, que foram incluídos em planilhas do Excel.

8 CÁLCULOS E RESULTADOS

A partir dos resultados da automação de caso, mencionada no item 7, foram produzidos diversos gráficos para o Estado Limite Último (ELU). Com isso, procura-se comparar a influência da variação dos ângulos alpha e beta nos diversos esforços e deformações nas estacas e na laje do modelo.

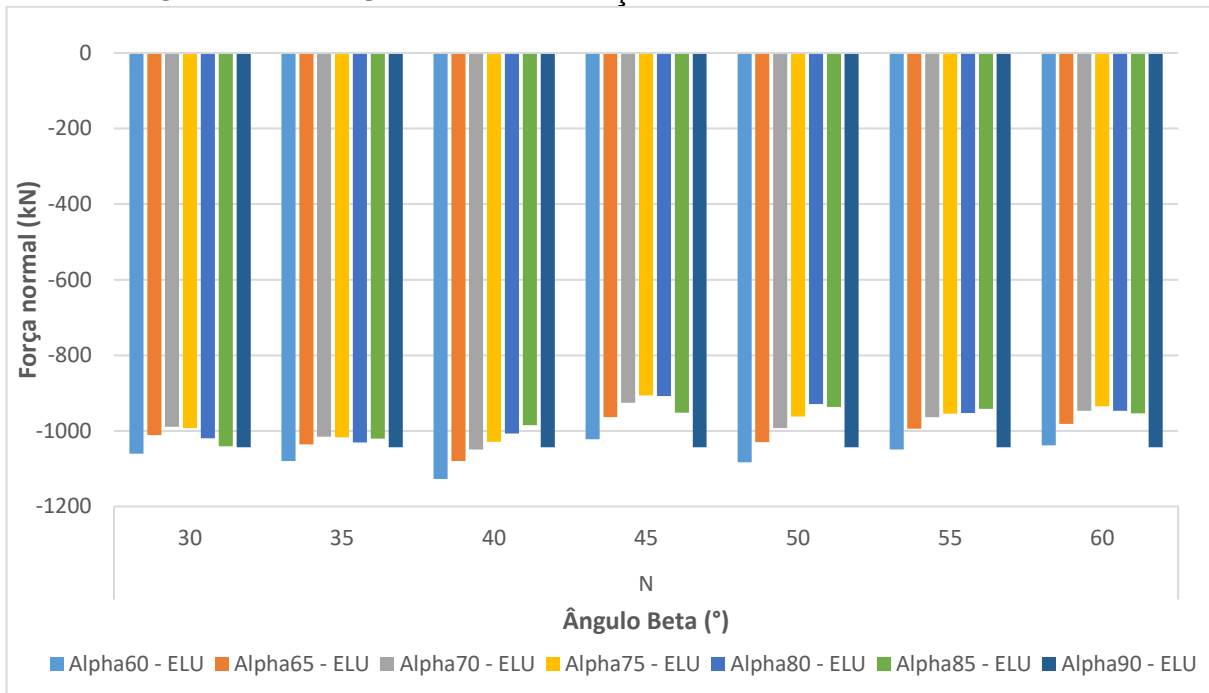
No caso das forças internas, são comparados os valores mínimos e máximos de cada magnitude para cada combinação de ângulos alpha e beta.

8.1 ESTACAS

8.1.1 Forças internas

Os gráfico 1 e o gráfico 2 apresentam os valores da menor e da maior força normal, respectivamente, aplicadas nas estacas. Pelo gráfico 1, verifica-se que o ângulo alpha de 75° e ângulo beta de 45° possuem o menor módulo mínimo, igual a 906 kN de compressão. Também, verifica-se que os ângulos alpha intermediários (70°, 75°, 80°) tendem a produzir menores esforços de compressão nas estacas.

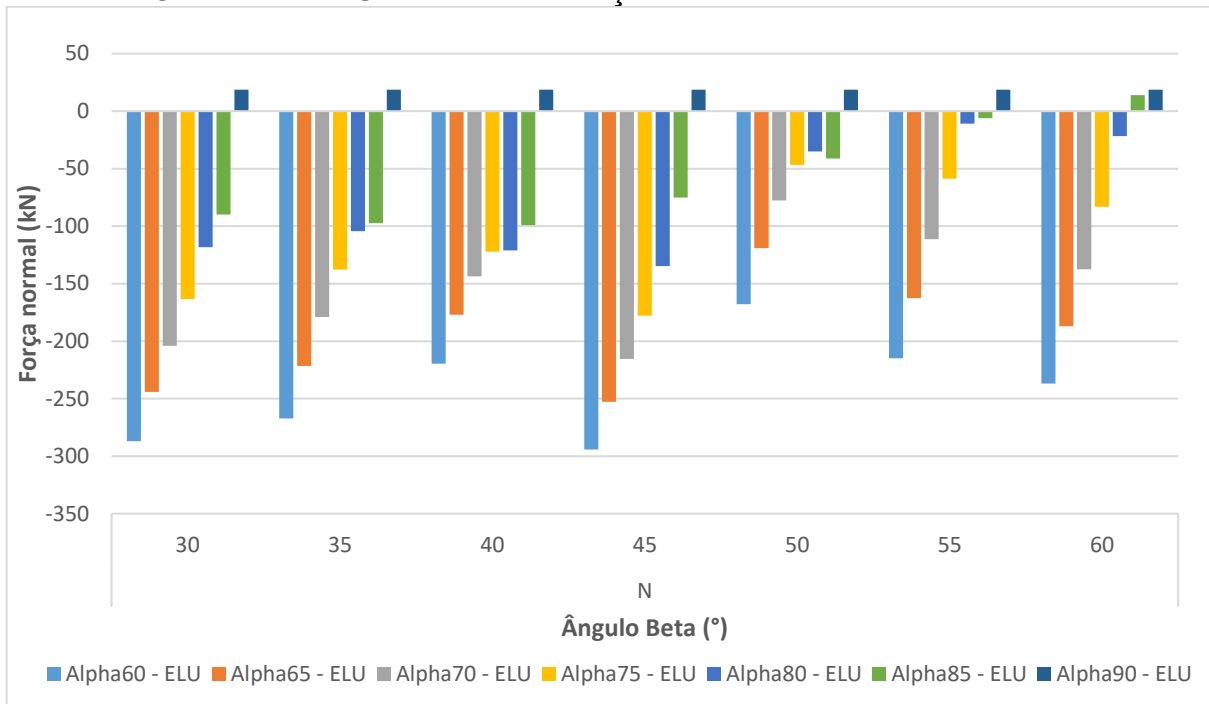
Gráfico 1 – ELU – Estacas – Forças internas – Normal – Mínimo



Fonte: Elaborado pelo autor.

Pelo gráfico 2, observa-se que os valores máximos de força normal também apresentam valores de compressão, com exceção do ângulo alpha de 90°. Nesse ângulo, há um pequeno montante de tração, de aproximadamente 18 kN. E também, o caso do ângulo alpha de 85° e ângulo beta de 60°, com 14 kN. Verifica-se que com o aumento do beta, os valores tendem a aumentar (diminuir em módulo). Como esses casos em módulo são inferiores aos do gráfico 1, a maior preocupação está na possibilidade de esforços de tração. Porém, como esses valores são baixos, os valores de normais significativos para a escolha do modelo seriam os do gráfico 1.

