

**INSTITUTO TECNOLÓGICO DE AERONÁUTICA**



**Gabriel Mendes Cabral Gondim**

**APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED  
ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE  
CONCRETO ARMADO SOB FLEXÃO OBLÍQUA**

Trabalho de Graduação  
2021

**Curso de Engenharia Civil-Aeronáutica**

**Gabriel Mendes Cabral Gondim**

**APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED  
ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE  
CONCRETO ARMADO SOB FLEXÃO OBLÍQUA**

Orientador

Prof. Dr. Francisco Alex Correia Monteiro (ITA)

Coorientador

Prof. Dr. Sergio Gustavo Ferreira Cordeiro (ITA)

**ENGENHARIA CIVIL-AERONÁUTICA**

**SÃO JOSÉ DOS CAMPOS**  
**INSTITUTO TECNOLÓGICO DE AERONÁUTICA**

**Dados Internacionais de Catalogação-na-Publicação (CIP)**

**Divisão de Informação e Documentação**

Gondim, Gabriel Mendes Cabral

APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE CONCRETO ARMADO SOB FLEXÃO OBLÍQUA / Gabriel Mendes Cabral Gondim.  
São José dos Campos, 2021.  
105f.

Trabalho de Graduação – Curso de Engenharia Civil-Aeronáutica– Instituto Tecnológico de Aeronáutica, 2021. Orientador: Prof. Dr. Francisco Alex Correia Monteiro. Coorientador: Prof. Dr. Sergio Gustavo Ferreira Cordeiro.

1. Concreto armado. 2. Metamodelo. 3. Simulated Annealing. I. Instituto Tecnológico de Aeronáutica. II. Aplicação do Método Augmented Simulated Annealing para Otimização de Seções de Concreto Armado sob Flexão Oblíqua.

**REFERÊNCIA BIBLIOGRÁFICA**

GONDIM, Gabriel Mendes Cabral. **APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE CONCRETO ARMADO SOB FLEXÃO OBLÍQUA**. 2021. 105f. Trabalho de Conclusão de Curso (Graduação) – Instituto Tecnológico de Aeronáutica, São José dos Campos.

**CESSÃO DE DIREITOS**

NOME DO AUTOR: Gabriel Mendes Cabral Gondim

TÍTULO DO TRABALHO: APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE CONCRETO ARMADO SOB FLEXÃO OBLÍQUA.

TIPO DO TRABALHO/ANO: Trabalho de Conclusão de Curso (Graduação) / 2021

É concedida ao Instituto Tecnológico de Aeronáutica permissão para reproduzir cópias deste trabalho de graduação e para emprestar ou vender cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte deste trabalho de graduação pode ser reproduzida sem a autorização do autor.

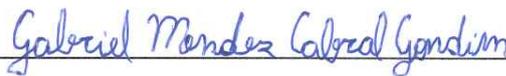
  
Gabriel Mendes Cabral Gondim

Av. H8B, 206

12.228-461 – São José dos Campos–SP

# APLICAÇÃO DO MÉTODO AUGMENTED SIMULATED ANNEALING PARA OTIMIZAÇÃO DE SEÇÕES DE CONCRETO ARMADO SOB FLEXÃO OBLÍQUA

Essa publicação foi aceita como Relatório Final de Trabalho de Graduação



---

Gabriel Mendes Cabral Gondim

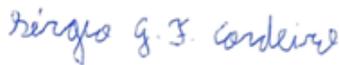
Autor



---

Francisco Alex Correia Monteiro (ITA)

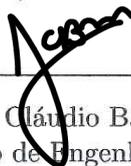
Orientador



---

Sergio Gustavo Ferreira Cordeiro (ITA)

Coorientador



---

Prof. Dr. João Cláudio Bassan de Moraes  
Coordenador do Curso de Engenharia Civil-Aeronáutica

À minha família e meus amigos, que sempre estiveram comigo, especialmente nos momentos mais difíceis da jornada.

# Agradecimentos

Primeiramente, gostaria de agradecer ao Prof. Dr. Francisco Alex Correia Monteiro, por ter me orientado, sem medir esforços, no desenvolvimento do presente trabalho.

Ao Prof. Dr. Sergio Gustavo Ferreira Cordeiro, por ter sido fundamental no desenvolvimento do código.

Ao Prof. Msc. Paulo de Tarso Machado Leite Soares, por ter lecionado a disciplina EDI-38, ponto de partida deste trabalho.

Ao Prof. Dr. Flávio Mendes Neto, por seu estudo em análise não linear de pórticos planos de concreto armado, base para o modelamento desenvolvido.

*“If something is important enough,  
even if the odds are against you, you should still do it.”*

— ELON MUSK

# Resumo

A concepção estrutural de uma edificação hoje ainda depende muito da criatividade e experiência do engenheiro responsável. Cabe a ele escolher o posicionamento de pilares, vigas e fundações que se adequem ao projeto arquitetônico e que resultem em uma estrutura barata e confiável. Feita a concepção estrutural, é prática comum utilizar softwares estruturais para que estes otimizem a utilização de aço dentro dos elementos. No entanto, ao otimizar apenas o aço, mantendo fixos os parâmetros de posição, forma e orientação das seções, é pouco provável que o mínimo global seja encontrado. O que se encontra é o mínimo para aquela configuração estrutural específica.

Pensando nisso, o presente trabalho implementa um método de otimização que leva em consideração todos os fatores mencionados acima, visando um resultado mais próximo de um mínimo global de custo. Uma seção genérica foi idealizada de modo que esta seção pode se particularizar nas seções mais utilizadas da engenharia civil, exceto seção circular. No entanto, os graus de liberdade aumentam bastante, levando à simplificação das armaduras com uma distribuição linear, para resolução em tempo hábil. Portanto, o modelo apresentado aqui se trata de um metamodelo que pretende, inicialmente, auxiliar o engenheiro a encontrar uma concepção estrutural otimizada. Porém não retorna um projeto detalhado pronto para a execução.

Durante a otimização, deve-se sempre verificar se as seções suportam os esforços solicitantes. Para isso foi implementado um modelo computacional para cálculo de esforços resistentes e verificação de seções sob a ótica do ELU (Estado Limite Último) previsto pela norma técnica brasileira NBR-6118. O modelo utilizado foi baseado no trabalho do (NETO; PIMENTA, 2000), com adição de distribuições lineares de aço e estudo da aplicação do método do gradiente conjugado para a verificação das seções.

A inclusão da distribuição linear de aço possibilitou ainda o estudo de seções não só de concreto armado, mas também seções mistas ou seções de aço, para as quais foram gerados gráficos das envoltórias de segurança de deformações e esforços.

Por fim, o modelo implementado foi utilizado em um processo de otimização com o método Augmented Simulated Annealing (AUSA) de acordo com o estudo (LEPŠ; ŠEJNOHA, 2003). Os parâmetros da seção genérica idealizada foram expressos em uma string

de binários, denominada cromossomo, que define um indivíduo em uma população. Esses cromossomos passam por mutações e reproduções, sendo selecionados os indivíduos com menor função custo e que resistem aos esforços solicitantes. Após um número predeterminado de iterações, o algoritmo retorna o indivíduo válido mais barato já visto.

# Abstract

The structural design of a building today still depends a lot on the creativity and experience of the responsible engineer. It is up to him to choose the placement of pillars, beams and foundations that suit the architectural project and that result in a cheap and reliable structure. Once the structural design is done, it is common practice to use structural software to optimize the use of steel inside the elements. However, by optimizing only the steel, keeping the position, shape and orientation parameters of the sections fixed, it is unlikely that an overall minimum will be found. What is found is the minimum for that specific structural configuration.

With that in mind, the present work implements an optimization method that takes into account all the factors mentioned above, aiming at a result closer to an overall cost minimum. A generic section was designed so that this section can be particularized in the most used sections of civil engineering, except circular section. However, the degrees of freedom are greatly increased, leading to the simplification of reinforcements with a linear distribution, for timely resolution. Therefore, the model presented here is a meta-model that intends, initially, to help the engineer to find an optimized structural design. However, it does not return a detailed project ready for execution.

During optimization, one should always check if the sections support the requesting efforts. For this purpose, a computational model was implemented to calculate resistant efforts and verify sections from the perspective of the Ultimate Limit State, foreseen by the Brazilian technical standard NBR-6118. The model implemented was based on the work of (NETO; PIMENTA, 2000), with addition of linear steel distributions and study of the application of the conjugate gradient method to verify the sections.

The inclusion of the linear steel distribution also allowed the study of sections not only of reinforced concrete, but also mixed sections or steel sections, for which graphics of the safety envelopes of deformations and stresses were generated.

Finally, the implemented model was used in an optimization process with the Augmented Simulated Annealing (AUSA) method according to the (LEPŠ; ŠEJNOHA, 2003) study. The parameters of the idealized generic section were expressed in a binary string, called a chromosome, which defines an individual in a population. These chromosomes

undergo mutations and reproductions, being selected the individuals with the lowest cost function and that resist the solicitant efforts. After a predetermined number of iterations, the algorithm returns the cheapest valid individual ever seen.

# Lista de Figuras

FIGURA 2.1 – Representação esquemática da seção transversal genérica. Concreto representado em azul e região da armadura em preto. . . . .	26
FIGURA 2.2 – Simplificação da armadura como uma distribuição linear de aço. . .	26
FIGURA 2.3 – Restrições da norma relativas ao espaçamento entre as armaduras. .	27
FIGURA 2.4 – Exemplo de Seção Quadrada. . . . .	28
FIGURA 2.5 – Exemplo de Seção Retangular. . . . .	29
FIGURA 2.6 – Exemplos de Seção Cruz. . . . .	29
FIGURA 2.7 – Exemplos de Seção T. . . . .	30
FIGURA 2.8 – Exemplos de Seção L. . . . .	31
FIGURA 2.9 – Exemplos de Seção I. . . . .	32
FIGURA 2.10 – Exemplos de Seção J (direita) e U (esquerda). . . . .	33
FIGURA 3.1 – Gráfico do diagrama tensão-deformação do concreto. (NETO; PIMENTA, 2000) . . . . .	34
FIGURA 3.2 – Diagrama tensão deformação do aço. (NETO; PIMENTA, 2000) . . . .	35
FIGURA 3.3 – Configuração deformada da seção. . . . .	37
FIGURA 4.1 – Polos de ruína definidos por norma (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2014) . . . . .	39
FIGURA 4.2 – Representação da seção genérica da seção transversal em estado deformado (NETO; PIMENTA, 2000) . . . . .	40
FIGURA 4.3 – Representação de uma iteração do método Newton-Raphson. (a) Primeira iteração (b) Segunda iteração. (GARRETT, 2015) . . . . .	41
FIGURA 4.4 – Exemplo de convergência utilizando o método Newton-Raphson. . .	43

FIGURA 4.5 – Possíveis causas de divergência no método Newton-Raphson. (YILDİRIM, ) . . . . .	43
FIGURA 4.6 – Representação de uma iteração do método do Gradiente. . . . .	45
FIGURA 4.7 – Representação de duas iterações do método do gradiente descendente, iniciando-se de dois pontos diferentes. Uma delas cai em um mínimo local (NG, 2004) . . . . .	46
FIGURA 4.8 – Aplicação dos métodos do gradiente descendente tradicional (à esquerda) e do gradiente conjugado (à direita). Percebe-se uma clara economia de iterações (JUMBO, 2011). . . . .	46
FIGURA 5.1 – Seções equivalentes linear (esquerda) e pontual (direita), que foram usadas no teste de sanidade. . . . .	49
FIGURA 5.2 – Newton-Raphson aplicado a funções com segunda derivada positiva (em cima) e negativa (embaixo). . . . .	50
FIGURA 5.3 – Emprego dos métodos do Gradiente Descendente Conjugado (superior) e Newton-Raphson (inferior). Esforços resistentes inicial em verde e objetivo em vermelho. . . . .	52
FIGURA 6.1 – Linha de iso-deformação de um estado de deformação, passando pela origem (C.G). À esquerda, sistema rotacionado. . . . .	53
FIGURA 6.2 – Fronteira de deformação da seção genérica para $\theta = 45^\circ$ . . . . .	54
FIGURA 6.3 – Esforços resistentes referentes à fronteira de deformação da seção genérica para $\theta = 45^\circ$ . . . . .	55
FIGURA 6.4 – Fronteira de deformação da seção genérica. . . . .	56
FIGURA 6.5 – Esforços resistentes referentes à fronteira de deformação da seção genérica. . . . .	57
FIGURA 6.6 – Seção retangular e sua fronteira de deformação. . . . .	59
FIGURA 6.7 – Esforços resistentes referentes à fronteira de deformação da seção retangular. . . . .	60
FIGURA 6.8 – Seção L e sua fronteira de deformação. . . . .	62
FIGURA 6.9 – Esforços resistentes referentes à fronteira de deformação da seção L. . . . .	63
FIGURA 7.1 – Translação e rotação da seção com relação ao CG. . . . .	65
FIGURA 7.2 – Crossover de dois cromossomos binários gerando um terceiro. . . . .	66

---

FIGURA 7.3 – Seção transversal resultante. . . . .	69
FIGURA 7.4 – Seção transversal resultante. . . . .	70
FIGURA 7.5 – Seção transversal resultante. . . . .	71
FIGURA 7.6 – Seção transversal resultante. . . . .	72
FIGURA A.1 – Distribuição linear de aço genérica (S). . . . .	77
FIGURA A.2 – Seções I e caixão, muito comumente utilizadas em estruturas de aço, também podem ser calculadas como distribuições lineares de aço. . .	81

## Lista de Tabelas

TABELA 5.1 – Resultados do processo de cálculo dos esforços resistentes, comparando resultados para as distribuições pontual e linear. . . . .	49
TABELA 5.2 – Resultados do processo de verificação da seção, comparando resultados para as distribuições pontual e linear. . . . .	49

# Lista de Abreviaturas e Siglas

- ELU Estado limite último.  
AUSA Augmented Simulated Annealing.

# Lista de Símbolos

$b$	Base da seção.
$h$	Altura da seção.
$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	Alturas referentes aos braços da seção genérica.
$\beta_1, \beta_2, \beta_3, \beta_4$	Comprimentos referentes aos braços da seção genérica.
$d'$	Cobrimento da armadura.
$\lambda_1, \lambda_2, \lambda_3$	Espessuras das distribuições lineares de aço.
$\theta$	Orientação da seção com relação a sua posição padrão.
$a_l$	Distanciamento mínimo entre duas armaduras.
$a_{max}$	Distanciamento máximo entre duas armaduras.
$\phi$	Diâmetro médio da brita.
$\phi$	Diâmetro médio da brita.
$\sigma_{cd}$	Tensão última de compressão de projeto do concreto.
$f_{cd}$	Resistência de cálculo do concreto.
$f_{ck}$	Resistência característica à compressão do concreto.
$\gamma_c$	Fator de segurança do concreto.
$\epsilon_{c2}$	Deformação específica de encurtamento do concreto no início do patamar plástico.
$\epsilon_{cu}$	Deformação específica de encurtamento do concreto na ruptura.
$f_{yd}$	Resistência de projeto ao escoamento do aço.
$f_{yk}$	Resistência característica de escoamento do aço.
$\gamma_s$	Fator de segurança do aço.
$E_s$	Módulo de elasticidade do aço.
$\epsilon_{yd}$	Deformação de escoamento de projeto do aço.
$\epsilon_0$	Deformação na origem do sistema de coordenadas.
$k_x$	Variação da deformação em por cm na direção x.
$k_y$	Variação da deformação em por cm na direção y.
$N_d, N_r, N_c, N_s$	Força normal de projeto, resistente, no concreto e no aço, respectivamente.
$M_{xd}, M_{xr}, M_{cx}, M_{sx}$	Momento fletor com relação ao eixo x de projeto, resistente, no concreto e no aço, respectivamente.

---

$M_{yd}, M_{yr}, M_{cy}, M_{sy}$	Momento fletor com relação ao eixo y de projeto, resistente, no concreto e no aço, respectivamente.
c	Custo por centímetro de seção construída.
i	Base maior sobre base menor de um retângulo.
$A_c$	Área da seção de concreto.
$A_a$	Área de aço na seção.
$P_c$	Perímetro da seção de concreto.

# Sumário

1	INTRODUÇÃO . . . . .	20
1.1	Objetivo . . . . .	20
1.2	Motivação . . . . .	20
1.3	Revisão Bibliográfica . . . . .	22
1.4	Organização do trabalho . . . . .	23
1.4.1	Parametrização da seção transversal . . . . .	23
1.4.2	Modelamento das seções genéricas . . . . .	23
1.4.3	Otimização de Seções com o Método Simulated Annealing . . . . .	24
2	PARAMETRIZAÇÃO DA SEÇÃO TRANSVERSAL . . . . .	25
2.1	Seção Genérica . . . . .	25
2.2	Casos Particulares . . . . .	28
2.2.1	Seção Quadrada . . . . .	28
2.2.2	Seção Retangular . . . . .	29
2.2.3	Seção Cruz . . . . .	29
2.2.4	Seção T . . . . .	30
2.2.5	Seção L . . . . .	31
2.2.6	Seção I . . . . .	32
2.2.7	Seções J e U . . . . .	32
3	CÁLCULO DOS ESFORÇOS RESISTENTES . . . . .	34
4	VERIFICAÇÃO DAS SEÇÕES . . . . .	39
5	COMPARAÇÃO DOS RESULTADOS . . . . .	48

---

6	FRONTEIRAS DO ELU PARA DIFERENTES SEÇÕES . . . . .	53
7	OTIMIZAÇÃO DE SEÇÕES COM O MÉTODO SIMULATED ANNEALING . . . . .	64
7.1	<b>Implementação do Simulated Annealing</b> . . . . .	64
7.1.1	Crossover . . . . .	65
7.1.2	Fator de Mérito . . . . .	66
7.1.3	Mutação . . . . .	67
7.1.4	Penalidades . . . . .	67
7.1.5	Função Custo . . . . .	68
7.2	<b>Resultados e Conclusões</b> . . . . .	68
	REFERÊNCIAS . . . . .	75
	APÊNDICE A – CÁLCULO DOS ESFORÇOS RESISTENTES DA DISTRIBUIÇÃO LINEAR DE AÇO. . . . .	77
	APÊNDICE B – VERIFICAÇÃO DOS ESFORÇOS RESISTENTES DA DISTRIBUIÇÃO LINEAR DE AÇO. . . . .	83
	APÊNDICE C – CÓDIGO IMPLEMENTADO EM C++. . . . .	90

# 1 Introdução

## 1.1 Objetivo

O presente trabalho tem o intuito de desenvolver um metamodelo para otimização e cálculo de custos de elementos em concreto armado. Além disso, esse metamodelo deve ser eficiente computacionalmente para que seja utilizado em ferramentas de otimização global, modificando posicionamento e seções transversais. A criação do metamodelo em si, não precisa ser computada rapidamente, uma vez que envolve muitos graus de liberdade mesmo para uma abordagem computacional. Sendo assim, o presente estudo tem a intenção encontrar seções ótimas para carregamentos dados aplicados no CG.

Pode-se dizer que o objetivo geral foi alcançado através dos seguintes objetivos específicos:

- Parametrização das seções transversais de concreto armado. A partir de uma seção genérica cujos parâmetros podem resultar em formatos variados: seção quadrada, retangular, cruz, L, I, T, J ou U;
- Parametização e simplificação geométrica linear das armaduras de aço;
- Verificação computacional das seções de acordo com o ELU;
- Otimização dos parâmetros de uma seção genérica através do método simulated annealing.

## 1.2 Motivação

Durante a fase de projeto, engenheiros civis procuram as melhores estruturas para ocuparem determinadas posições na estrutura, resistindo aos esforços requeridos àquele elemento. A decisão da “melhor” estrutura, no entanto, pode levar em consideração inúmeros fatores, entre eles fatores arquitetônicos, ambientais, energéticos ou logísticos (medido

em homens-hora). Além disso, existem inúmeras possibilidades para cada elemento escolhido no que tange à seção transversal. Por exemplo, um pilar circular, quadrado ou retangular pode ser uma boa opção no centro das estruturas; pilares em L podem ser úteis para os cantos das estruturas, enquanto outras seções de pilares (cruz, I, T, J, C ou U) podem ser as melhores em situações específicas (ANWAR; YAHYAH, 2009).

Além disso, o posicionamento dos elementos também pode resultar em estruturas não otimizadas ainda que os elementos estejam, individualmente, otimizados. Sendo assim, faz parte do projeto escolher concepções estruturais que estejam otimizadas não só localmente, mas que também abordem o problema de otimização de forma global. Desta forma, reduz-se o risco de atingir mínimos locais através de um algoritmo guloso de otimização, como seria a otimização dos elementos após fixar suas posições, formas e orientações.

Percebe-se que são inúmeras as possibilidades de escolha que o engenheiro se depara nessa situação. Tornando, portanto, o processo de tomada de decisão de forma otimizada muito difícil sem auxílio computacional. Fica a cargo do engenheiro escolher os parâmetros utilizados, estando sujeito a erros, resultando em estruturas não ótimas. Quanto menos experiente o engenheiro, menos precisas serão suas tentativas de concepção estrutural. Ainda que se recorra aos softwares atualmente disponíveis, estas ferramentas modelam uma pre-concepção do engenheiro, otimizando armaduras e verificando as normas coerentes, sem sugerir concepções melhores que resultariam em um menor custo total da obra.

Nesse contexto, seria muito interessante que houvesse uma ferramenta computacional que sugerisse estruturas que otimizassem o posicionamento dos elementos, bem como o formato da seção transversal e a armadura utilizada, ainda que como um metamodelo. Essa ferramenta poderia utilizar como função objetivo o custo total financeiro, ecológico, energético ou mesmo uma ponderação dos mesmos, de acordo com os interesses do projetista.

Quando se trata de concreto armado, lidamos com um material compósito com inúmeras configurações possíveis de formato ou armaduras. Uma mesma seção de concreto pode resistir ou não, dependendo do aço utilizado. Logo, para cada seção de concreto armado testada, deve ser otimizada a armadura utilizada de modo a utilizar a mínima área de aço possível, ou próximo disso.

Esse é o tipo de otimização tradicionalmente utilizada nos softwares estruturais de concreto armado e esse método, por si só, é computacionalmente custoso pois envolve o posicionamento de  $n$  barras com bitolas e posicionamentos diferentes. Além disso, a ferramenta hipotética de otimização global deveria testar várias seções diferentes para cada elemento. Sendo assim, o objetivo do presente estudo é desenvolver um metamodelo computacionalmente eficiente para obter e minimizar custos financeiros envolvidos na

construção de elementos com variadas seções transversais.

### 1.3 Revisão Bibliográfica

A temática de otimização estrutural de estruturas de concreto armado na engenharia civil é uma temática abordada por diversos autores, variando métodos computacionais e os alvos da otimização. Em especial de destacam quatro temas principais: eficiência dos materiais, eficiência em custo e materiais utilizados, eficiência ambiental das obras e eficiência de projeto (AFZAL *et al.*, 2020). Dentre essas, o presente trabalho pretende diminuir custos a partir de uma menor utilização de materiais, respeitando as normas técnicas vigentes, em especial a NBR - 6118 (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2014).

Dentro do tema de otimização de custos, mas não exclusivamente contidos, temos a otimização de pórticos tridimensionais modelada como um clássico problema de mochila (SHARAFI *et al.*, 2013) e (SHARAFI *et al.*, 2015) onde os autores otimizam o posicionamento dos pilares da estrutura de um edifício utilizando a largura e o comprimento do mesmo como limitadores da soma das distâncias entre as colunas. A otimização do posicionamento das colunas é uma otimização que muitas vezes é feita com base em tentativa e erro, sendo assim, é um ótimo ponto de partida para o presente trabalho.

As estruturas a serem otimizadas também podem ser modeladas como indivíduos em populações dentro de um algoritmo genético, como fez (TOĞAN; DALOĞLU, 2006) ao otimizar treliças submetidas a dados carregamentos, como por exemplo o de fios de transmissão nas torres de transmissão metálicas. Ou então para a otimização de lajes. Já (KOCER; ARORA, 1996) otimizou torres de transmissão em concreto armado utilizando também algoritmos genéticos.

Combinando os métodos de otimização por algoritmos genéticos e o algoritmo simulated annealing, (LEPŠ; ŠEJNOHA, 2003) utilizou o método AUSA (Augmented Simulated Annealing) para otimizar vigas retangulares submetidas a flexão simples. O método AUSA combina elementos dos algoritmos genéticos, como por exemplo a existência de cromossomos que definem um indivíduo em uma população, com elementos do método simulated annealing, como por exemplo o aquecimento da temperatura de tempos em tempos, que facilita a "fuga" de mínimos locais por parte das populações.

Existe também uma lista extensa de trabalhos que utilizam métodos de machine learning, como redes neurais, para otimizar estruturas de concreto armado. Um exemplo é a otimização de lajes parametrizadas (de forma similar ao que será aplicado na parametrização das seções transversais no presente trabalho) por (FERREIRO-CABELLO *et al.*, 2018). Este último trabalho foca na redução de impactos ambientais.

Dentre os métodos mencionados e diversos outros já utilizados na área da engenharia civil ou não, o método AUSA foi escolhido devido à sua aplicação com êxito em (LEPŠ; ŠEJNOHA, 2003), que aborda um problema semelhante ao aqui desenvolvido. Além disso, para futuros estudos e agilidade de otimização na prática, a otimização aqui implementada poderia ser utilizada para a criação de uma base de dados e desenvolvimento de uma rede neural treinada e eficiente.

Antes de fazer qualquer otimização, porém, é preciso modelar as estruturas que serão otimizadas. Dado que o presente trabalho visa uma busca entre diversas seções transversais possíveis no mercado (como enumerado, parametrizado e estudado em (ANWAR; YAHYAH, 2009)), foi pensada uma parametrização genérica e eficiente na utilização com o método simulated annealing.

Além disso, a modelagem de esforços resistentes e verificação das seções submetidas a dados carregamentos com o método Newton-Raphson desenvolvida em (NETO; PIMENTA, 2000) foi essencial e embasou boa parte da modelagem aqui implementada, acrescentando-se a distribuição linear e testando a possibilidade de utilização do método do gradiente conjugado (FLETCHER; REEVES, 1964) para reduzir falsos negativos e aumentar a velocidade de convergência da otimização.

## 1.4 Organização do trabalho

### 1.4.1 Parametrização da seção transversal

O capítulo 2 contém uma descrição detalhada de uma seção genérica de concreto armado. Explicando como essa seção genérica pode se particularizar em quaisquer dos casos mais conhecidos de seções, como seção retangular, em L, em T, entre outras. Além disso, é introduzida uma simplificação no modelamento de armadura que será conveniente nos processos de otimização.

### 1.4.2 Modelamento das seções genéricas

No capítulo 3 apresentamos o comportamento adotado para os materiais utilizados e uma descrição superficial do cálculo dos esforços resistentes para uma dada configuração deformada da seção, assim como descrito em (NETO; PIMENTA, 2000), introduzindo o cálculo de esforços resistentes para uma distribuição linear de aço. Os cálculos detalhados estão nos Apêndices.

O capítulo 4 faz o caminho inverso do capítulo 3: dados os esforços solicitantes, que devem ser iguais aos resistentes no equilíbrio, busca-se a configuração deformada da seção.

São levantadas duas formas de buscar a solução: método Newton Raphson e método do gradiente conjugado, apontando as vantagens e desvantagens deles.

A simplificação introduzida para a distribuição de armaduras é posta à prova no capítulo 5, assim como o código desenvolvido. Nele, são comparados os resultados obtidos com o software nFOCCA (MEDEIROS, 2004), com os resultados do modelo desenvolvido aqui (armadura contínua simplificada). Além disso, os resultados do emprego do Newton-Raphson e método do Gradiente Descendente são confrontados, escolhendo-se o que performou melhor para que seja utilizado no decorrer do trabalho.

No capítulo 6, o modelo desenvolvido até aqui é utilizado para obter superfícies de segurança para as seções, no espaço tridimensional das deformações ou dos esforços. São mostradas fronteiras do ELU para formatos variados de seção.

### **1.4.3 Otimização de Seções com o Método Simulated Annealing**

No capítulo 7, a parametrização das seções é expressa por uma cadeia de binários denominada cromossomos. Diversos indivíduos (seções expressas por seus cromossomos) são gerados aleatoriamente e passam por operações de reprodução e mutação, bem como seleção daqueles indivíduos menos custosos e que resistem aos esforços solicitantes.

Ao fim do processo de otimização, o indivíduo menos custoso é retornado e é analisada a coerência e performance do modelo, bem como idealizados trabalhos futuros que podem explorar ainda mais o modelo desenvolvido.

## 2 Parametrização da Seção Transversal

### 2.1 Seção Genérica

Como mencionado anteriormente, existem muitas seções transversais possíveis para os elementos estruturais de concreto armado. Com o intuito de construir um metamodelo para otimizar essas seções, é interessante criar uma forma de definição única. Que possa definir quaisquer tipos de seção com base nos mesmos parâmetros, de modo que os tipos específicos de seção transversal sejam apenas particularizações de uma seção genérica.

Nesse sentido, o seguinte modelo foi criado para generalizar todas as seções formadas por associações de retângulos. Isto é, todas as seções mencionadas exceto a seção circular. Para isso, são necessários os parâmetros relativos às dimensões de cada retângulo (três: um vertical e dois horizontais) e suas posições relativas ao retângulo vertical. Ou seja, seis parâmetros de dimensões e quatro parâmetros de posição.

Além disso, deve-se definir o posicionamento das armaduras de aço na seção de concreto armado. Por ora, definir-se-á apenas o cobrimento delas e serão representadas abaixo por retângulos pretos, um para cada retângulo que define a seção. Sendo assim, pode-se representar a seção genérica conforme a figura 2.1.

Essa representação foi pensada de modo a exprimir os parâmetros da seção em função de uma base e uma altura de referência, a partir das quais são obtidas outras dimensões através de multiplicadores, como os coeficientes  $\alpha$  e  $\beta$ . Mais adiante, na seção de Otimização de Seções pelo método Simulated Annealing, ficará clara a conveniência dessa parametrização (entre inúmeras possíveis) para o método de otimização utilizado. Também serão definidos os limites inferior e superior de cada parâmetro.

Resta apenas a definição das armaduras utilizadas na seção para que ela seja completamente definida e sejam possíveis cálculos de esforços resistentes, verificação do ELU, entre outros. No entanto, a definição das armaduras utilizadas é ainda mais livre que a definição da seção dado que há inúmeras bitolas disponíveis, espaçamentos podem variar, número de barras, camadas, assimetrias entre outras possibilidades. Sendo assim, é impraticável analisar todas as configurações possíveis de armadura para cada seção estudada por haver

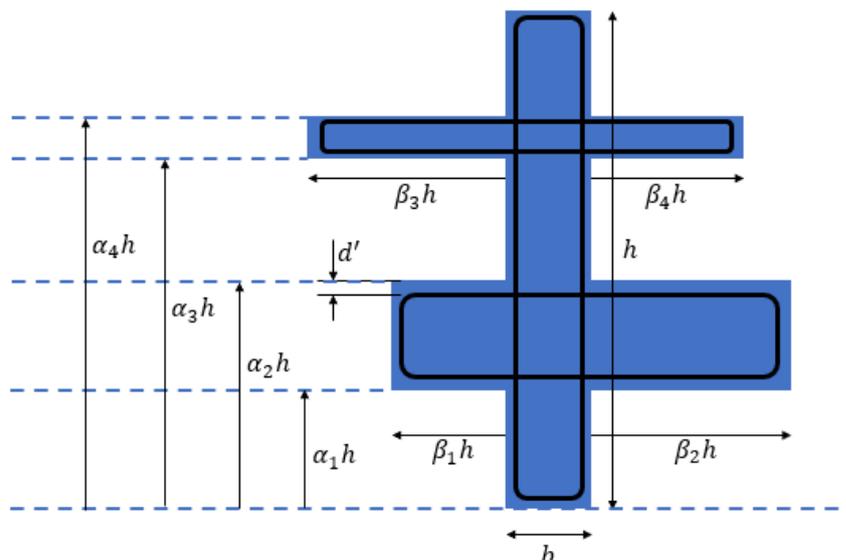


FIGURA 2.1 – Representação esquemática da seção transversal genérica. Concreto representado em azul e região da armadura em preto.

um custo computacional altíssimo. Além do fato de que muitas das configurações possíveis não são viáveis construtivamente.

Como o presente trabalho se trata de um metamodelo, visando sugerir concepções estruturais ao engenheiro sem necessariamente fazer o detalhamento das seções, é válido fazer simplificações no problema. Para isso, a configuração de armaduras foi simplificada como uma distribuição linear uniforme, substituindo as barras pontuais utilizadas ao longo do perímetro da seção. Com isso, cada retângulo utilizado na seção possui sua armadura simplificada definida pelo cobrimento e pela espessura da distribuição linear. Conforme mostrado na figura 2.2

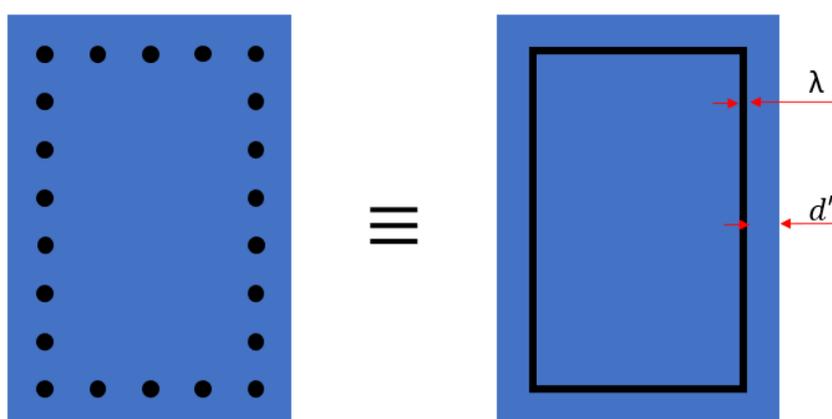


FIGURA 2.2 – Simplificação da armadura como uma distribuição linear de aço.

O cobrimento é determinado pela norma NBR 6118 e depende da classe de agressividade ambiental que a estrutura está exposta. Na grande maioria dos casos, elementos

estruturais estão expostos a uma mesma classe de agressividade ambiental em todas as suas faces. Portanto, o cobrimento será considerado o mesmo para todos os três retângulos que definem a seção transversal genérica e esta ficará completamente determinada quando especificados os parâmetros  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$  e a otimização da área de aço será simplificada para a busca de uma tripla ordenada que minimize a área, obedecendo o ELU.

Para limitar o espaço de busca, utilizaremos a norma referente aos espaçamentos entre as armaduras e as bitolas mínimas e máximas.

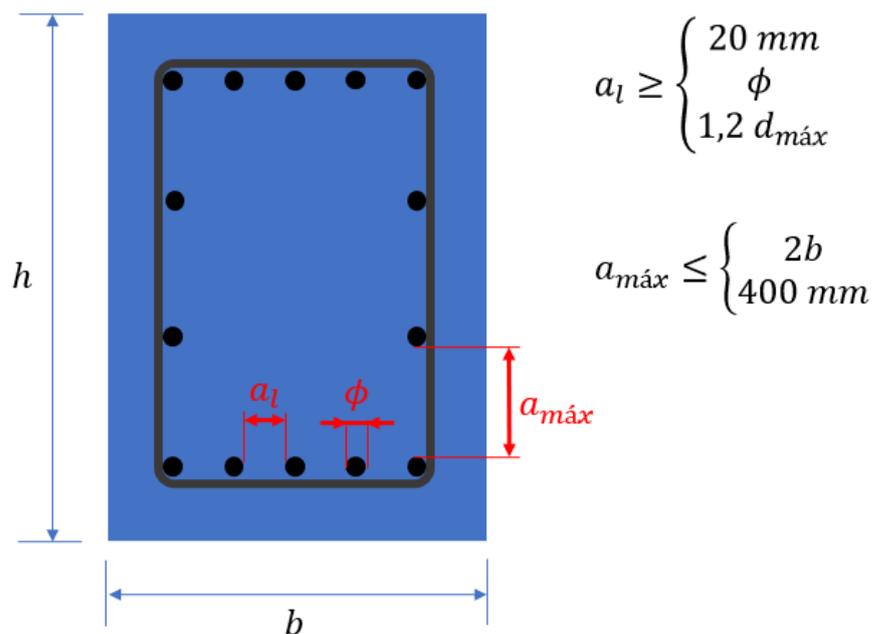


FIGURA 2.3 – Restrições da norma relativas ao espaçamento entre as armaduras.