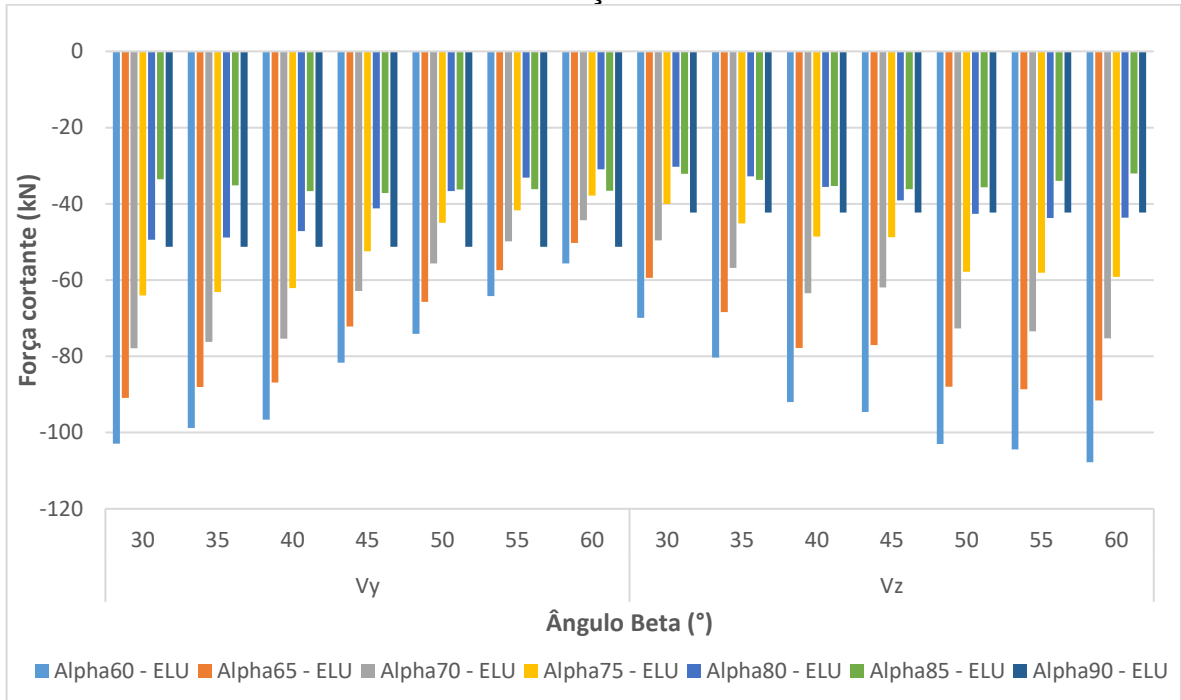
Gráfico 2 – ELU – Estacas – Forças internas – Normal – Máximo



Fonte: Elaborado pelo autor

Para as forças cortantes, os valores mínimos se encontram no gráfico 3 e os valores máximos se encontram no gráfico 4. Os valores mínimos são todos negativos e os valores máximos são todos positivos. Os menores módulos dos valores negativos se encontram no ângulo alpha 85° e o ângulo beta podendo ser 45°, 50° e 55°, apresentando V_y e V_z máximo em torno de 36~37 kN.

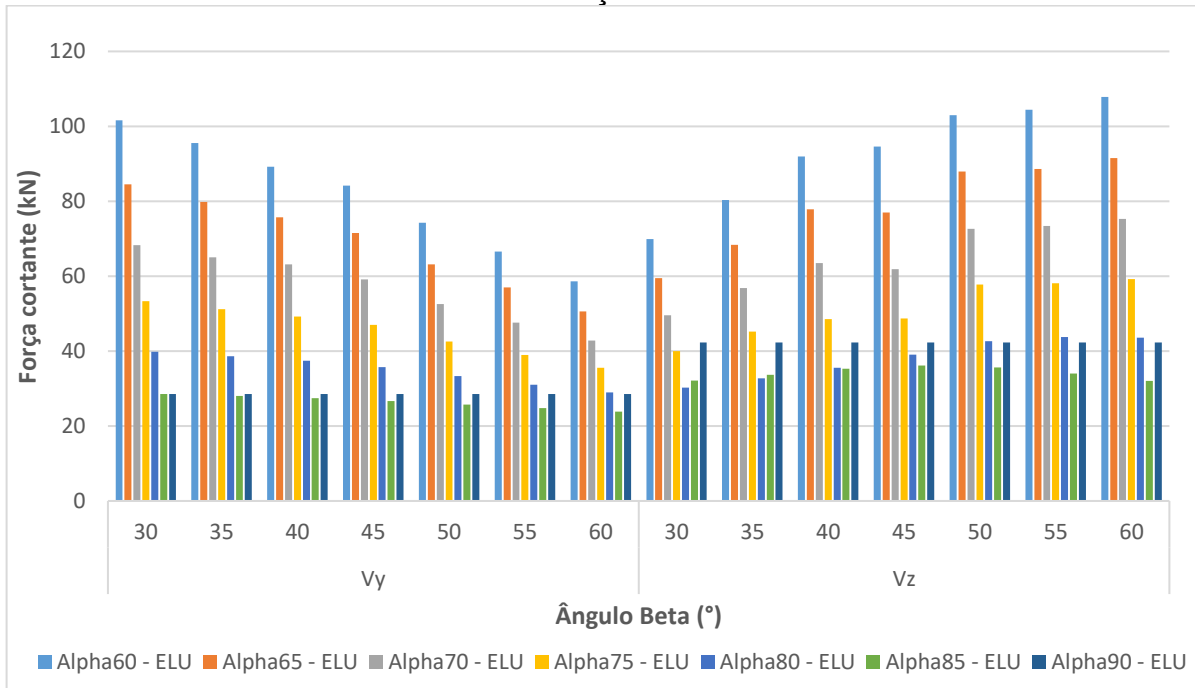
Gráfico 3 – ELU – Estacas – Forças internas – Cortante – Mínimo



Fonte: Elaborado pelo autor

Pelo gráfico 4 é possível observar que as forças cortantes máximas em módulo são inferiores aos módulos das mínimas, em especial entre os ângulos alpha 80° e 85°. Portanto, para a análise dos esforços cortantes, o gráfico 3 prevalece.

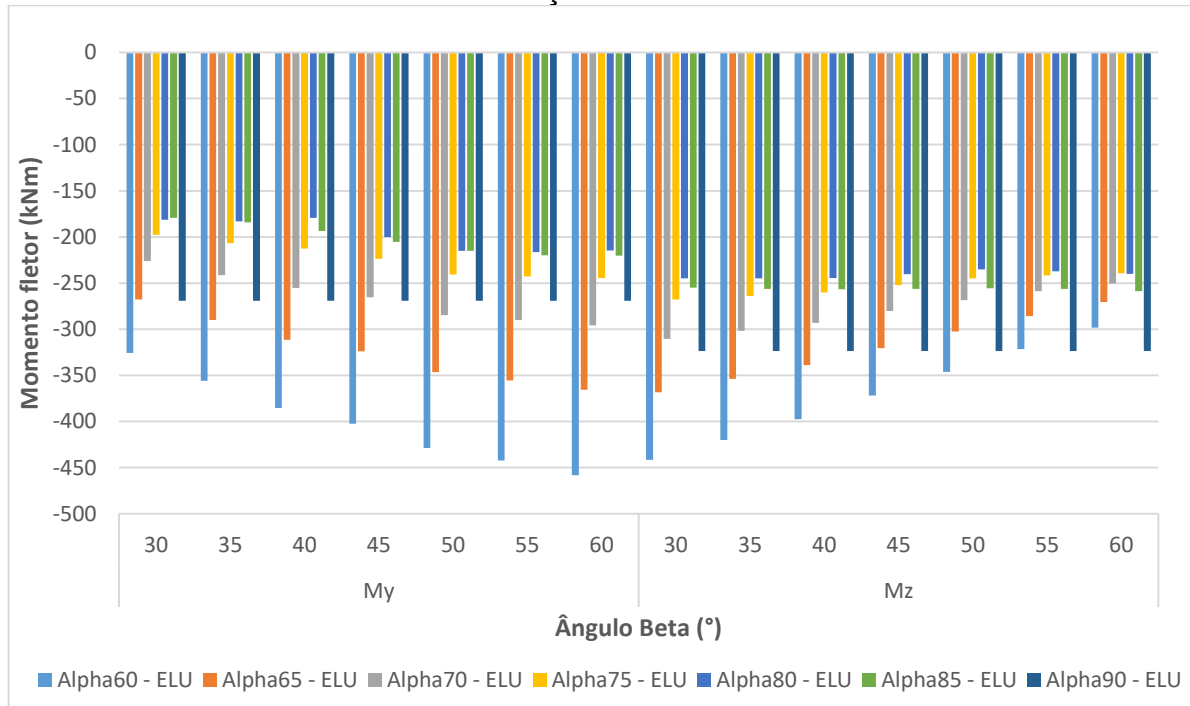
Gráfico 4 – ELU – Estacas – Forças internas – Cortante – Máximo