- Espessura Mínima:

Para a situação de menor espessura possível, utilizaremos a menor bitola comercial disponível (6,3 mm) associada ao maior espaçamento possível (400 mm), desconsiderando a dimensão da seção. Para calcular a espessura equivalente, basta igualar a área de aço da distribuição linear com a área de aço da distribuição pontual:

$$e \cdot 400 = \pi \cdot \left(\frac{6,3}{2}\right)^2$$

$$e = 0,0779 \text{ mm}$$

- Espessura Máxima:

Para a situação de maior espessura possível, utilizaremos a maior bitola comercial disponível (40 mm) associada ao menor espaçamento possível (40 mm), desconsiderando o diâmetro do agregado. Cálculo análogo ao anterior:

$$e \cdot 40 = \pi \cdot \left(\frac{40}{2}\right)^2$$
$$e = 31,41 \text{ mm}$$

Perceba que essas contas são simplificações da norma, que prevê cálculos adicionais com relação as áreas máxima e mínima, que levam em conta fatores variáveis como a área da seção.

## 2.2 Casos Particulares

Como mencionado, a seção genérica deve possuir como casos particulares todas as outras seções mencionadas, exceto a circular. Vejamos as particularizações.

### 2.2.1 Seção Quadrada

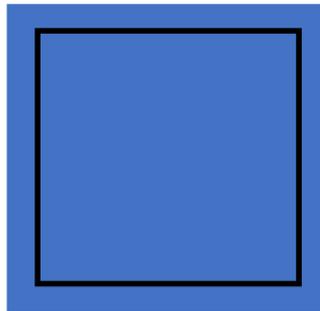


FIGURA 2.4 – Exemplo de Seção Quadrada.

São seções duplamente simétricas, ou seja, que não privilegiam nenhum sentido de solicitação de esforços por apresentarem o mesmo momento de inércia nas direções horizontal e vertical. A particularização paramétrica que resulta em seções quadradas é a seguinte:

$$h = b$$

$$\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$$

$$\lambda_2 = \lambda_3 = 0$$

### 2.2.2 Seção Retangular

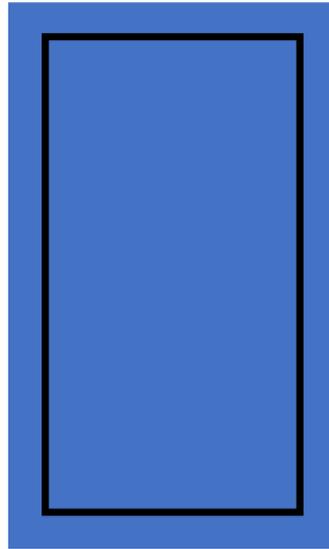


FIGURA 2.5 – Exemplo de Seção Retangular.

São seções monossimétricas que privilegiam um sentido de solicitação de esforços, pois possuem maior inércia associada à flexão em torno da sua menor dimensão. Por privilegiarem uma direção, costuma-se projetar plantas estruturais que equilibrem o momento inercial nas duas direções principais da planta. São largamente utilizadas na maior parte das estruturas de concreto armado. A particularização paramétrica que resulta em seções retangulares é a seguinte:

$$\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$$

$$\lambda_2 = \lambda_3 = 0$$

### 2.2.3 Seção Cruz

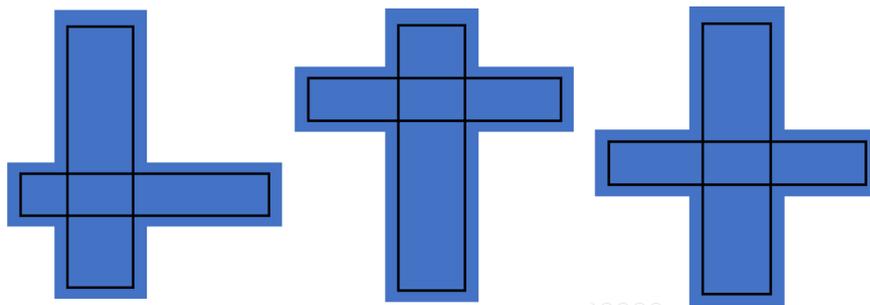


FIGURA 2.6 – Exemplos de Seção Cruz.

Seções Cruz podem ser duplamente simétricas, monossimétricas ou assimétricas. Por combinarem dois retângulos, podem privilegiar a direção que o projetista quiser, aumentando ou diminuindo as dimensões dos retângulos. Podem ser ideais em cruzamentos de paredes ou por fins estéticos. A particularização paramétrica que resulta em seções cruz é a seguinte:

- Para ser Seção Cruz:

$$\beta_3 = \beta_4 = 0$$

$$\lambda_3 = 0$$

- Para ser Simétrica:

$$\beta_1 = \beta_2$$

- Para ser Duplamente Simétrica:

$$\frac{\alpha_1 + \alpha_2}{2} = 0.5$$

### 2.2.4 Seção T

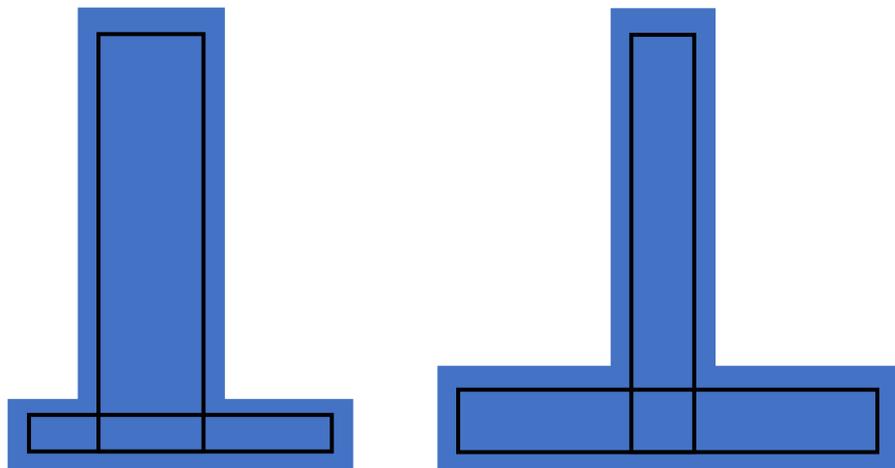


FIGURA 2.7 – Exemplos de Seção T.

Seções em T podem ser monossimétricas. São casos particulares da seção cruz. Assim como as anteriores, por combinarem dois retângulos, podem privilegiar a direção que o projetista quiser, assim como a seção anterior. São muito utilizadas em vigas submetidas a flexão simples ou lajes, mas também podem ser utilizadas em pilares. A particularização paramétrica que resulta em seções T é:

- Para ser Seção T, mesmos critérios para a cruz, mas também:

$$\alpha_1 = 0$$

- Para ser Simétrica:

$$\beta_1 = \beta_2$$

### 2.2.5 Seção L

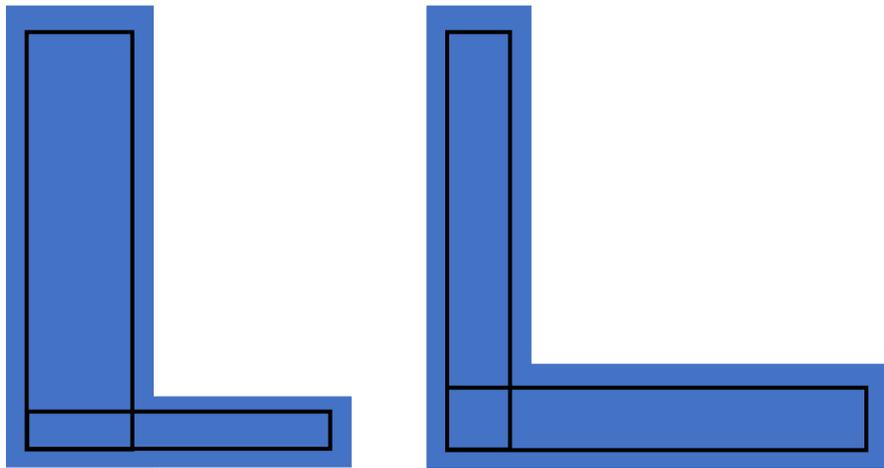


FIGURA 2.8 – Exemplos de Seção L.

Seções em L podem possuir simetria diagonal. São casos particulares da seção T. Assim como as anteriores, por combinarem dois retângulos, podem privilegiar a direção que o projetista quiser, assim como a seção anterior. São muito utilizadas em nos cantos das edificações. A particularização paramétrica que resulta em seções L é:

- Para ser Seção L, mesmos critérios da seção cruz, mas também:

$$\alpha_1 = 0$$

$$\beta_1 = 0$$

- Para ser Simétrica:

$$h = b + \beta_2 \cdot h$$

$$b = \alpha_2 \cdot h$$

### 2.2.6 Seção I

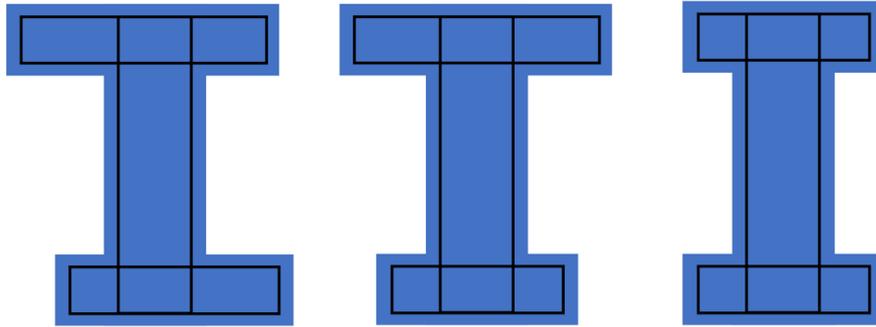


FIGURA 2.9 – Exemplos de Seção I.

Seções I podem ser duplamente simétricas, monossimétricas ou assimétricas. Por combinarem três retângulos, podem privilegiar a direção que o projetista quiser, aumentando ou diminuindo as dimensões dos retângulos. Podem ser ideais em cruzamentos de paredes ou por fins estéticos. A particularização paramétrica que resulta em seções I é a seguinte:

- Para ser simétrica:

$$\beta_1 = \beta_2$$

$$\beta_3 = \beta_4$$

- Para ser duplamente simétrica:

$$\alpha_2 = 1 - \alpha_3$$

$$\beta_2 = \beta_3$$

- Para ser duplamente simétrica:

$$h_2 = h_3$$

$$b_2 = b_3$$

### 2.2.7 Seções J e U

Seções I podem ser duplamente simétricas, monossimétricas ou assimétricas. Por combinarem três retângulos, podem privilegiar a direção que o projetista quiser, aumentando ou diminuindo as dimensões dos retângulos. Podem ser ideais em cruzamentos de paredes, envolvendo elevadores ou por fins estéticos. A particularização paramétrica que resulta em seções I é a seguinte:

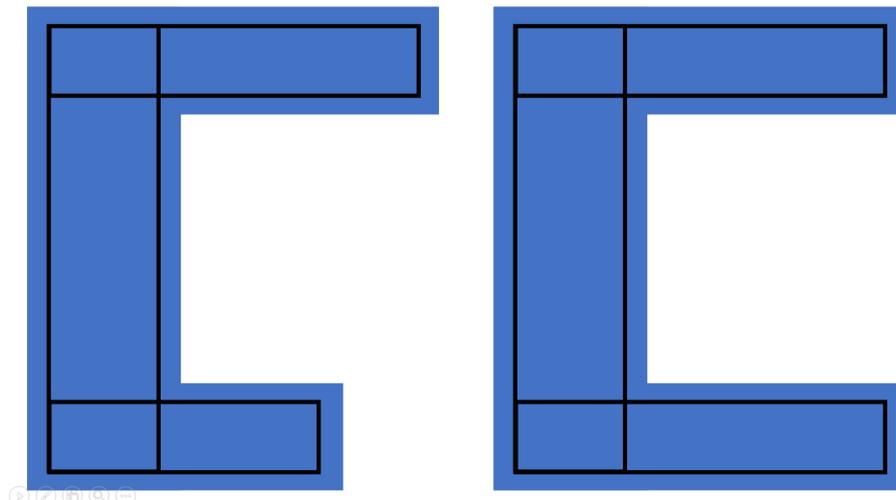


FIGURA 2.10 – Exemplos de Seção J (direita) e U (esquerda).

- Para ser Seção J:

$$\alpha_1 = 0$$

$$\beta_1 = 0$$

$$\alpha_4 = 1$$

$$\beta_3 = 0$$

- Para ser Seção U:

$$\beta_2 = \beta_4$$

- Para ser Seção U Simétrica:

$$\alpha_2 = 1 - \alpha_3$$

### 3 Cálculo dos Esforços Resistentes

Uma vez definida completamente a seção transversal do elemento, ou seja, definidas as dimensões e posicionamento das armaduras, é possível calcular os esforços resistentes analiticamente para quaisquer configurações deformadas. Esse cálculo será utilizado posteriormente para a verificação das seções perante o ELU. Esta última, por sua vez, será utilizada mais adiante para a otimização da área de aço da seção.

Sabe-se que o diagrama tensão-deformação do concreto é definido pela NBR 6118 conforme representado abaixo:

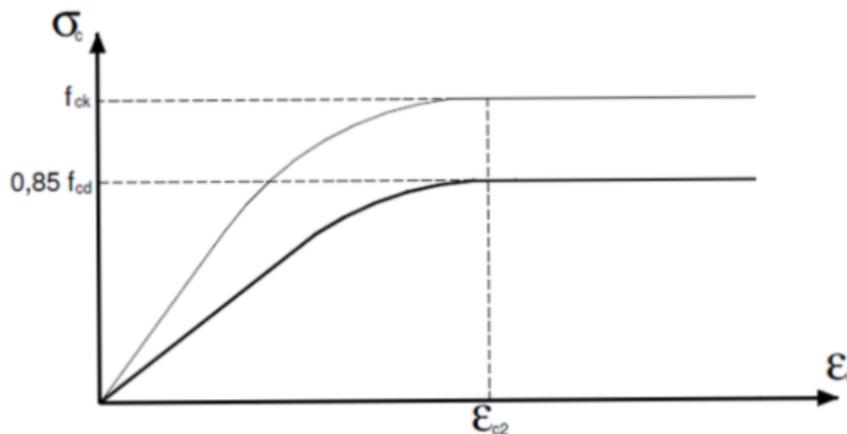


FIGURA 3.1 – Gráfico do diagrama tensão-deformação do concreto. (NETO; PIMENTA, 2000)

Onde as fórmulas que definem o diagrama tensão deformação do concreto, em função da classe do concreto, são:

$$\sigma_{cd} = 0,85 f_{cd} = 0,85 \frac{f_{ck}}{\gamma_c}$$

$$\gamma_c = 1,40$$

$$\sigma = \begin{cases} 0 & \epsilon \leq 0 \\ \sigma_{cd} \left[ 1 - \left( 1 - \frac{\epsilon}{\epsilon_{c2}} \right)^n \right] & 0 \leq \epsilon \leq \epsilon_{c2} \\ \sigma_{cd} & \epsilon \geq \epsilon_{c2} \end{cases}$$

$$\epsilon_{c2} = \begin{cases} 2 & \text{até C50, inclusive} \\ 2 + 0,85 (f_{ck} - 50)^{0,53} & \text{entre C50 e C90} \end{cases}$$

$$n = \begin{cases} 2 & \text{até C50, inclusive} \\ 1,4 + 23,4 \left( \frac{90 - f_{ck}}{100} \right)^4 & \text{entre C50 e C90} \end{cases}$$

Já para o aço, temos o seguinte diagrama tensão-deformação:

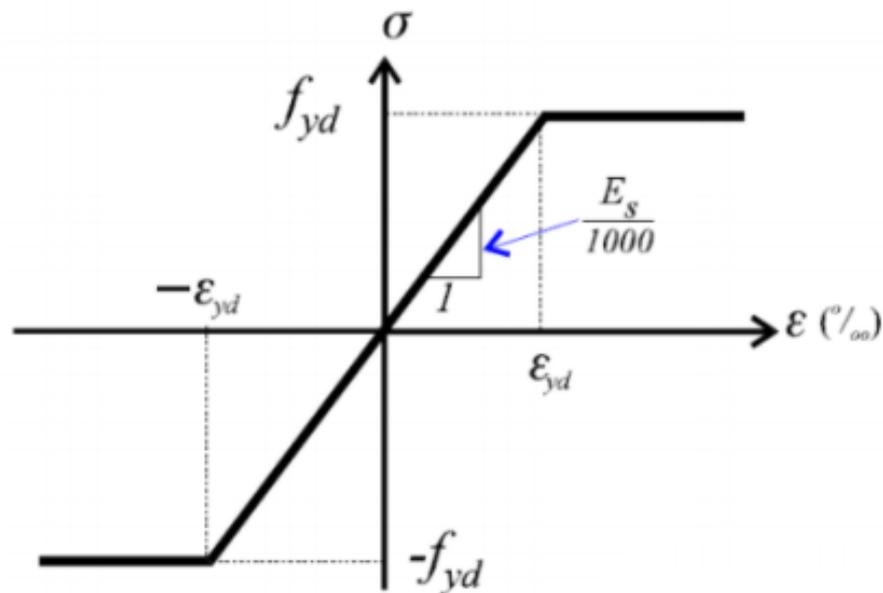


FIGURA 3.2 – Diagrama tensão deformação do aço. (NETO; PIMENTA, 2000)

Onde as fórmulas que definem o diagrama tensão deformação do aço, em função da resistência característica do aço e do módulo de elasticidade, são:

$$f_{yd} = \frac{f_{yk}}{\gamma_s}$$

$$\gamma_s = 1,15$$

$$E_s = 210 \text{ GPa} = 2,1 \cdot 10^6 \text{ kgf/cm}^2$$

$$\epsilon_{yd} = 1000 \frac{f_{yd}}{E_s}$$

$$\sigma = \begin{cases} -f_{yd} & \epsilon \leq -\epsilon_{yd} \\ f_{yd} \frac{\epsilon}{\epsilon_{yd}} & |\epsilon| \leq \epsilon_{yd} \\ f_{yd} & \epsilon \geq \epsilon_{yd} \end{cases}$$

A partir destas definições, podemos calcular a tensão em qualquer ponto material da seção (concreto ou aço) desde que saibamos a deformação  $\epsilon$ . Utilizando o modelo de viga de Euler-Bernoulli, assumindo que a seção se mantém plana e perpendicular à linha neutra, precisamos de apenas 3 valores para expressar a deformação (e conseqüentemente a tensão) em qualquer ponto da seção: a deformação em um ponto, por exemplo o centro de massa da seção, e as inclinações do plano relativas aos eixos x e y.