Fonte: Elaborado pelo autor

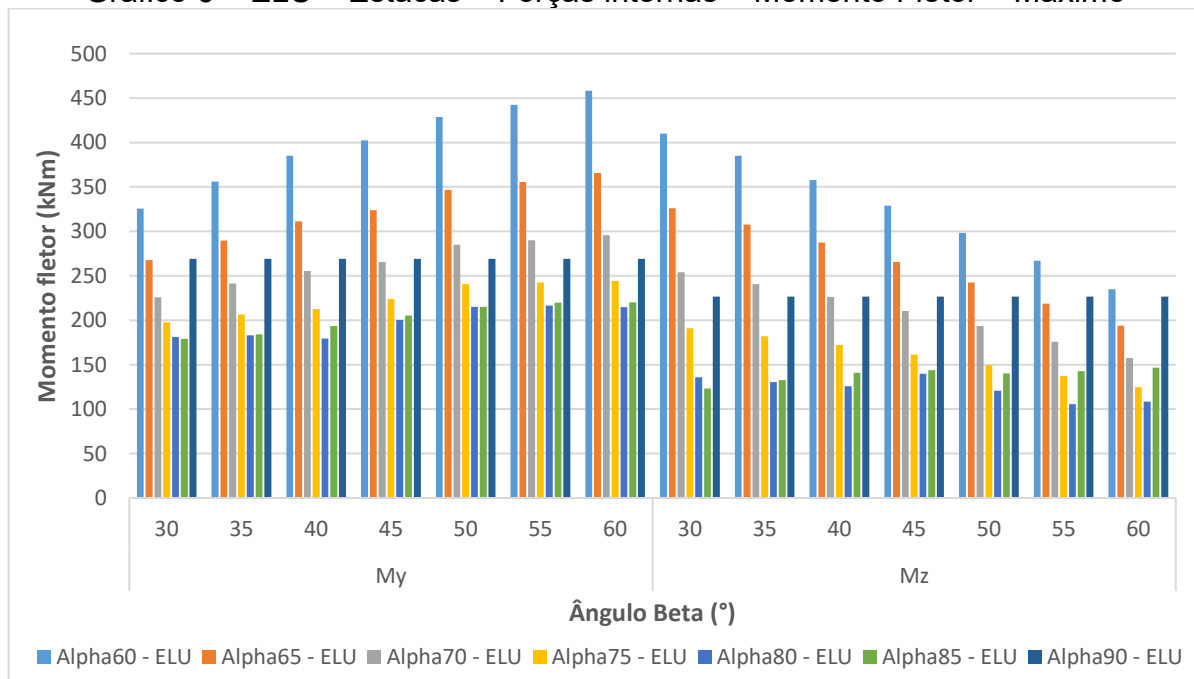
Pelo gráfico 5 e pelo gráfico 6, verifica-se que os ângulos alpha de 80° e 85° apresentam os melhores valores com o momento fletor entre 170 e 250 kNm.

Gráfico 5 – ELU – Estacas – Forças internas – Momento Fletor – Mínimo



Fonte: Elaborado pelo autor

Gráfico 6 – ELU – Estacas – Forças internas – Momento Fletor – Máximo



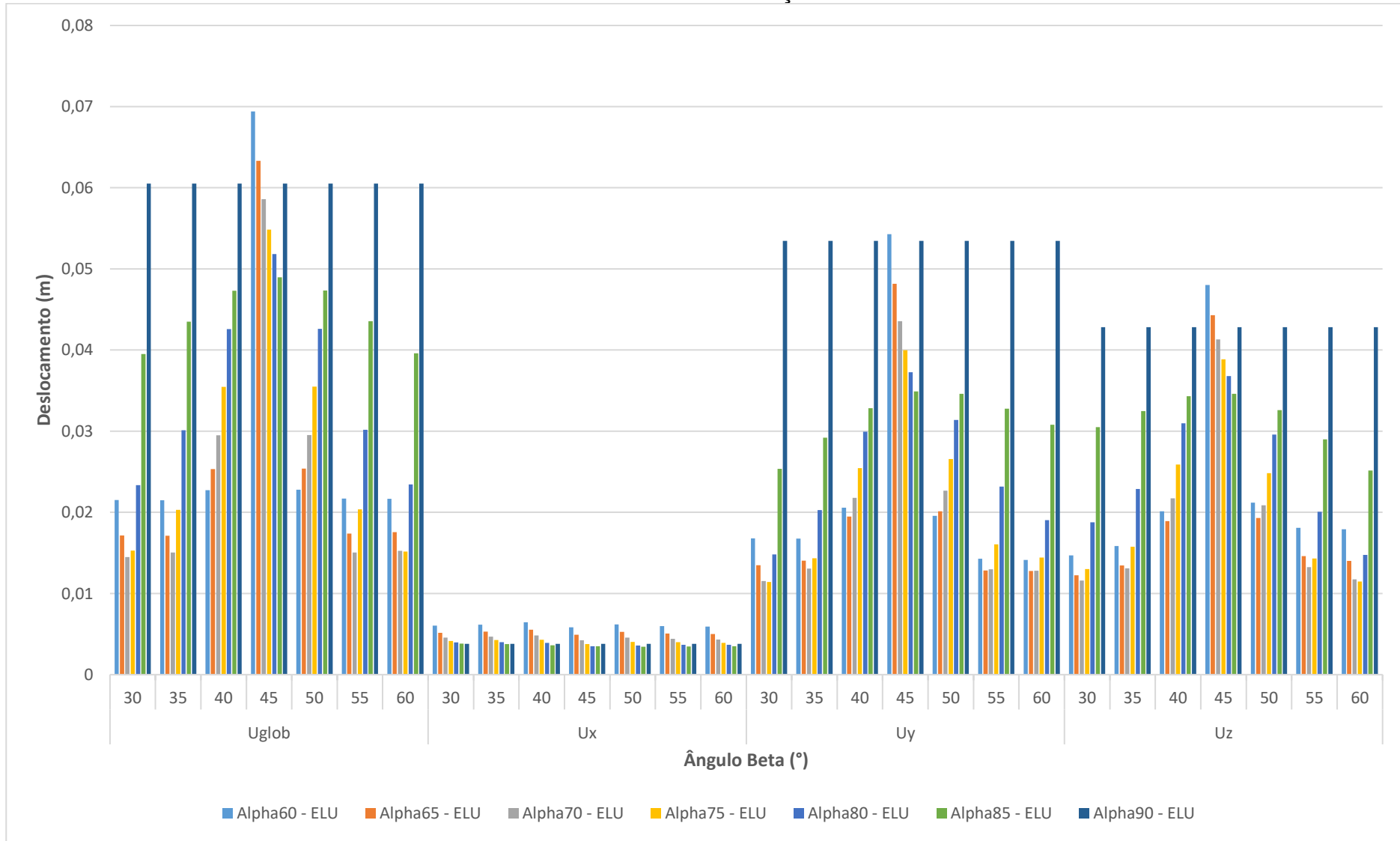
Fonte: Elaborado pelo autor

8.1.2 Deformações

Nas deformações verifica-se pelo gráfico 7 que o deslocamento de magnitude mais importante é o U_{glob} . O pior caso encontra-se com o ângulo beta de 45° , com o deslocamento diminuindo tanto em direção aos 30° quanto em direção aos 60° . Em relação ao ângulo alpha, é nítido que os melhores ângulos são o 70° e o 75° .

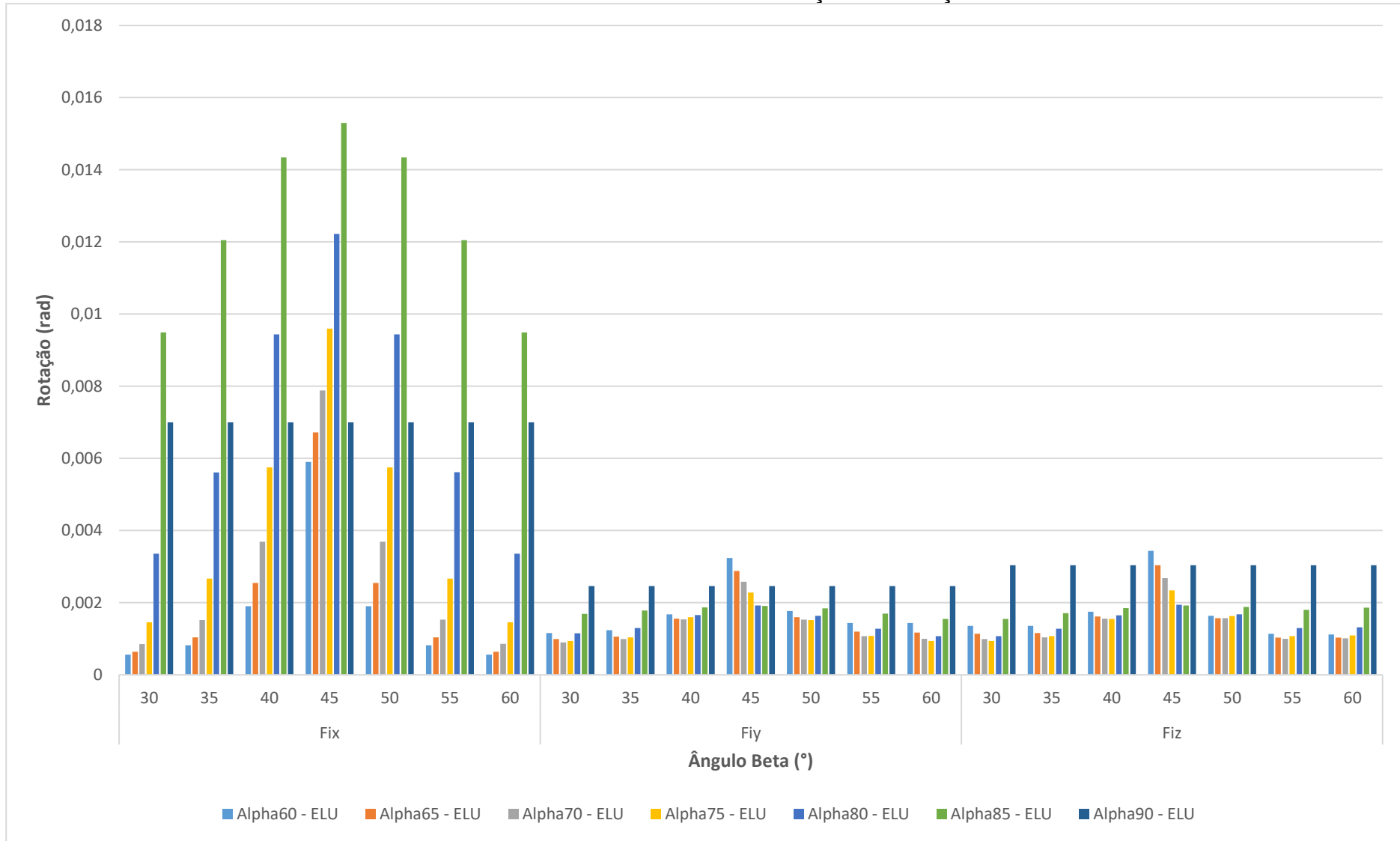
Pelo gráfico 8 os piores casos de rotação são dados quando o ângulo alpha é igual a 85° e para os ângulos de alpha inferiores a 80° , os valores de rotação são baixos e sem grandes variações.

Gráfico 7 – ELU – Estacas – Deformações – Deslocamentos



Fonte: Elaborado pelo autor

Gráfico 8 – ELU – Estacas – Deformações – Rotações



Fonte: Elaborado pelo autor

Analisando os casos das estacas, percebe-se que os valores significativos são em relação à força normal, em especial ao caso mínimo, como comentado na análise do gráfico 1 e do gráfico 2. Além disso, o valor das deformações é o outro fator preponderante. Ocorre que os valores normais não possuem grandes dispersões em relação aos ângulos. As deformações se comportam em um formato de pico no ângulo beta de 45° , diminuindo de uma forma constante a medida que se afasta desse pico. Por outro lado, a força normal tem valores menores nos ângulos beta maiores que 45° . Assim, em relação ao ângulo beta, tem-se melhores resultados nos ângulos 55° e 60° .

Pelo gráfico 1, a melhor escolha seria a de ângulo alpha igual a 75° e de ângulo beta 60° . Da mesma forma, essa mesma combinação de ângulos apresenta os menores deslocamentos, sendo a escolha para o caso das estacas.

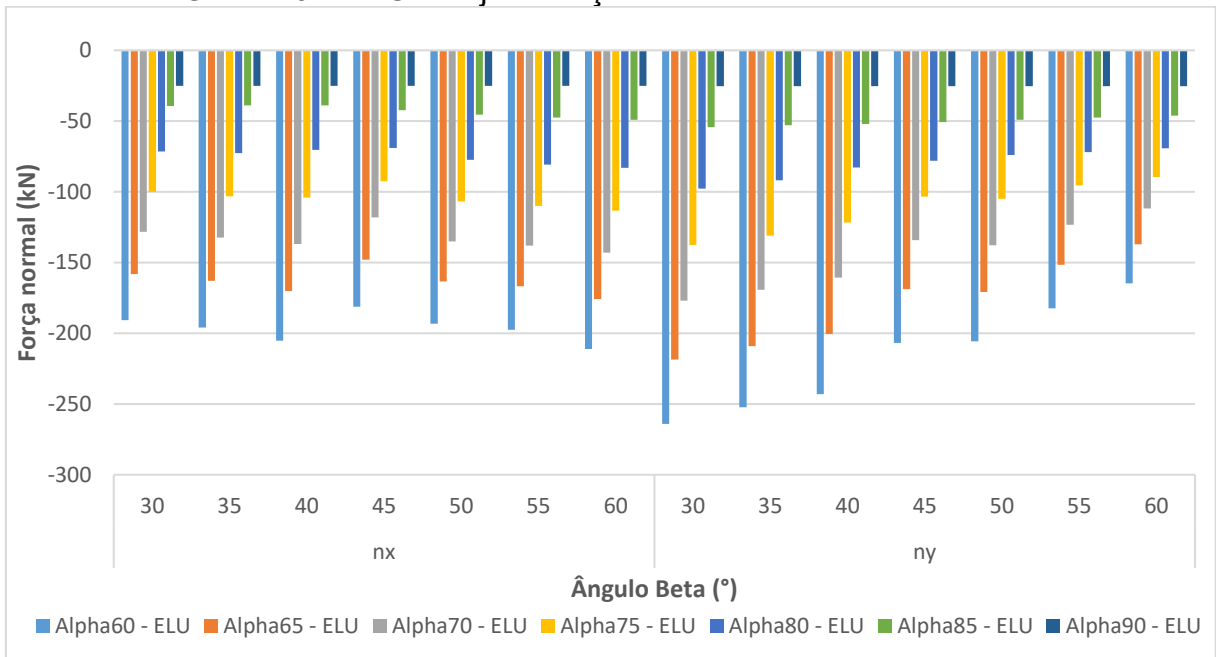
Considerando a robustez da laje, o fator preponderante na escolha do modelo será dado pelas condições das estacas, razão pela qual, dos modelos apresentados, o melhor seria o de alpha igual a 75° e beta igual a 60° .