Quanto às inclinações, definiremos vetores de deslocamento angular nos eixos x e y, os quais definirão as inclinações (derivadas parciais), observe:

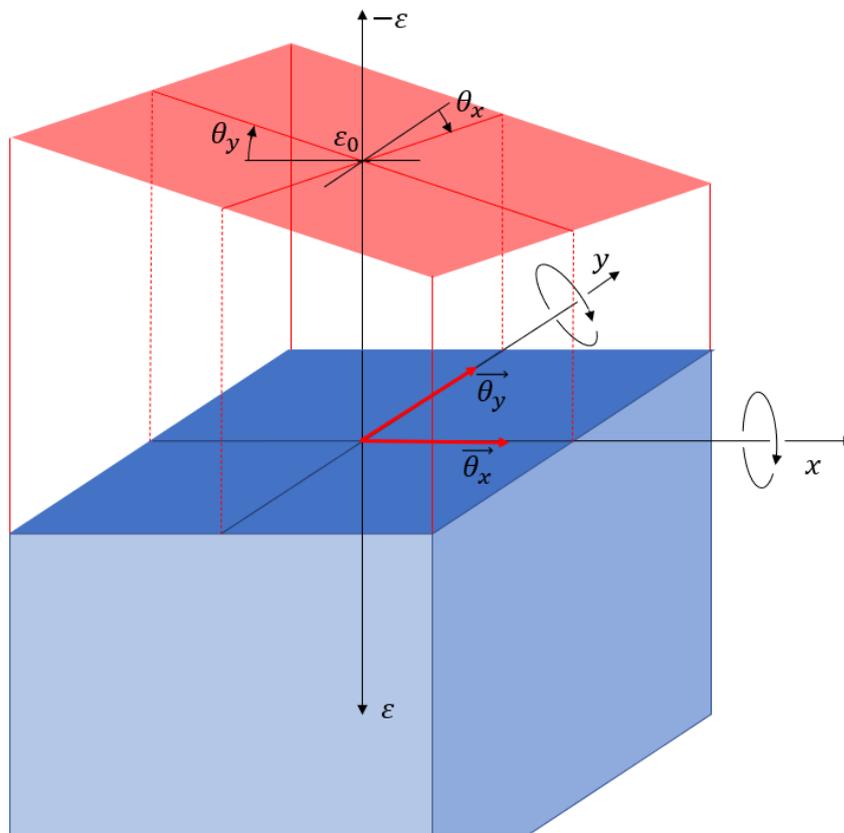


FIGURA 3.3 – Configuração deformada da seção.

Perceba que o sentido positivo da deformação é o da compressão da seção, assim como o sentido positivo das tensões e forças. Logo, o sentido negativo é o da tração.

Definindo então os coeficientes angulares como  $k_x$  e  $k_y$  :

$$\tan(\theta_x) \stackrel{N}{=} k_x$$

$$\tan(\theta_y) \stackrel{N}{=} k_y$$

Temos:

$$\varepsilon = \varepsilon_0 + k_y x - k_x y$$

Além disso, pela teoria de Euler-Bernoulli, pode-se demonstrar que:

$$k_x = \frac{1}{R_x}$$

$$k_y = \frac{1}{R_y}$$

Onde  $R_x$  e  $R_y$  são os raios de curvatura da viga em relação aos eixos x e y, respecti-

vamente.

Ora, se definirmos  $\varepsilon_0$ ,  $k_x$  e  $k_y$  definiremos, portanto o estado de deformação em todos os pontos da seção e podemos, por consequência, calcular através de cálculo integral os esforços resistentes força normal (N), momento em relação ao eixo x ( $M_x$ ) e momento em relação ao eixo y ( $M_y$ ), através das seguintes equações.

Separemos em dois casos, esforços resistentes devido ao concreto (subscrito c) e ao aço (subscrito s):

$$\begin{aligned}N_c &= \iint_A \sigma_c(\varepsilon) dA \\M_{cx} &= - \iint_A \sigma_c(\varepsilon) y dA \\M_{cy} &= \iint_A \sigma_c(\varepsilon) x dA\end{aligned}$$

No entanto, em se tratando de seções de concreto muito variadas, torna-se complicado fazer a integral de área de forma genérica. Podemos abordar o problema utilizando o Teorema de Green, simplificando a integral dupla para uma simples integral de linha sobre a fronteira da seção transversal. A dedução pode ser encontrada em (NETO; PIMENTA, 2000), para as expressões dos esforços resistentes do concreto, em função da geometria da seção e do estado de deformação ( $\varepsilon_0$ ,  $k_x$ ,  $k_y$ ).

No caso da seção das armaduras, que foi simplificada neste estudo como uma distribuição linear, a aplicação da integral de linha é direta e muito similar à realizada para o concreto. Neste caso, porém, utilizamos os retângulos que definem as armaduras simplificadas, bem como a espessura da distribuição linear. A dedução para este caso pode ser encontrada no Apêndice A.

Dessa forma, conseguimos calcular os esforços resistentes na seção transversal em função das características geométricas da seção e de seu estado de deformação. Na seção seguinte, faremos o caminho inverso, isto é, dados os esforços calcularemos o estado de deformação.

## 4 Verificação das Seções

Na seção anterior, discutimos e calculamos no Apêndice A funções do tipo  $f(\varepsilon_0, k_x, k_y) = (N, M_x, M_y)$  dadas as características da seção. Na prática, porém, modelam-se os elementos e carregamentos, calculam-se os esforços para então descobrir o estado de deformação e descobrir se é aceitável ou não (estado de ruína segundo o Estado Limite Último – ELU).

Da teoria relativa ao concreto armado, existem 3 critérios para determinar ruína de uma determinada seção. São eles:

Deformação de alongamento máxima do aço:  $-10\%$

Deformação de compressão máxima do concreto:  $\varepsilon_{cu}$

Deformação de compressão máxima do concreto na fibra à altura  $\frac{\varepsilon_{cu} - \varepsilon_{c2}}{\varepsilon_{cu}} h$ :  $\varepsilon_{c2}$

Ilustrando os polos de ruína abaixo:

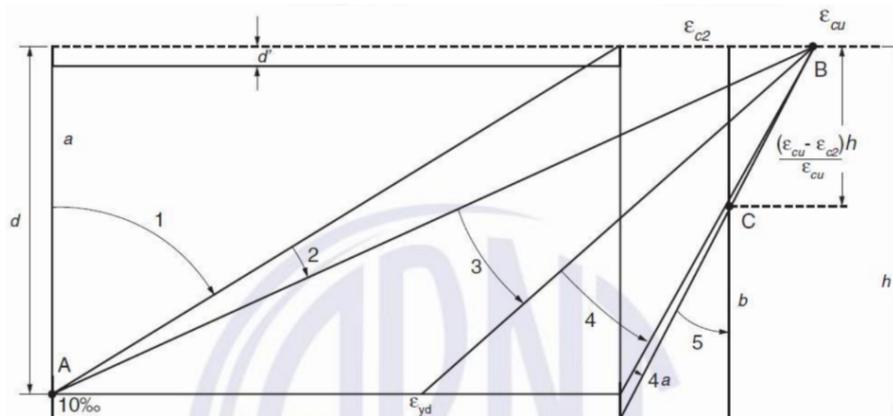


FIGURA 4.1 – Polos de ruína definidos por norma (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2014)

No caso de uma seção genérica, a altura a ser considerada é a altura máxima na direção perpendicular à linha neutra da seção. Nos pontos extremos da seção, teremos as máximas deformações. Veja a figura abaixo:

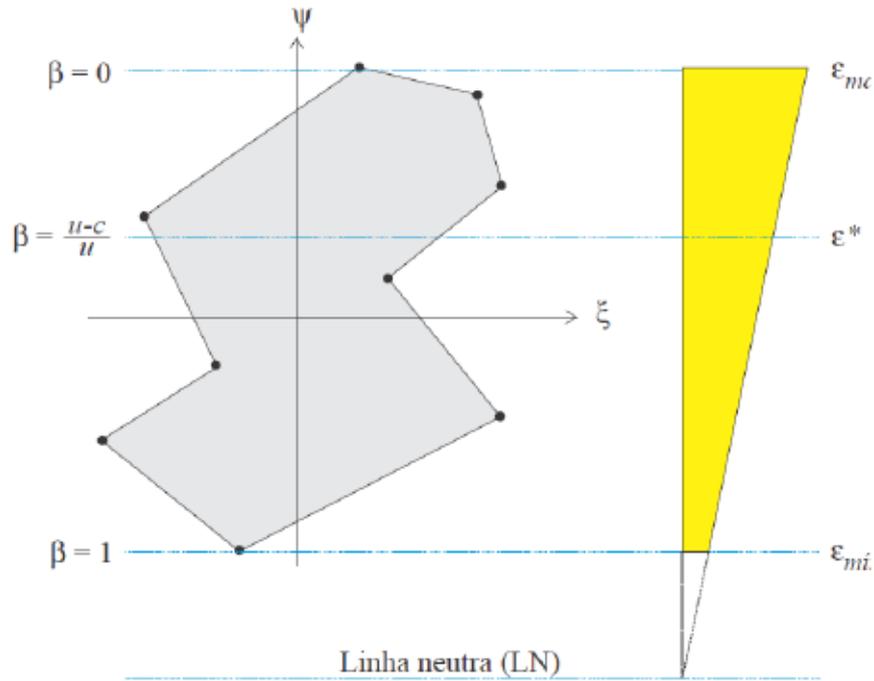


FIGURA 4.2 – Representação da seção genérica da seção transversal em estado deformado (NETO; PIMENTA, 2000)

Para obedecer aos dois primeiros polos de ruína, basta que os vértices extremos do aço e do concreto obedçam as deformações máximas possíveis, isto é:

$$\varepsilon_{s, \text{mín}} \geq -10 \text{ ‰}$$

$$\varepsilon_{c, \text{máx}} = \varepsilon_{\text{máx}} \leq u$$

Já para o terceiro polo de ruína, calculamos a deformação na fibra à altura  $\frac{\varepsilon_{cu} - \varepsilon_{c2}}{\varepsilon_{cu}} h$  e a limitamos:

$$\varepsilon^* = \varepsilon_{\text{máx}} - \frac{\varepsilon_{cu} - \varepsilon_{c2}}{\varepsilon_{cu}} (\varepsilon_{\text{máx}} - \varepsilon_{\text{mín}})$$

$$\varepsilon^* = \frac{(\varepsilon_{cu} - \varepsilon_{c2}) \varepsilon_{\text{mín}} + \varepsilon_{c2} \varepsilon_{\text{máx}}}{\varepsilon_{cu}} \leq \varepsilon_{c2}$$

$$(\varepsilon_{cu} - \varepsilon_{c2}) \varepsilon_{\text{mín}} + \varepsilon_{c2} \varepsilon_{\text{máx}} \leq \varepsilon_{c2} \varepsilon_{cu}$$

Nosso objetivo para esta seção é fazer o caminho inverso da seção anterior, isto é, dados os esforços  $(N, M_x, M_y)$  calcular o estado de deformações  $(\varepsilon_0, k_x, k_y)$  e em seguida verificarmos se as deformações obedecem às três desigualdades mostradas para o ELU. Como nós já possuímos a função  $f(\varepsilon_0, k_x, k_y) = (N, M_x, M_y)$ , nosso problema consiste então em achar a raiz da equação:

$$f(\varepsilon_0, k_x, k_y) - (N, M_x, M_y) = 0$$

Temos então um ponto em um espaço imagem tridimensional e queremos achar o ponto correspondente em um domínio tridimensional. Para abordar este problema, serão abordados dois métodos numéricos diferentes: Newton-Raphson e método do Gradiente Conjugado. Lembrando que, como esse problema é apenas uma pequena etapa no processo de otimização e este, por sua vez, será repetido para milhões de casos diferentes para a elaboração da base de dados, é fundamental que a resolução do problema não seja muito custosa computacionalmente.

Antes de aplicarmos os métodos de forma propriamente dita, é importante levantar as características dos dois métodos e as características do problema em questão, pois não existe método computacional melhor que os outros em todos os problemas possíveis.

Primeiramente, o método Newton-Raphson é um método tradicional para descobrir raízes e o faz vasculhando o domínio a “passos largos”. A vantagem de dar passos largos é que, em casos favoráveis, a convergência é alcançada de forma mais rápida, reduzindo o custo computacional. Por outro lado, quando se dá passos largos no domínio, as características da função podem mudar abruptamente entre uma iteração e a próxima. Para esclarecer o que foi dito, veja o seguinte pseudo-código do método Newton-Raphson:

1. Define-se um ponto inicial  $x_i$  ( $i=0$ ) no domínio.
2. Calcula-se a imagem  $f(x_i)$  e a derivada  $f'(x_i)$ .
3. Calcula-se o próximo ponto do domínio:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Trata-se da interseção da tangente à função no ponto  $(x_i, f(x_i))$  e o eixo das abscissas. Conforme a figura abaixo:

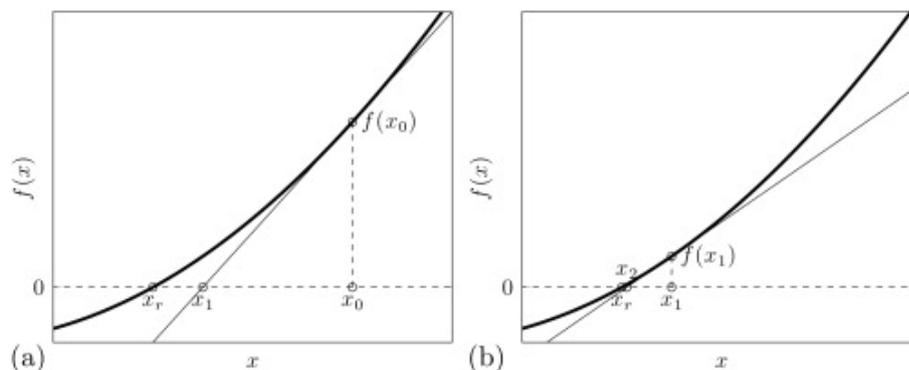


FIGURA 4.3 – Representação de uma iteração do método Newton-Raphson. (a) Primeira iteração (b) Segunda iteração. (GARRETT, 2015)

Na situação em questão, temos:

$$x_{i+1} = x_i + [\nabla f(x_i)]^{-1} f(x_i)$$

Onde (NETO; PIMENTA, 2000):

$$f(x_i) = \begin{Bmatrix} N - N_R \\ M_x - M_{xR} \\ M_y - M_{yR} \end{Bmatrix}$$

$$x_i = \begin{Bmatrix} \epsilon_0 \\ k_x \\ k_y \end{Bmatrix}$$

$$\nabla f(x_i) = \begin{bmatrix} \frac{\partial N_R}{\partial \epsilon_0} & \frac{\partial N_R}{\partial k_x} & \frac{\partial N_R}{\partial k_y} \\ \frac{\partial M_{xR}}{\partial \epsilon_0} & \frac{\partial M_{xR}}{\partial k_x} & \frac{\partial M_{xR}}{\partial k_y} \\ \frac{\partial M_{yR}}{\partial \epsilon_0} & \frac{\partial M_{yR}}{\partial k_x} & \frac{\partial M_{yR}}{\partial k_y} \end{bmatrix}$$

A dedução de cada uma dessas expressões para a distribuição linear de aço se encontram no Apêndice B.

1. Repete-se o procedimento a partir de 2 (faz-se  $x_i = x_{i+1}$ ) até que:

$$|f(x_i) - y| < \textit{tolerância}$$

ou

$$i > \textit{máximo de iterações}$$

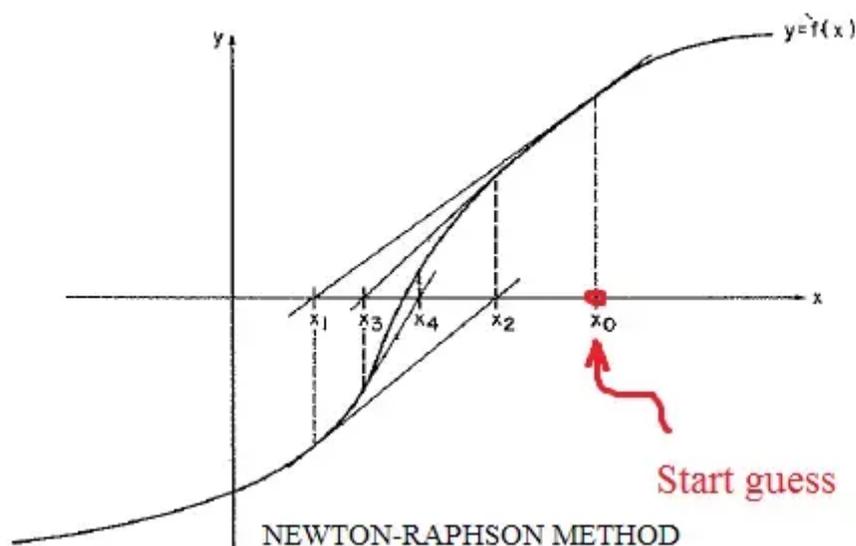


FIGURA 4.4 – Exemplo de convergência utilizando o método Newton-Raphson.

Existem, porém, alguns pontos fracos no método Newton-Raphson. Os principais são: pontos de inflexão nas proximidades de um ponto, derivada nula (ou determinante do Jacobiano, no nosso caso) e máximos (ou mínimos) locais. Abaixo, estão representadas funções hipotéticas que gerariam divergência no método Newton-Raphson devido aos pontos fracos mencionados.

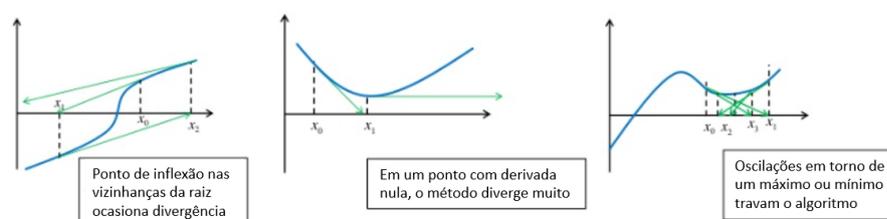


FIGURA 4.5 – Possíveis causas de divergência no método Newton-Raphson. (YILDIRIM, )

Se olharmos os diagramas tensão-deformação do aço e do concreto, veremos que ambos possuem patamares a partir de certas deformações. Para essas deformações, a derivada da tensão é nula, ou seja, uma variação infinitesimal da deformação não gera variação da tensão. Denominamos o conjunto de pontos nessa situação de patamar dos materiais (ou, simplificada e daqui em diante, patamar). Seções com grandes proporções de patamares costumam gerar divergência devido às derivadas nulas que geram Jacobianos Nulos ou quase singulares.