8.2 LAJE

8.2.1 Forças internas

Pelo gráfico 9, observa-se que quanto maior o ângulo alpha, menores são as forças, sendo todas elas de compressão.

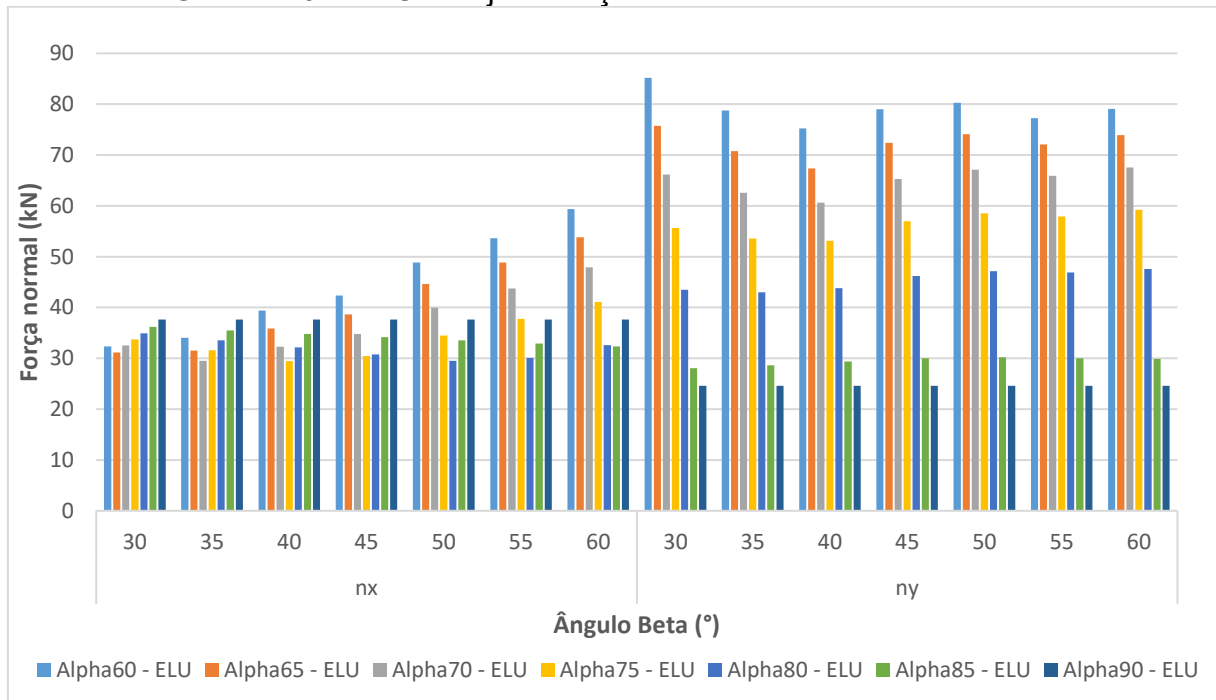
Gráfico 9 – ELU – Laje – Forças internas – Normal – Mínimo



Fonte: Elaborado pelo autor

Por outro lado, ao analisar o gráfico 10, todas as forças são de tração. Observa-se um pico do n_y para ângulos α menores. Mas mesmo assim, os valores de força normal são inferiores a 90 kN. Dado a escolha óbvia da análise de resultados das estacas, α igual a 75° e β igual a 60° , apresentam um resultado de força normal em torno de 60 kN.

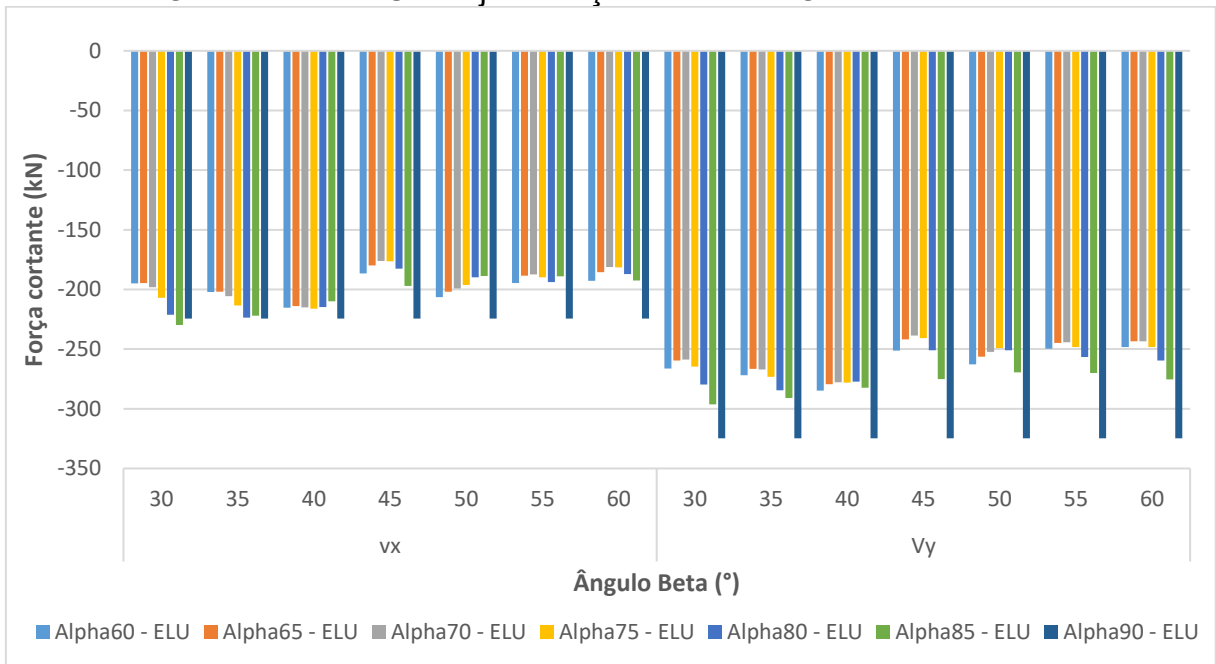
Gráfico 10 – ELU – Laje – Forças internas – Normal – Máximo