A desvantagem de dar “passos largos” é que, seções otimizadas estão propositalmente próximas do limite de ruína e, conseqüentemente, próximas das regiões de divergência. Um

passo largo poderia resultar em uma falsa divergência por recair em seções inteiramente comprometidas por regiões inúteis.

No caso do método do Gradiente, temos um método que visa maximizar ou minimizar objetivos dando “passos curtos” nos sentidos que maximizem ou minimizem esses objetivos de acordo com a inclinação da curva no ponto atual da iteração. Sendo assim, o método do gradiente não é propriamente um método para encontrar uma raiz, mas se definirmos a função objetivo a ser minimizada como a distância euclidiana entre a imagem encontrada e a imagem desejada; temos então um método útil ao nosso problema. Porém, como mencionado anteriormente, o método pode ser computacionalmente custoso. Para esclarecer o que foi dito, veja o seguinte pseudo-código do método do Gradiente Descendente:

1. Define-se um ponto inicial  $x_i$  ( $i=0$ ) no domínio.
2. Calcula-se a imagem  $f(x_i)$  e a direção de máximo aclave  $\nabla f(x_i)$ .

Onde:

$$f(x_i) = (N_d - N_r)^2 + (M_{xd} - M_{xr})^2 + (M_{yd} - M_{yr})^2$$

$$\nabla f(x_i) = \left( \frac{\partial f}{\partial \varepsilon}, \frac{\partial f}{\partial k_x}, \frac{\partial f}{\partial k_y} \right)$$

Calculando:

$$\frac{\partial f}{\partial \varepsilon} = 2 \left( \frac{\partial N_r}{\partial \varepsilon} (N_r - N_d) + \frac{\partial M_{xr}}{\partial \varepsilon} (M_{xr} - M_{xd}) + \frac{\partial M_{yr}}{\partial \varepsilon} (M_{xr} - M_{yd}) \right)$$

$$\frac{\partial f}{\partial k_x} = 2 \left( \frac{\partial N_r}{\partial k_x} (N_r - N_d) + \frac{\partial M_{xr}}{\partial k_x} (M_{xr} - M_{xd}) + \frac{\partial M_{yr}}{\partial k_x} (M_{xr} - M_{yd}) \right)$$

$$\frac{\partial f}{\partial k_y} = 2 \left( \frac{\partial N_r}{\partial k_y} (N_r - N_d) + \frac{\partial M_{xr}}{\partial k_y} (M_{xr} - M_{xd}) + \frac{\partial M_{yr}}{\partial k_y} (M_{xr} - M_{yd}) \right)$$

1. Calcula-se o próximo ponto do domínio:

$$x_{i+1} = x_i - t \nabla f(x_i)$$

Onde  $t$  é o “comprimento” do passo, que pode ser estático ou variável.

No presente artigo, utilizou-se passo variável por busca linear no intervalo unidimensional de um dado passo máximo.

1. Repete-se o procedimento a partir de 2 (faz-se  $x_i = x_{i+1}$ ) até que:

$$|f(x_i) - y_i| < \textit{tolerância}$$

ou

$$i > \textit{máximo de iterações}$$

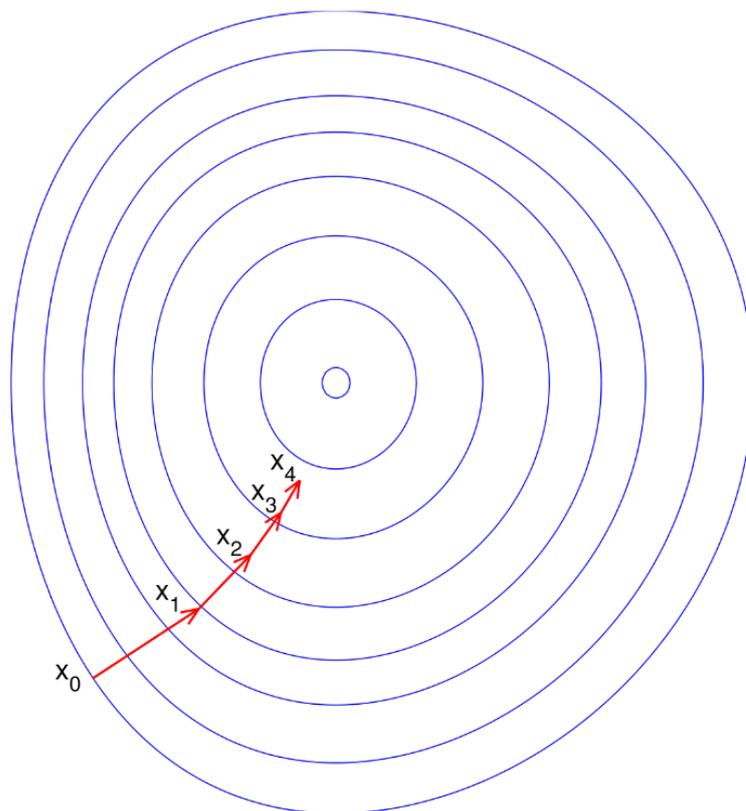


FIGURA 4.6 – Representação de uma iteração do método do Gradiente.

Assim como no método de Newton-Raphson, o método do gradiente descendente também possui os mesmos pontos fracos do Newton Raphson, exceto os relacionados aos passos largos (caso escolha-se bem o comprimento do passo) já comentados e adicionando o custo computacional. Porém, existem algumas formas de reduzir o custo computacional otimizando o tamanho dos passos e a direção do passo (evitando zigue-zagues) como por exemplo o método do gradiente conjugado. As iterações do método do gradiente conjugado seguem o seguinte algoritmo:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \dots,$$

Onde  $x_0$  é o ponto inicial,  $\alpha_k$  é o tamanho do passo e:

$$d_k = \begin{cases} -g_k & k = 0 \\ -g_k + \beta_k d_{k-1} & k \geq 1 \end{cases}$$

Onde  $g_k = \nabla f(x_k)$  e o  $\beta_k$  utilizado foi:

$$\beta_k = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \text{Fletcher – Reeves [10]}$$

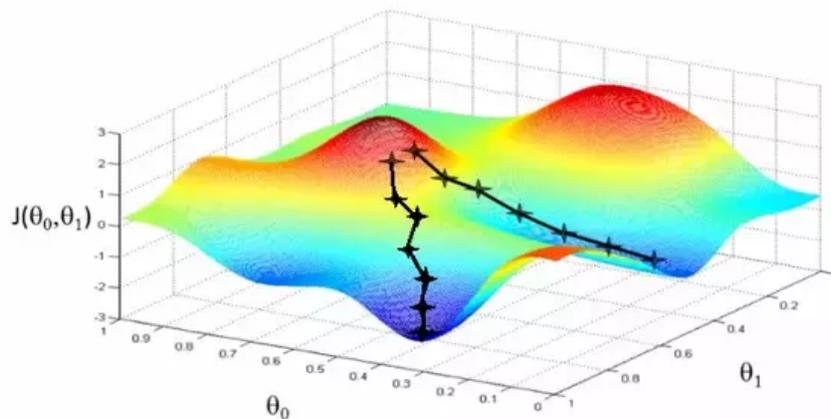


FIGURA 4.7 – Representação de duas iterações do método do gradiente descendente, iniciando-se de dois pontos diferentes. Uma delas cai em um mínimo local (NG, 2004)

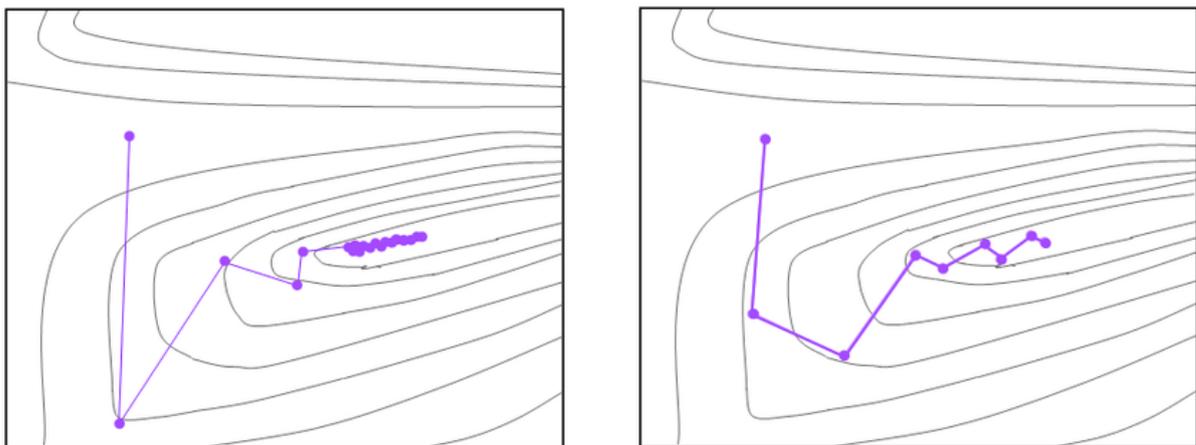


FIGURA 4.8 – Aplicação dos métodos do gradiente descendente tradicional (à esquerda) e do gradiente conjugado (à direita). Percebe-se uma clara economia de iterações (JUMBO, 2011).

Mostradas as vantagens e desvantagens associadas a cada método, levando-se em consideração o problema em questão, podemos agora aplicá-los e analisar os resultados. Sendo

---

ambos satisfatórios, escolher-se-á o melhor para prosseguir com a elaboração do modelo de otimização.

# 5 Comparação dos Resultados

## 1. Comparação dos Resultados

### 1. (a) Armaduras Pontuais vs Distribuição Linear Simplificada

Na presente seção, será feito um teste de sanidade para verificar e comparar os resultados de duas seções análogas. A primeira será composta por armaduras pontuais distribuídas uniformemente e a segunda será a seção equivalente simplificada como uma distribuição linear. É esperado que os resultados obtidos fiquem próximos, uma vez que a simplificação mantém a área de armadura da seção. Porém, uma vez que a distribuição desta área não é igual, é natural que ocorram pequenas divergências.

A seção utilizada no teste de sanidade foi uma seção quadrada (50x50cm) com 3 cm de cobrimento (aqui considerado como a distância entre centros das barras pontuais ou linha central da distribuição linear e a borda da seção) e distribuição linear de 1 cm de espessura e 44 cm (50 – 3 – 3) de comprimento. A distribuição pontual aproximada será feita com 4 barras em cada lado, com área igual a um quarto da área da distribuição linear, isto é 11 cm<sup>2</sup>.

$$\pi \left( \frac{\emptyset}{2} \right)^2 = 11 \text{ cm}^2 \rightarrow \emptyset \cong 3,742 \text{ cm}$$

O posicionamento das barras pontuais será o ponto médio de cada trecho de 11 cm. Sendo assim, em se tratando de uma deformação linear, o ponto médio estará submetido a uma deformação média. Desta forma, em se tratando de esforços normais, a distribuição linear é completamente equivalente à distribuição linear desde que o aço não escoe (patamar do diagrama tensão deformação). No entanto, como o ponto de aplicação de uma distribuição linear de forças não é necessariamente o ponto médio, é natural que ocorram diferenças nos momentos fletores. Quanto maior o número de barras, por exemplo se usássemos 32, e não 16, mais próximo seria o resultado.

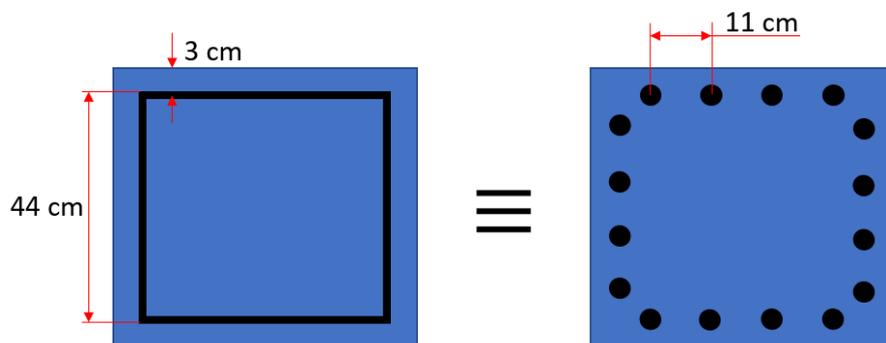


FIGURA 5.1 – Seções equivalentes linear (esquerda) e pontual (direita), que foram usadas no teste de sanidade.

Definidas as seções, a seção pontual foi reproduzida no software nFOCCA (MEDEIROS, 2004) e a distribuição linear foi reproduzida em um script no Matlab desenvolvido pelo autor do presente trabalho. Os resultados obtidos, para as duas abordagens, foram os seguintes:

Estado de Deformação			Esforços Resistentes								
e0	kx	ky	Distribuição Pontual			Distribuição Linear			Diferença Relativa Absoluta		
			Nr (kN)	Mxr (kN)	Myr (kN)	Nr (kN)	Mxr (kN)	Myr (kN)	Nr (kN)	Mxr (kN)	Myr (kN)
0,5	0,00	0,00	4,504	0,000	0,000	4,504	0,000	0,000	0,00%	0,00%	0,00%
0,5	0,01	0,00	4,473	21,226	0,000	4,473	21,412	0,000	0,00%	0,88%	0,00%
0,5	0,01	0,01	4,441	21,226	21,226	4,441	21,412	21,412	0,00%	0,88%	0,88%
0,5	0,03	0,01	4,367	59,936	18,959	4,367	60,495	19,146	0,00%	0,93%	0,99%
1	0,04	0,01	7,729	71,885	17,767	7,728	72,605	17,929	0,01%	1,00%	0,91%
1	0,08	0,03	6,384	109,398	35,739	6,361	110,476	35,535	0,36%	0,99%	0,57%
1	0,14	0,04	1,376	108,379	26,044	1,393	109,132	25,750	1,24%	0,69%	1,13%

TABELA 5.1 – Resultados do processo de cálculo dos esforços resistentes, comparando resultados para as distribuições pontual e linear.

Esforços Solicitantes			Estado de Deformação								
Nr (kN)	Mxr (kN)	Myr (kN)	Distribuição Pontual			Distribuição Linear			Diferença Relativa Absoluta		
			e0	kx	ky	e0	kx	ky	e0	kx	ky
1000	10000	10000	0,1013	0,0045	0,0045	0,1015	0,0045	0,0045	0,20%	0,00%	0,00%
1000	20000	10000	0,0875	0,0098	0,0051	0,0880	0,0097	0,0050	0,57%	1,02%	1,96%
2000	30000	10000	0,2068	0,014	0,0049	0,2072	0,0139	0,0048	0,19%	0,71%	2,04%
5000	50000	20000	0,5838	0,0246	0,0100	0,5836	0,0243	0,0099	0,03%	1,22%	1,00%

TABELA 5.2 – Resultados do processo de verificação da seção, comparando resultados para as distribuições pontual e linear.

Perceba que, apesar da mudança no conceito da distribuição, uma distribuição linear pode aproximar muito bem uma distribuição pontual, sem precisar entretanto definir o posicionamento e a bitola de cada barra, facilitando a otimização que será feita posteriormente.

### 1. (a) **Newton-Raphson vs. Gradiente Descendente Conjugado**

Como mencionado na seção anterior, o problema de verificação das seções foi abordado com dois métodos distintos, de modo a compará-los e escolher o mais adequado:

1. Newton-Raphson
2. Gradiente Descendente Conjugado

Na primeira abordagem, observou-se uma conclusão da verificação computacional (convergente ou não) muito rápida. Isso se deu, como explicado, devido aos passos largos dados pelo método Newton-Raphson. Diferentemente do que se esperava, seções próximas do ELU não sofreram falsos negativos devido a iterações que passavam direto para o patamar. O que ocorreu devido ao formato dos diagramas tensão-deformação dos materiais.

Perceba que, tanto para o aço quanto para o concreto, as maiores derivadas se encontram no ponto de deformação nula (ponto de início das iterações do Newton-Raphson. Sendo assim, a medida que a seção sofre deformações, as derivadas dos esforços em relação às deformações também diminuem. Temos então uma função crescente cuja derivada é decrescente. Observe as seguintes ilustrações do método Newton-Raphson:

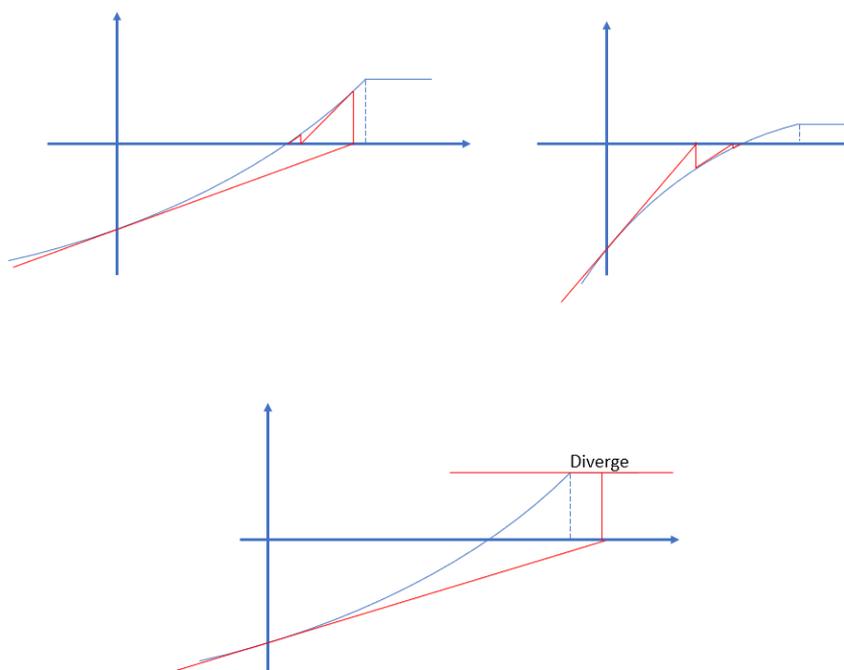
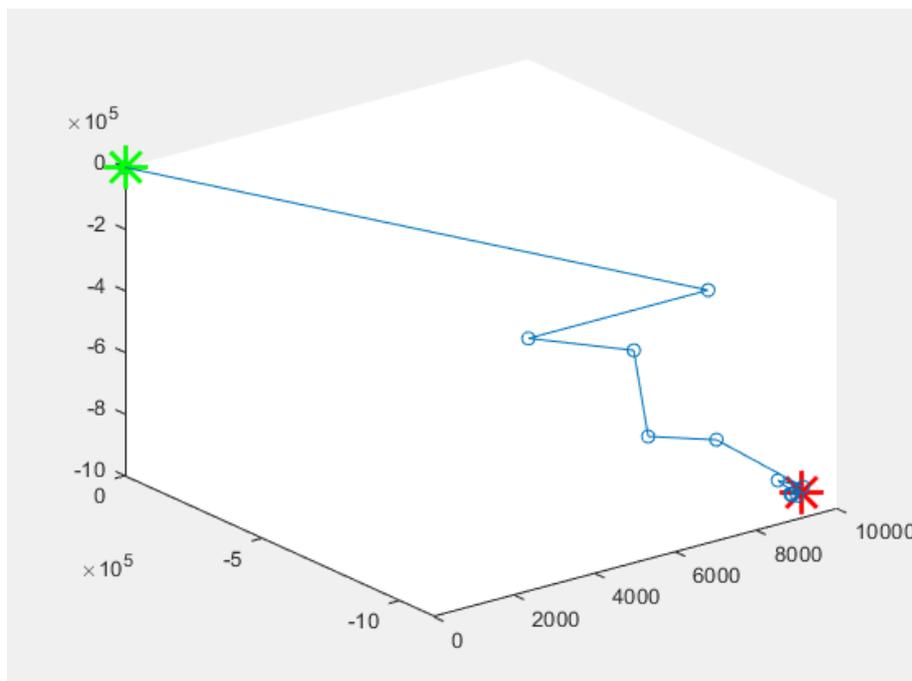


FIGURA 5.2 – Newton-Raphson aplicado a funções com segunda derivada positiva (em cima) e negativa (embaixo).