Fonte: Elaborado pelo autor

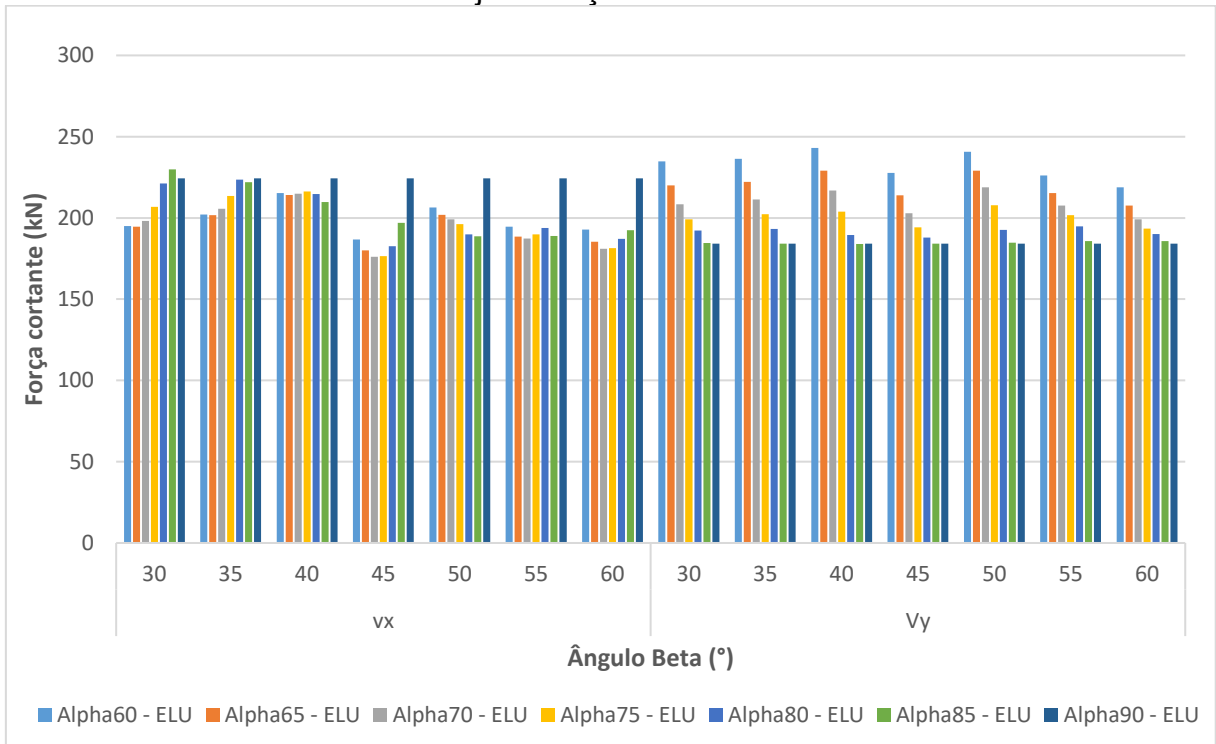
Segundo o gráfico 11, observa-se um pico no valor da força cortante v_y com α de 90° . Considerando a escolha de α igual a 75° e β igual a 60° , a força cortante máxima, em módulo, seria de 250 kN. Os esforços cortantes v_y do gráfico 11 são superiores, em módulo, ao v_x dessa figura e ao v_x e v_y do gráfico 12, sendo o valor crítico analisado para a escolha de α e β .

Gráfico 11 – ELU – Laje – Forças internas – Cortante – Mínimo



Fonte: Elaborado pelo autor

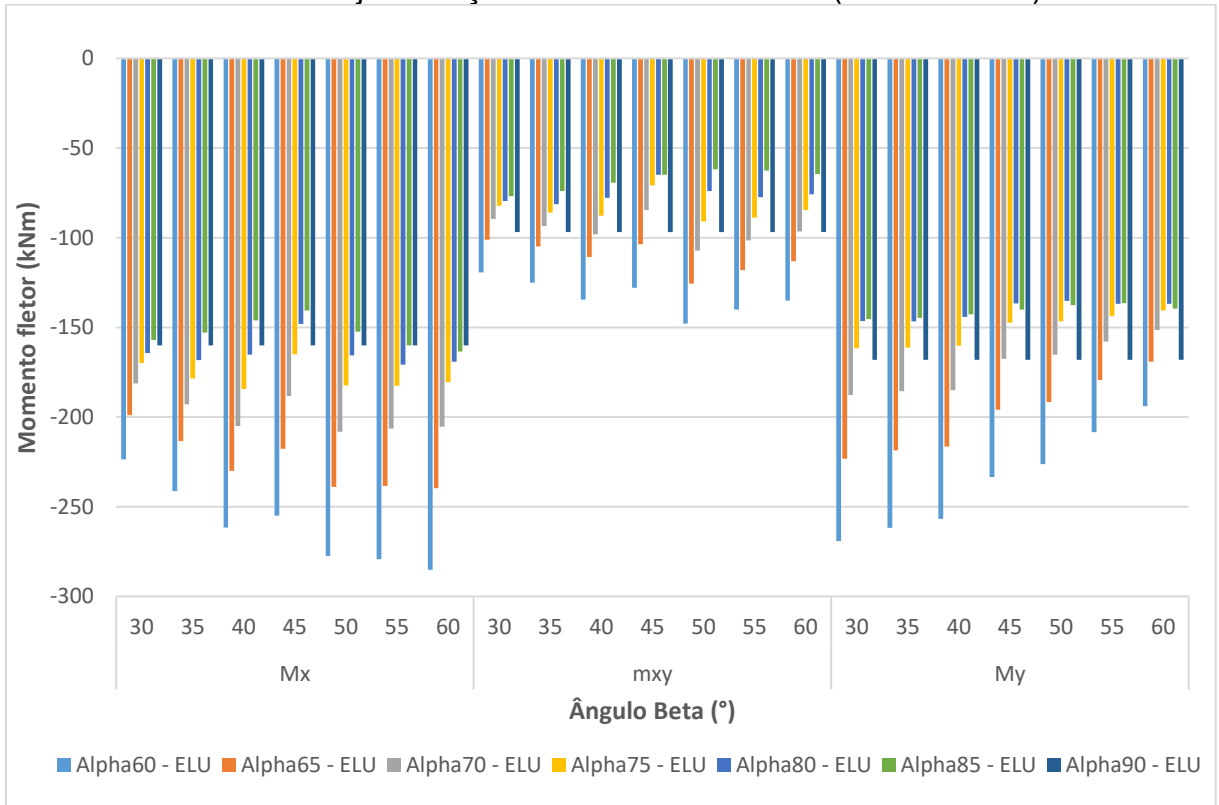
Gráfico 12 – ELU – Laje – Forças internas – Cortante – Máximo



Fonte: Elaborado pelo autor

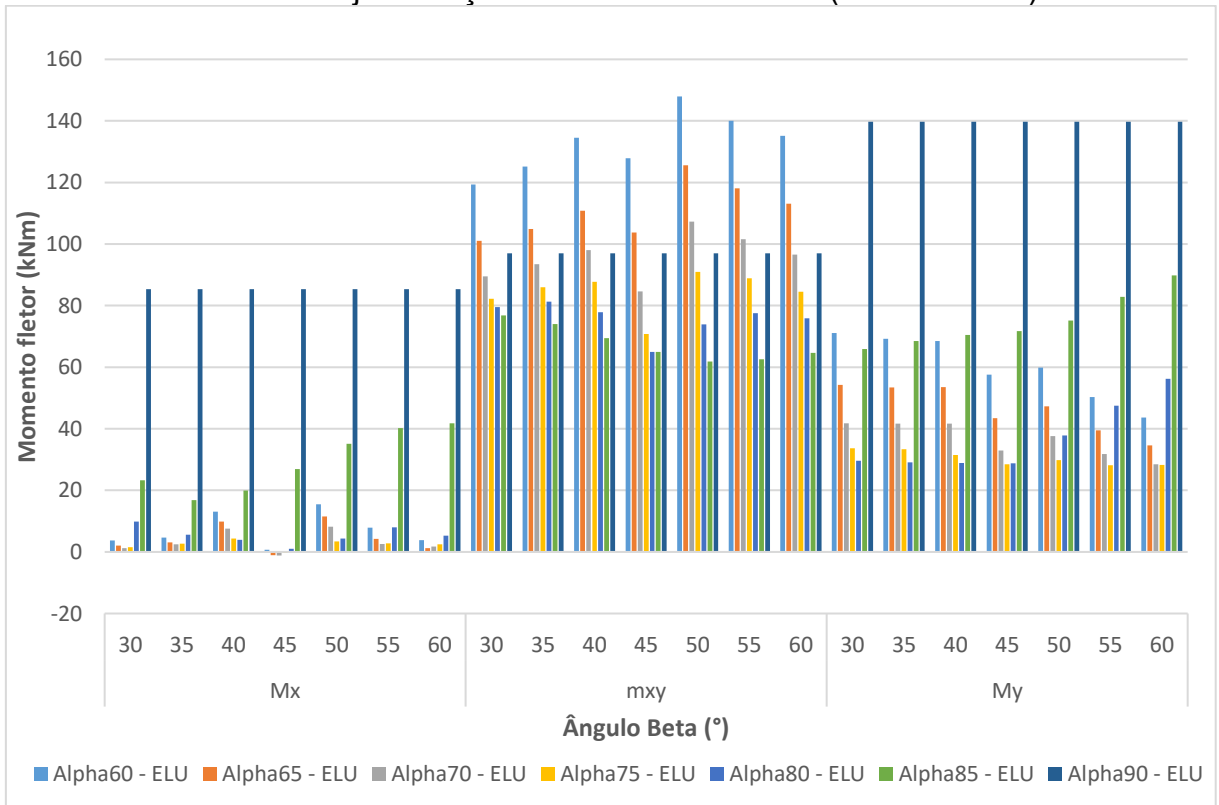
Pela análise do gráfico 13 e do gráfico 14, verifica-se que o momento fletor M_x mínimo prepondera sobre os demais, com quase todos os valores superiores a 150 kNm. Na escolha dos ângulos alpha igual a 75° e beta igual a 60° , o valor máximo, em módulo, de momento fletor é de 180 kNm.