Perceba na figura acima que nos casos em que a função possui derivada crescente, ou seja, segunda derivada positiva, uma iteração do Newton-Raphson vai além da raiz para

só então voltar. Essa situação poderia recair em um patamar. No entanto, em situações de derivada decrescente, como é o nosso caso, as iterações do Newton-Raphson não vão além da raiz, mas se aproximam gradativamente dela. Sendo assim, o risco comentado anteriormente de se atingir um patamar foi descartado e as iterações foram pouco custosas computacionalmente.

Já para o método do Gradiente Descendente Conjugado, ainda que o tamanho do passo em cada iteração variasse utilizando uma busca linear, o método precisava de muitas iterações para chegar próximo à raiz e, mesmo que chegasse, travava antes de atingir a tolerância mínima. Ainda que utilizássemos uma tolerância exageradamente alta, o método do gradiente precisou de mais iterações (iterações mais custosas, também) que o método Newton-Raphson. Veja o emprego dos métodos no espaço tridimensional de uma seção genérica:



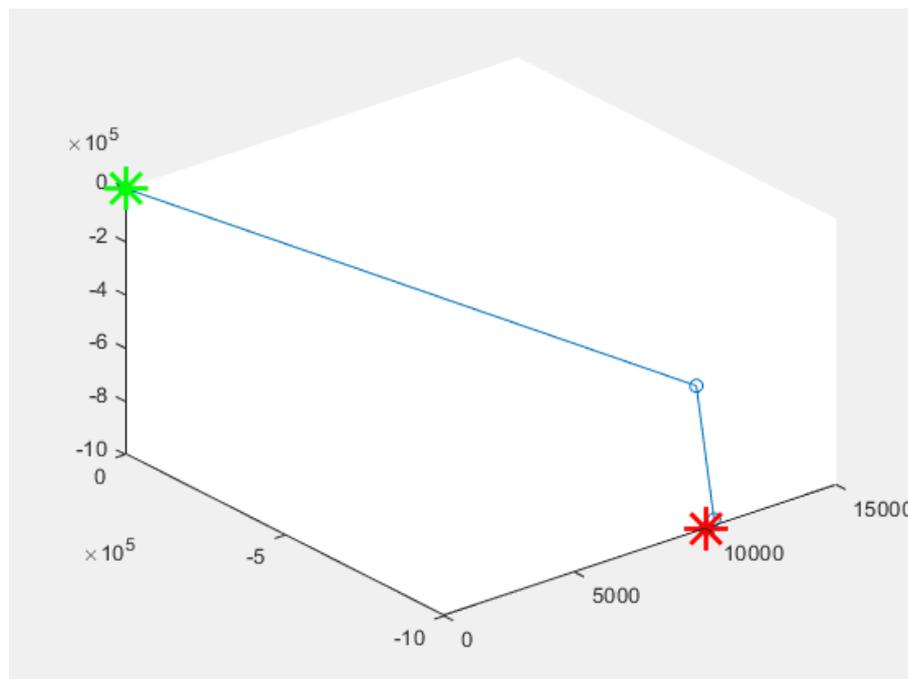


FIGURA 5.3 – Emprego dos métodos do Gradiente Descendente Conjugado (superior) e Newton-Raphson (inferior). Esforços resistentes inicial em verde e objetivo em vermelho.

No exemplo ilustrado nas figuras acima, o método Newton-Raphson convergiu em 3 iterações enquanto o método do gradiente conjugado precisou de milhares de iterações e, mesmo assim, não convergiu. Sendo assim, o método de verificação de seções que será utilizado é o método Newton-Raphson.

# 6 Fronteiras do ELU para diferentes seções

## 1. Fronteiras do ELU para Diferentes Seções

Como mostrado anteriormente, o ELU deve obedecer as restrições expostas nas equações X, Y e Z. Na figura XX, foi definida uma direção da linha neutra, mas esta poderia estar em qualquer direção em um caso de flexão oblíqua. Para analisar a fronteira de segurança do ELU em uma situação de flexão oblíqua, definiremos essa direção com diferentes ângulos. Denominaremos como linha de iso-deformação.

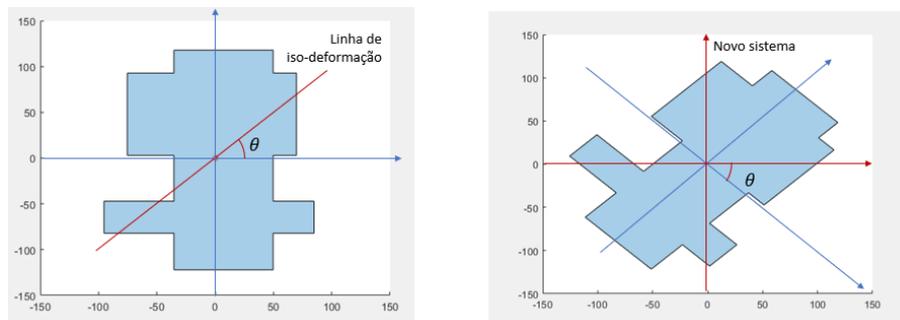


FIGURA 6.1 – Linha de iso-deformação de um estado de deformação, passando pela origem (C.G). À esquerda, sistema rotacionado.

Ao adotarmos um novo sistema de coordenadas no qual o eixo x passe pela linha de iso-deformação do ponto central, podemos analisar o problema de deformação oblíqua (duas componentes,  $k_x$  e  $k_y$ ) como um sistema de deformação simples (uma componente  $k$ ), em torno do novo eixo x. Devemos tomar cuidado, porém, que apesar de a deformação estar restrita a um único eixo, o momento fletor não necessariamente possuirá a segunda componente nula, pois a seção não é necessariamente simétrica. Sendo assim, uma deformação em torno do eixo x pode resultar em esforços resistentes com componentes do momento fletor em torno do eixo y.

Nesse novo sistema de coordenadas, pode-se demonstrar facilmente que:

$$k = \frac{k_x}{\cos\theta} = \frac{k_y}{\sin\theta}$$

Definida uma direção de iso-deformação, percorreremos as combinações  $(\varepsilon_0, k)$  que estejam na fronteira do ELU, segundo os passos mostrados na figura 4.1 (polos de ruína). Para a seção genérica da figura anterior, por exemplo, teremos para  $\theta = 45^\circ$ :

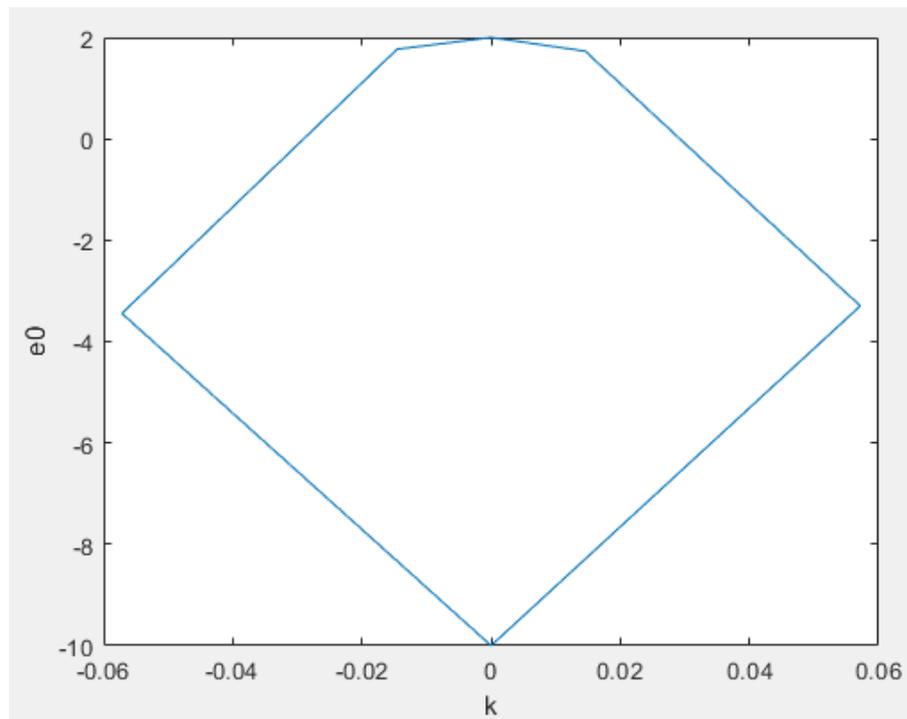


FIGURA 6.2 – Fronteira de deformação da seção genérica para  $\theta = 45^\circ$ .

Utilizando os pontos dessa fronteira na função de esforços resistentes desenvolvida anteriormente, teremos as respectivas triplas ordenadas correspondentes aos esforços resistentes. Unindo esses pontos, temos o gráfico mostrado abaixo:

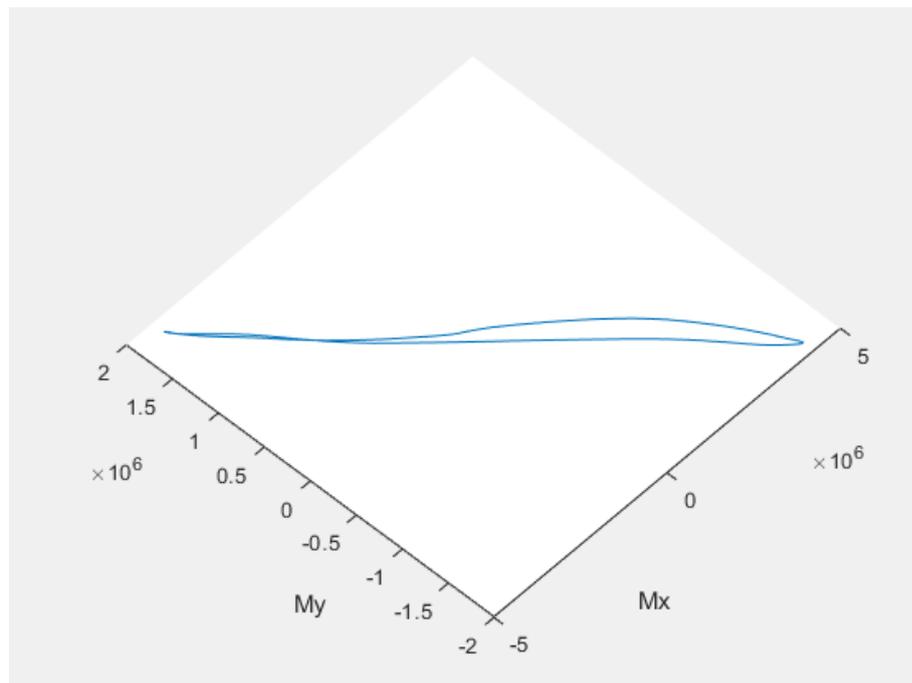
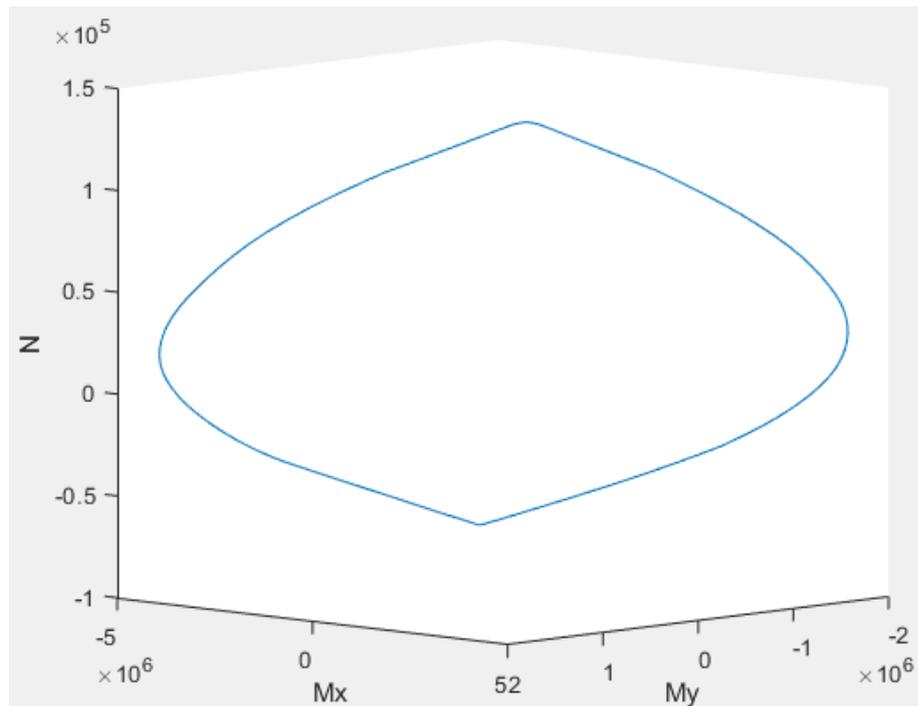


FIGURA 6.3 – Esforços resistentes referentes à fronteira de deformação da seção genérica para  $\theta = 45^\circ$ .

Repetindo o procedimento para  $\theta$  variando de  $-90^\circ$  a  $90^\circ$ , de 3 em 3 graus e plotando os resultados em um mesmo gráfico, temos as seguintes superfícies:

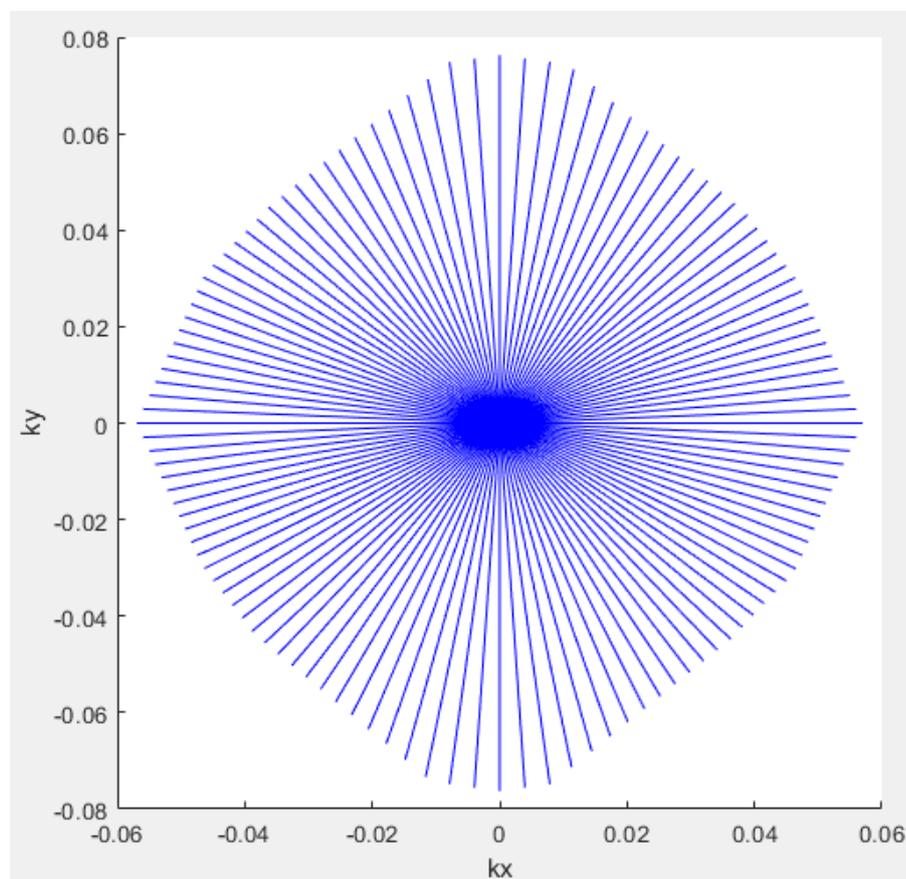
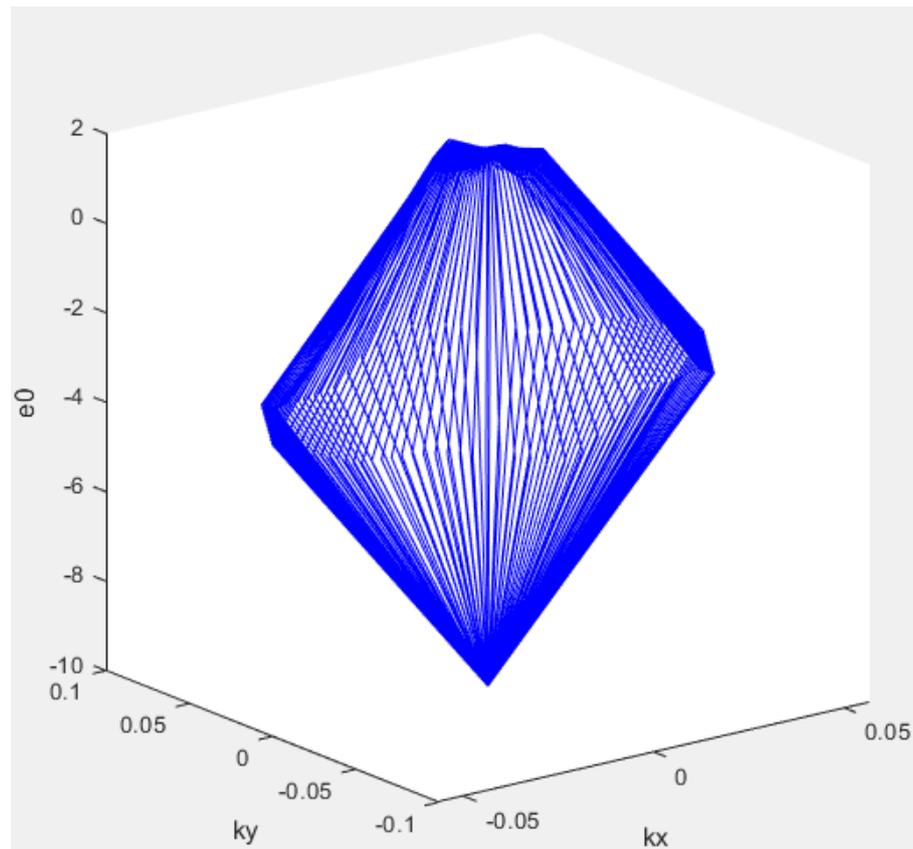


FIGURA 6.4 – Fronteira de deformação da seção genérica.