Gráfico 13 – ELU – Laje – Forças internas – Momentos (fletor e torsor) – Mínimo



Fonte: Elaborado pelo autor

Gráfico 14 – ELU – Laje – Forças internas – Momentos (fletor e torsor) – Máximo



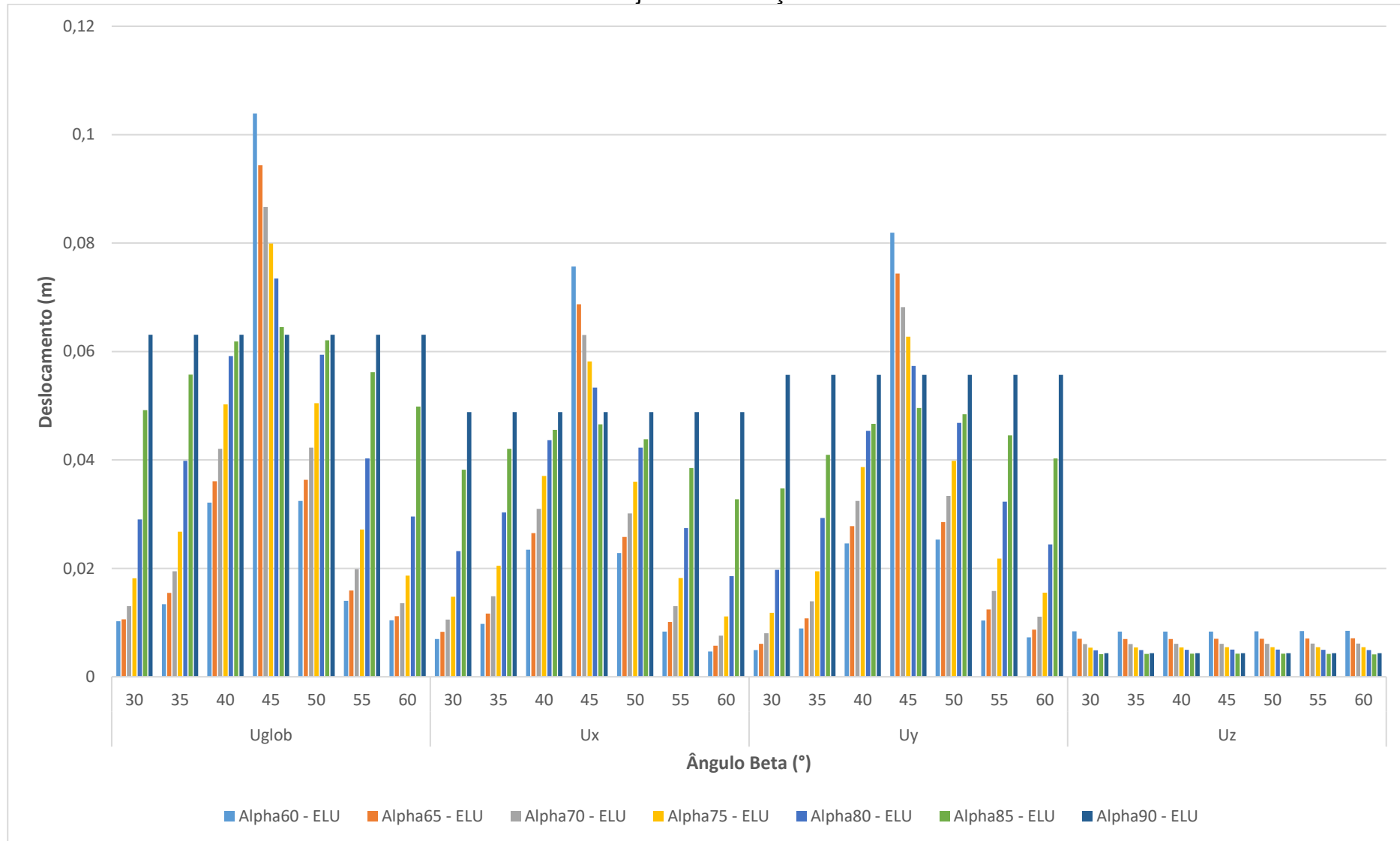
Fonte: Elaborado pelo autor

8.2.2 Deformações

Ao analisar o gráfico 15, observa-se que a magnitude U_{glob} é a preponderante. Assim como já analisado nas deformações das estacas, a laje também se comporta com o pior caso sendo o de beta igual a 45° , amenizando as deformações à medida que se afasta desse ângulo. Para o caso escolhido de alpha igual a 75° e beta igual a 60° , o valor máximo de deslocamento foi de 19 mm.

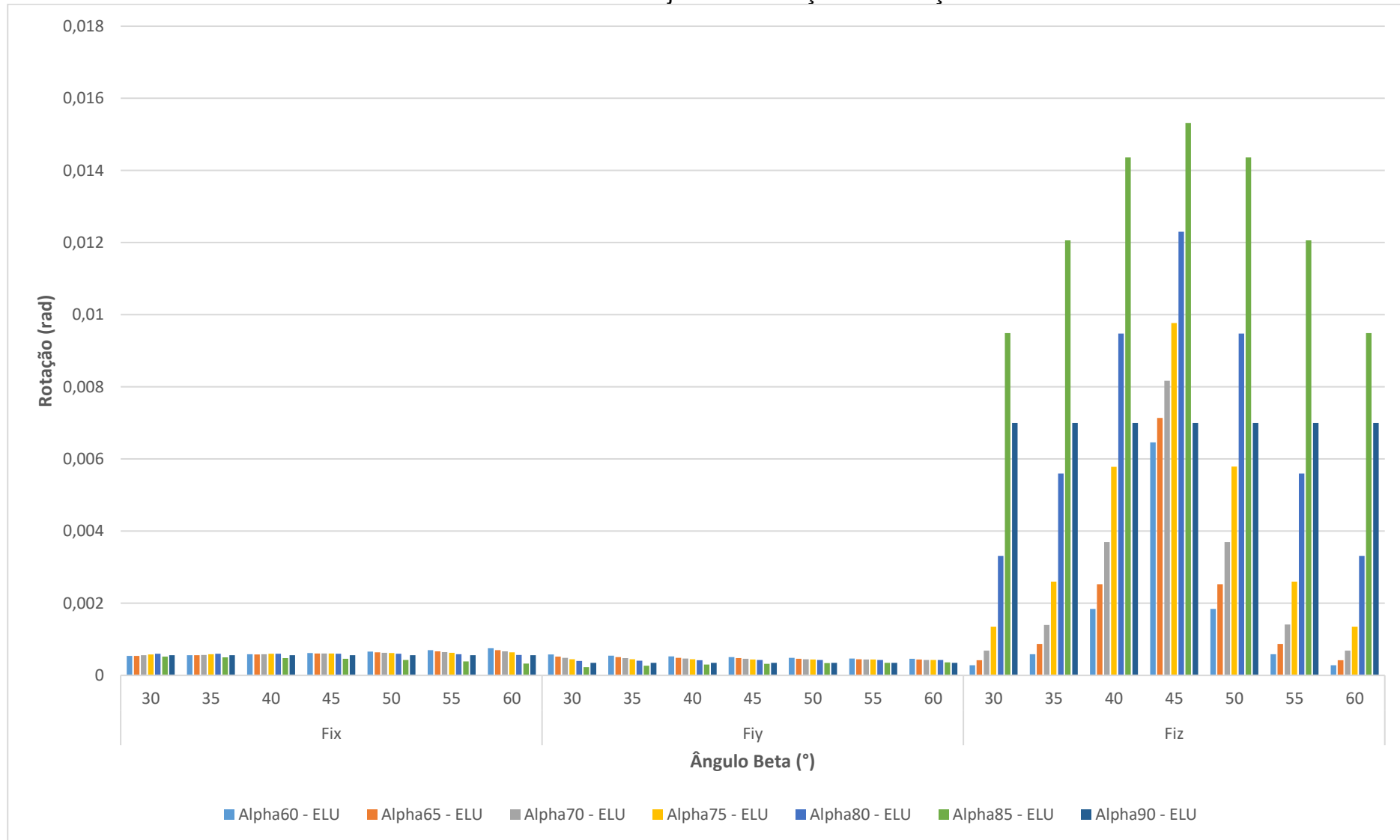
Pelo gráfico 16, os piores casos de rotação acontecem quando o ângulo alpha é igual a 85° diminuindo gradativamente quanto menor o ângulo alpha. Para a combinação escolhida de alpha igual a 75° e beta igual a 60° , obtém-se o valor máximo de rotação igual a 1,35 mrad.