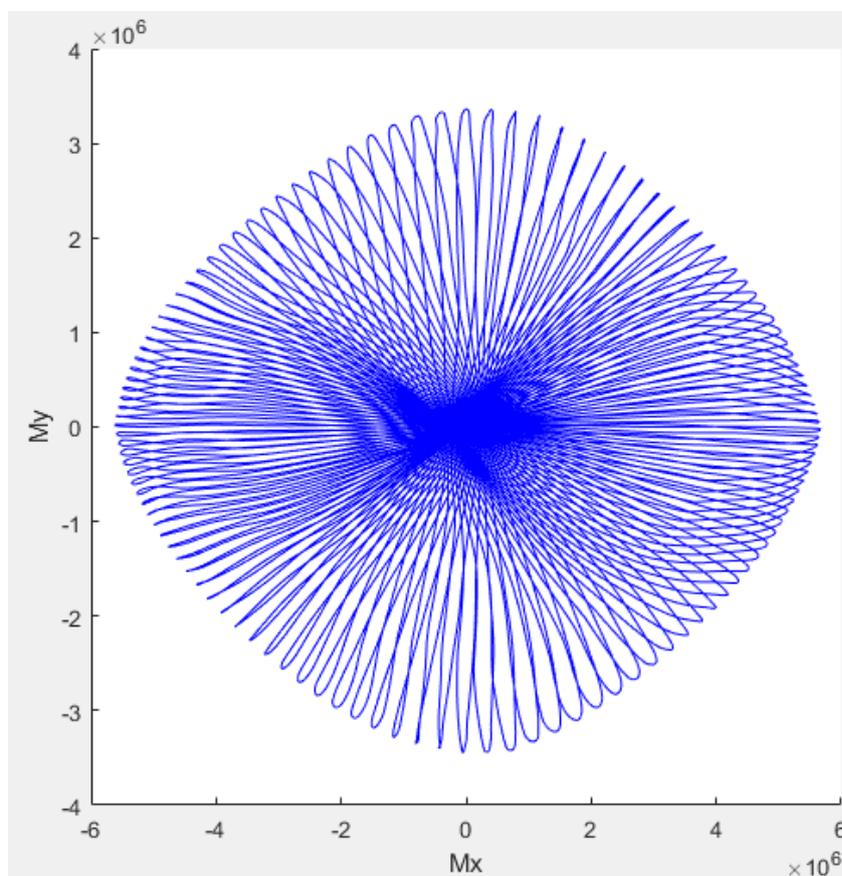
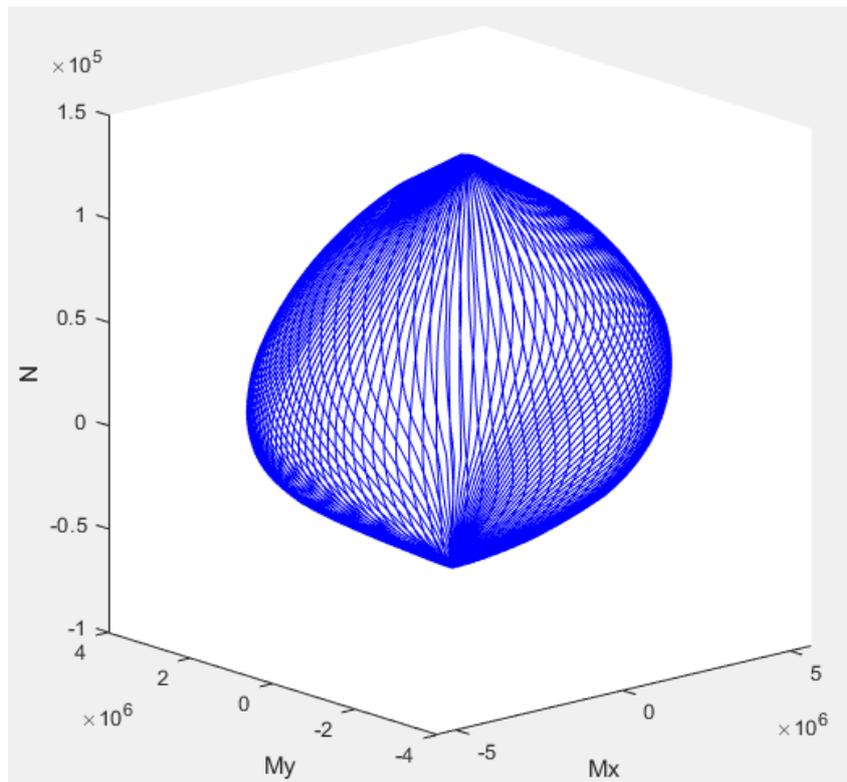
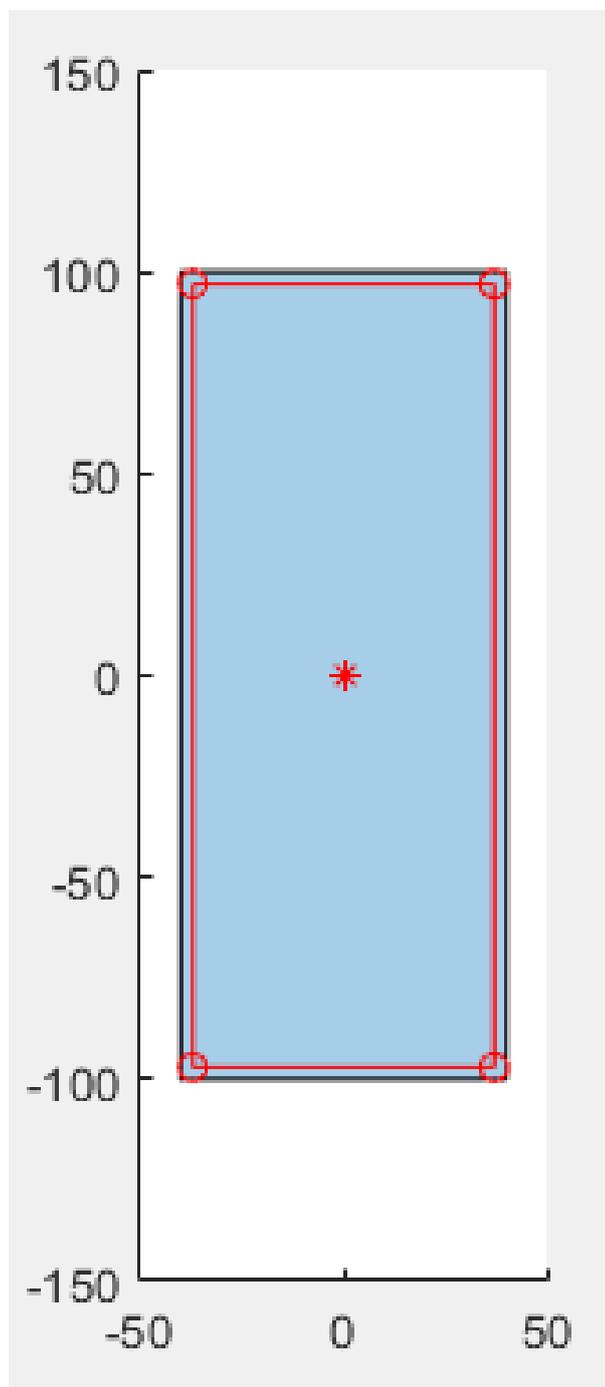


FIGURA 6.5 – Esforços resistentes referentes à fronteira de deformação da seção genérica.

Para as outras seções, os formatos das superfícies do ELU foram bastante similares. As diferenças maiores foram percebidas para seções com distribuição pouco “circular”, isto é, com inércias muito diferentes entre os eixos  $x$  e  $y$ . Por exemplo:



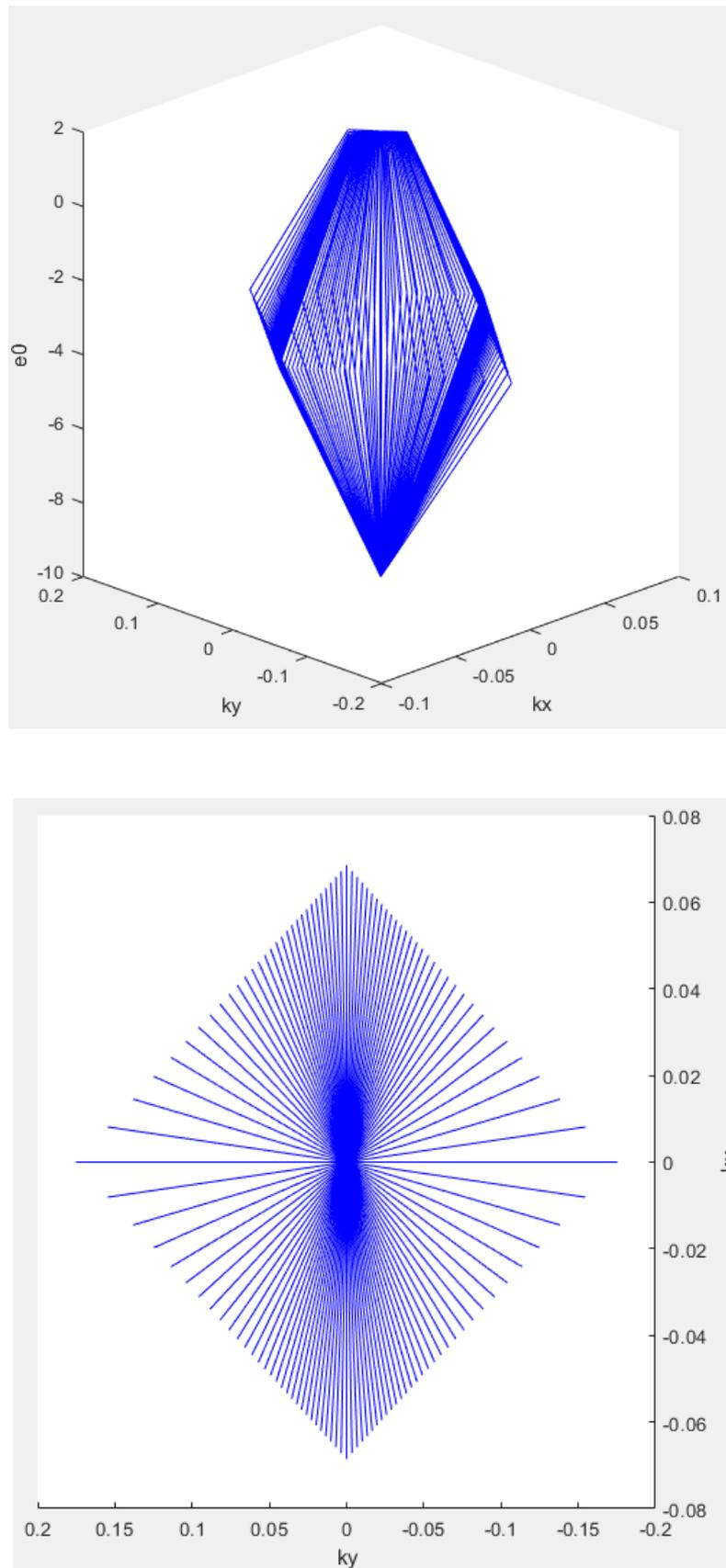


FIGURA 6.6 – Seção retangular e sua fronteira de deformação.

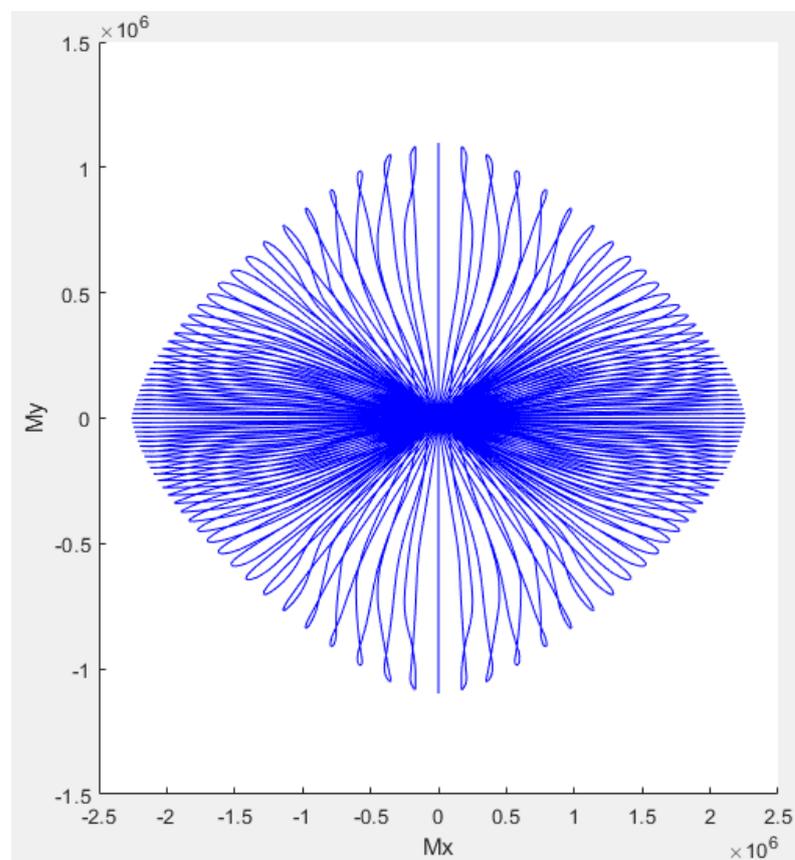
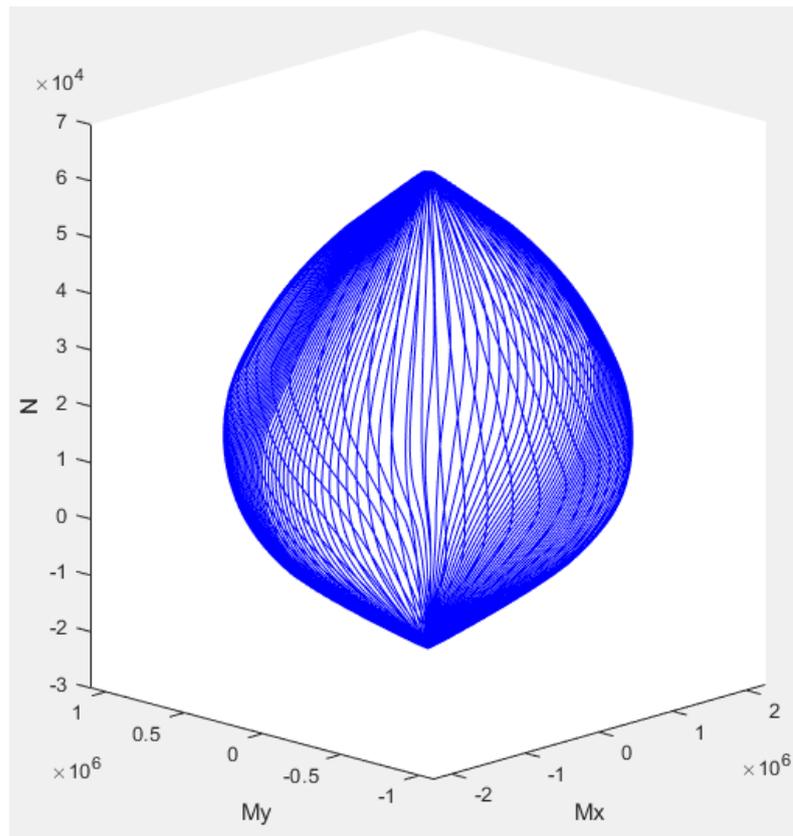
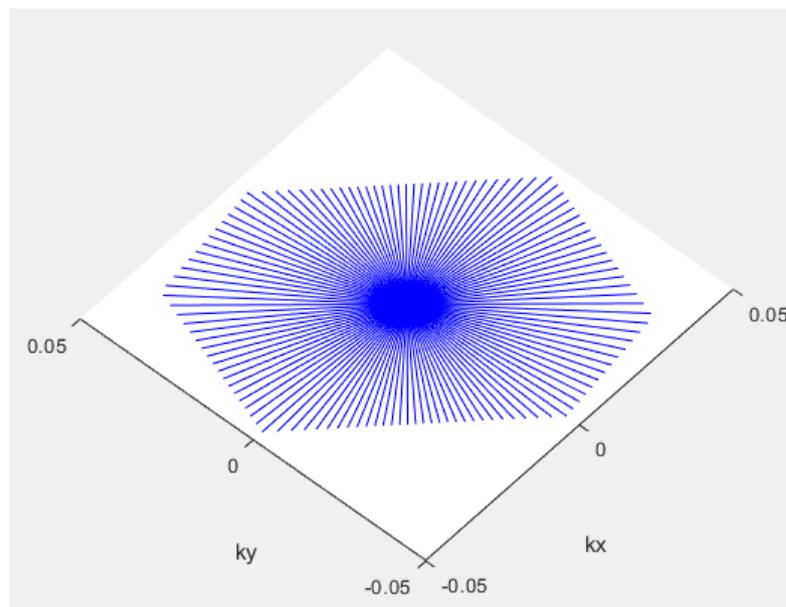
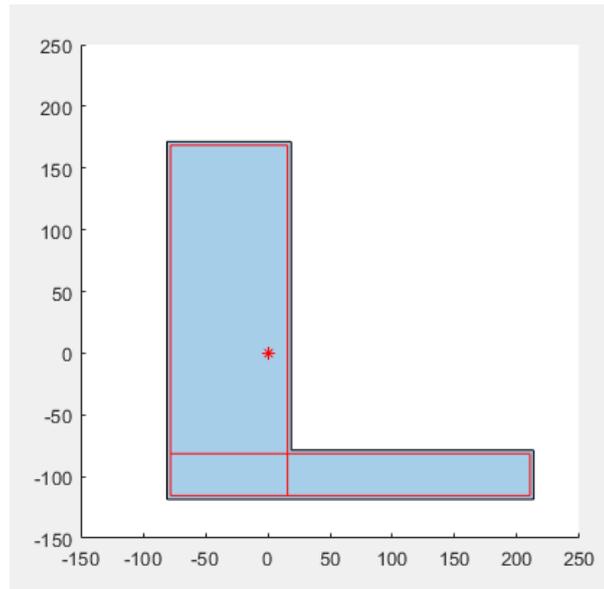


FIGURA 6.7 – Esforços resistentes referentes à fronteira de deformação da seção retangular.