Gráfico 15 – ELU – Laje – Deformações – Deslocamentos



Fonte: Elaborado pelo autor

Gráfico 16 – ELU – Laje – Deformações – Rotações



Fonte: Elaborado pelo autor

9 CONCLUSÃO

Este trabalho teve como principal objetivo testar a funcionalidade do *OpenAPI* para a automação do processo de projeto de estruturas estaqueadas de *dolphins* de amarração. Apesar da falta de algumas funções presentes na interface gráfica do software *SCIA Engineer*, como ações de temperatura ou conexão de elementos, o *OpenAPI* se mostrou capaz de automatizar o processo completo do projeto de um *dolphin* satisfatoriamente.

Além disso, também foi possível implementar a iteração dos parâmetros mais importantes na concepção dessas estruturas, com a automação apresentada no item 7 gerando, com apenas um clique, 49 (quarenta e nove) *dolphins* de amarração diferentes com seus respectivos resultados de análise estrutural. Com adaptações do programa gerado neste trabalho, engenheiros portuários experientes podem diminuir o esforço realizado na tarefa repetitiva de projetar *dolphins* por meio de tentativa e erro.

Concluindo, o *OpenAPI* do *SCIA Engineer* demonstra ser uma ferramenta poderosa na automação de projetos de *dolphins*. Acrescentando, a ferramenta avançada *Analysis Data Model* (ADM) do *SCIA Engineer* permite criar automações ainda melhores, com o ônus de ser mais complexa e trabalhosa na fase de desenvolvimento do programa.

Como recomendação do autor para quem utiliza o *SCIA Engineer*, a implementação de rotinas utilizando tanto o *OpenAPI* quanto o ADM pode aumentar a eficiência na concepção de projetos estruturais. Além disso, abre-se a possibilidade da otimização estrutural, que pode gerar economia de material e, conseqüentemente, economia financeira na fase posterior de execução das estruturas otimizadas.

Como sugestão para trabalhos futuros, recomenda-se:

- a) Expandir a análise para englobar *dolphins* de atracação;
- b) Incluir o número de estacas do *dolphin* como um dos parâmetros iterativos, gerando configurações de estaqueamento mais compatíveis com a realidade;
- c) Implementar algoritmos metaheurísticos de otimização estrutural, visando minimizar o custo global da construção do *dolphin* de amarração;

- d) Adaptar o programa desenvolvido para a ferramenta avançada de interação com o *SCIA Engineer*, o *Analysis Data Model (ADM)*, incluindo as ações de temperatura³ e as conexões de membros.

³ O programa desenvolvido já inclui os casos e as combinações de ações prontos para receber as ações de temperatura.

REFERÊNCIAS

ALONSO, Urbano R. **Dimensionamento de fundações profundas**. 1. ed. São Paulo: Blucher, 1989.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 6118**: projeto de estruturas de concreto. Rio de Janeiro: ABNT, 2023.

_____. **ABNT NBR 8800**: projeto de estruturas de aço e de estruturas mistas de aço e concreto de edifícios. Rio de Janeiro: ABNT, 2008.

_____. **ABNT NBR 9782**: ações em estruturas portuárias, marítimas ou fluviais: procedimento. Rio de Janeiro: ABNT, 1987. (NORMA CANCELADA EM 04/05/2015).

BRITISH STANDARD. **BS 6349-1**: maritime structures: part 1: code of practice for general criteria. London, 2000.

_____. **BS 6349-2**: maritime structure: part 2: design of quay walls, jetties and dolphins. London, 1988.

GAYTHWAITE, John W. **Design of marine facilities for the berthing, mooring, and repair of vessels**. 2. ed. Reston: American Society of Civil Engineers, 2004.

_____. **Design of marine facilities**: engineering for port and harbor structures. 3. ed. Reston: American Society of Civil Engineers, 2016.

GERWICK JUNIOR, Ben C. **Construction of marine and offshore structures**. 3. ed. Boca Raton: CRC Press, 2007.

HENRIQUE, Thiago. **Modelagem estrutural de dolphins de atracação de navios através da análise modal operacional**. Orientador: Professor Doutor. Walnório Graça Ferreira. 2019. 74 f. Dissertação (Mestrado em Engenharia Civil) – Universidade Federal do Espírito Santo, Centro Tecnológico, Vitória, 2019. Disponível em: <https://repositorio.ufes.br/handle/10/11170>. Acesso em: 08 out. 2023.

SCIA ENGINEER. **Basic Concept Training**: SCIA Enginner 21. Disponível em: <https://downloads.scia.net/support/sciaengineer/manuals/MAN%20%5bEng%5d%20SCIA21.1%20-%20Fundamentals%20Training.pdf>. Acesso em: 20 de out. 2023.

SHARP, John. **Microsoft visual C#**: step by step. 10. ed. [S.I.]: Pearson, 2022.

SOUSA, Augusto Teixeira de. **Estudo do comportamento de esforços em estacas de estruturas portuárias sob variações de altura de ondas e maré**. Orientadora: Professora Doutora Helena Paula Nierwinski. 2021. Trabalho de Conclusão de Curso (Graduação em Engenharia Civil de Infraestrutura) – Universidade Federal de Santa

Catarina, Joinville, 2021. Disponível em:
<https://repositorio.ufsc.br/handle/123456789/228622>. Acesso em: 12 out. 2023.

SPHAIR, Bruno Eduardo. **Análise de esforços em estruturas portuárias**. Orientador: Professor Doutor Rafael Holdorf Lopez. 2019. Trabalho de Conclusão de Curso (Graduação de Engenharia Civil) – Universidade Federal de Santa Catarina, Florianópolis, 2019. Disponível em:
<https://repositorio.ufsc.br/handle/123456789/202856>. Acesso em: 10 out. 2023.

THORESEN, Carl A. **Port designer's handbook**. 3. ed. London: ICE Publishing, 2014.

TIEGS, Luís Henrique. **Dimensionamento de ocupação e identificação de possíveis alocações para contêineres vazios no pátio público do porto de Itajaí**. Orientador: Professor Doutor Amir Mattar Valente. 2019. Trabalho de Conclusão de Curso (Graduação de Engenharia Civil) – Universidade Federal de Santa Catarina, Florianópolis, 2019. Disponível em:
<https://repositorio.ufsc.br/handle/123456789/202842>. Acesso em: 09 out. 2023.

TSINKER, Gregory P. **Handbook of port and harbor engineering: geotechnical and structural aspects**. Florence: Springer Science, 1997.