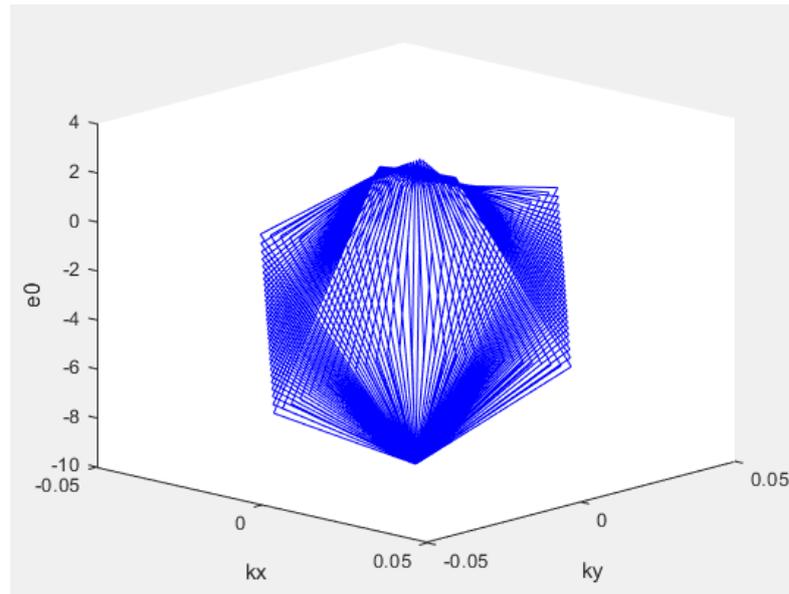
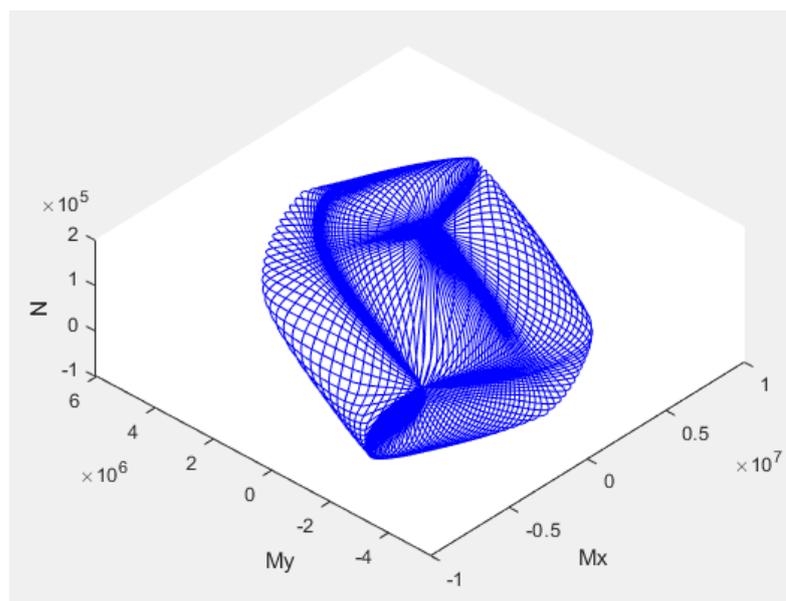


FIGURA 6.8 – Seção L e sua fronteira de deformação.



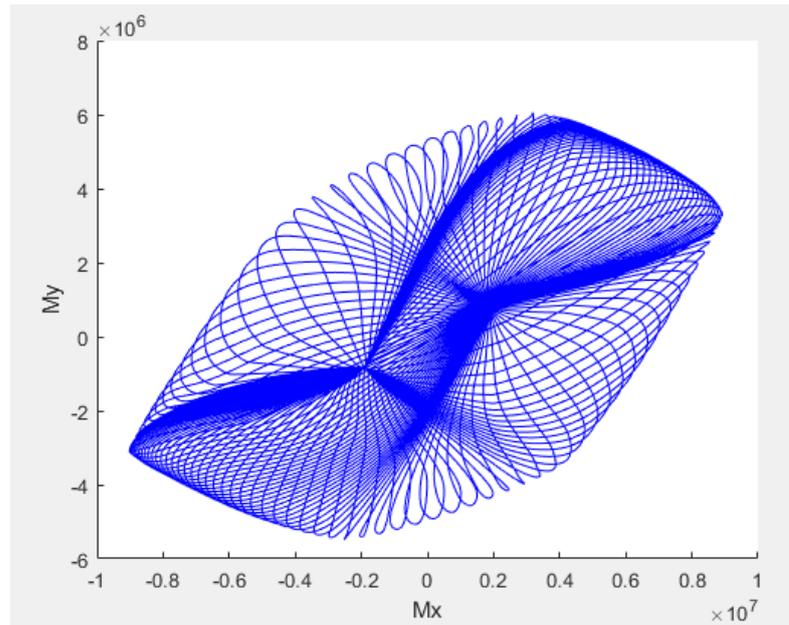


FIGURA 6.9 – Esforços resistentes referentes à fronteira de deformação da seção L.

# 7 Otimização de Seções com o Método Simulated Annealing

## 7.1 Implementação do Simulated Annealing

Até o momento, o presente estudo tem utilizado seções genéricas que podem recair em diversos tipos diferentes de seção. No entanto, a prática mostra, em geral, uma ampla utilização das seções retangulares mesmo em situações nas quais seria mais eficiente utilizar outros tipos de seção, como seções em L ou em U. Isso se dá pela facilidade de cálculo e de otimização pelos métodos tradicionais, uma vez que são poucos parâmetros a serem modificados. Com a utilização de softwares estruturais que calculam a área de aço necessária, é necessário apenas definir a base e altura da seção retangular.

Já a ponderação entre diferentes seções é, naturalmente, mais trabalhosa. No entanto, a parametrização definida na primeira seção aliada ao modelamento para verificação de seções genéricas permite a criação de uma ferramenta de otimização de seções de quaisquer formatos, sem supor um tipo de seção inicialmente. Essa ferramenta tem a intenção de encontrar a seção otimizada (menor custo) frente a uma dada solicitação por força normal e flexão oblíqua.

Os 11 parâmetros definidos anteriormente (figura 2.1) serão os inputs que definem uma potencial seção vencedora, podendo ou não recair no caso da seção retangular. Uma vez definidos parâmetros, estará definido o formato da seção de concreto. Porém, restam ainda alguns parâmetros para definir a seção completamente. O primeiro diz respeito ao aço, qual a área de aço utilizar na seção, como distribuí-la? Seguindo a simplificação adotada anteriormente, utilizaremos uma distribuição uniforme modelada como uma distribuição linear. Como são 3 retângulos que definem a seção genérica, serão 3 parâmetros representando a espessura da distribuição linear. Além disso, devem ser respeitados os limites discutidos mínimos e máximos calculados anteriormente.

Com esses 14 parâmetros, a seção modelada está completamente definida, mas não o seu posicionamento com relação ao ponto de aplicação dos esforços. A seção em questão pode ser rotacionada em torno do CG (ponto de aplicação dos esforços solicitantes, por

convenção) com relação a uma posição de referência. Sendo assim, é adicionado 1 parâmetro referente ao ângulo  $\theta$  de orientação da seção. Resultando em 15 parâmetros no total.

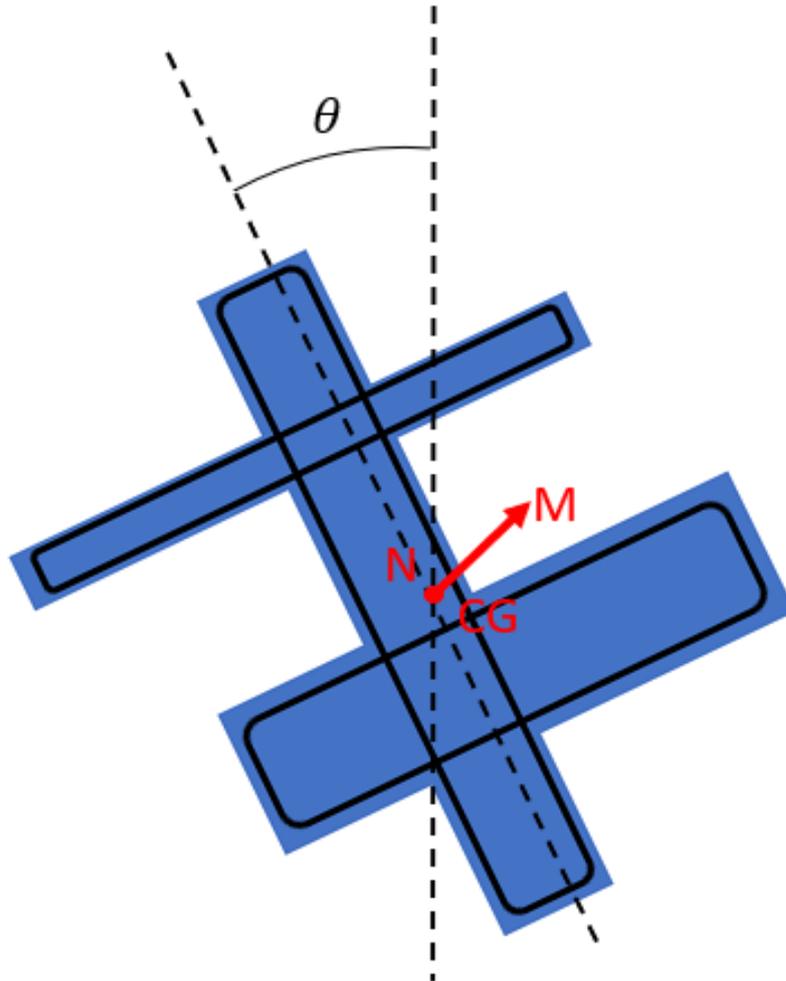


FIGURA 7.1 – Translação e rotação da seção com relação ao CG.

### 7.1.1 Crossover

O método AUSA possui forte fundamentação no processo evolutivo biológico e possui muitos fatores em comum por tratar-se de um algoritmo genético. Assim como na evolução das espécies, os indivíduos de uma população geram novos indivíduos ou se modificam através de reprodução (alvo da presente seção) e mutação, respectivamente. Desta forma, os novos indivíduos são selecionados pelo meio, que elimina os menos aptos e polpa os mais adaptados, conservando seus genes.

Durante a reprodução, o material genético de um indivíduo é combinado com o de outro indivíduo através de um processo denominado crossover, no qual parte dos genes

presentes em um indivíduo reprodutor são passadas ao novo indivíduo e os outros genes são herdados pelo segundo indivíduo.

No algoritmo desenvolvido, os indivíduos são selecionados a cada geração para reprodução com base em um fator de mérito que será explicado na seção posterior. Uma vez selecionados, os cromossomos de cada indivíduo são combinados de modo que cada dígito da string possui 50% de chance de vir do primeiro ou do segundo indivíduo. A figura 7.5 abaixo ilustra o processo de crossover dos cromossomos binários.

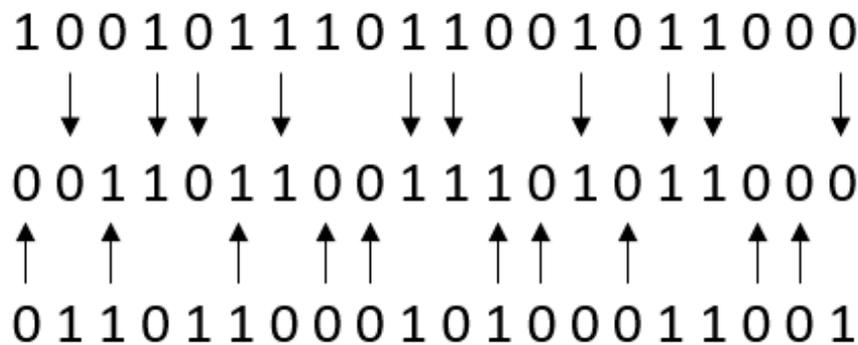


FIGURA 7.2 – Crossover de dois cromossomos binários gerando um terceiro.

O indivíduo gerado dessa forma pode não resistir aos esforços solicitantes, isto é, ultrapassar algum estado limite último, sendo descartado e, portanto, não modificando a população pre-existente. Caso o "filho"gerado seja mais barato que seus "pais", ele toma o lugar do melhor entre eles. Caso contrário, o método AUSA prevê ainda a possibilidade (menor que 100%) de que o filho substitua um dos pais). Essa característica do método possibilita que a otimização avance sobre regiões menos otimizadas rumando a ótimos globais. Conforme as épocas de iteração passam, o fator temperatura do AUSA diminui, diminuindo também as chances de que um indivíduo mais caro substitua um pai mais barato.

### 7.1.2 Fator de Mérito

Para selecionar os indivíduos utilizados nos processos de crossover ou de mutação, utilizou-se um fator de mérito igual ao inverso do custo somado a 1 (número inteiro pequeno para que o denominador não se anule).

$$F = \frac{1}{1 + c}$$

Quanto maior o fator de mérito de um indivíduo, isto é, quanto menor o custo, maiores as chances de que o indivíduo seja escolhido para o processo. Para isso, basta normalizar

os fatores de mérito de modo que a soma seja 1. Utilizando a mesma ordenação do vetor relativo à população atual, sorteia-se um número de 0 a 1 e escolhe-se o primeiro indivíduo cujo fator de mérito acumulado seja maior que esse número.

Além disso, o fator de mérito também foi utilizado para escolher os indivíduos que iriam sofrer mutação, e para determinar se um indivíduo mais caro substituirá um pai mais barato. A probabilidade de substituição é dada pela expressão abaixo:

$$p = e^{\frac{s_p - s_f}{T}}$$

Onde:

$$s_p = \frac{F_p}{F_p + F_f}$$
$$s_f = \frac{F_f}{F_p + F_f}$$

Desta forma, caso o fator de mérito do filho seja maior que o do pai, a probabilidade de substituição é maior que 1 e a troca ocorrerá. Caso contrário, a probabilidade será menor que 1, mas ainda pode acontecer. Para valores muito baixos de temperatura, o expoente negativo fica ainda mais negativo, diminuindo as chances de que ocorra a troca.

### 7.1.3 Mutação

A ocorrência de mutações nos cromossomos é dada pela inversão dos genes binários de um indivíduo com uma dada probabilidade. Utilizou-se uma probabilidade de 20% de mutação no processo de otimização das seções. Além disso, a probabilidade de ocorrer uma mutação foi 10 vezes mais alta que a probabilidade de ocorrência de crossover.

A mutação é interessante, por exemplo, em um cenário no qual a otimização fique presa a um mínimo local. Indivíduos iguais reproduzindo gerariam um mesmo indivíduo. Sendo assim a mutação insere uma variabilidade aleatória que pode tirar a população de um mínimo local.

### 7.1.4 Penalidades

Alguns fatores podem ser interessantes de inserir como custos adicionais, ainda que não causem efetivamente aumento no orçamento, para penalizar determinados indivíduos.

Por exemplo, ao aplicar uma carga normal pura em uma seção unicamente de concreto (por simplificação) o concreto se deformará uniformemente e a área da seção é a única

importante para saber se a esta obedecerá o ELU. Sendo assim, quaisquer retângulos ou quadrados com áreas maior que uma área mínima serão satisfatórios. No entanto, são evitadas seções muito finas (coeficiente base maior sobre base menor muito alto) e, sem outras restrições arquitetônicas, uma seção quadrada é preferencial dado que minimiza o perímetro e, portanto o gasto com formas. Pensando nisso, foi conveniente utilizar uma função custo que penaliza de acordo com esse fator:

$$p = 1 + e^{i-11} - e^{-10}$$

Onde  $i$  é igual à base maior sobre a base menor de um dado retângulo.

Além disso, foi utilizada uma penalidade para aplicar uma dimensão mínima aos lados dos retângulos. Caso a dimensão fosse menor que 10 centímetros, soma-se ao custo o maior decimal possível, de modo a invalidar o indivíduo com um custo muito alto.

### 7.1.5 Função Custo

A função custo é a função que expressa o custo por comprimento de um dado elemento, para uma dada seção. Para isso, foi realizada uma pesquisa de mercado de modo a obter os custos médios de concreto, aço e formas.

Para um concreto de 40 MPA e aço CA-50 os preços médios obtidos foram 0,00025R\$/cm<sup>3</sup> e 0,0692R\$/cm<sup>3</sup>, respectivamente. Já para as formas, o preço médio é de 0,0012R\$/cm<sup>2</sup>. Portanto, o custo por centímetro construído de uma determinada seção é dado por:

$$c = 0,00025 \cdot A_c + 0,0692 \cdot A_a + 0,0012 \cdot P_c \text{ R\$/cm}$$

Onde  $A_c$  é a área de concreto,  $A_a$  é a área de aço e  $P_c$  é o perímetro do concreto.

## 7.2 Resultados e Conclusões

De modo a se validar o modelo de otimização, alguns esforços solicitantes foram utilizados como input para o modelo de otimização. As seções ótimas resultantes serão mostradas a seguir.

Inicialmente, utilizou-se o método de otimização particular para seções retangulares, uma vez que o número de dígitos dos cromossomos é menor e os resultados esperados são mais intuitivos. O primeiro carregamento foi:

$$(N, M_x, M_y) = (10000 \text{ kN}, 100000 \text{ kN.cm}, 0 \text{ kN.cm})$$

A seção resultante foi a seguinte:

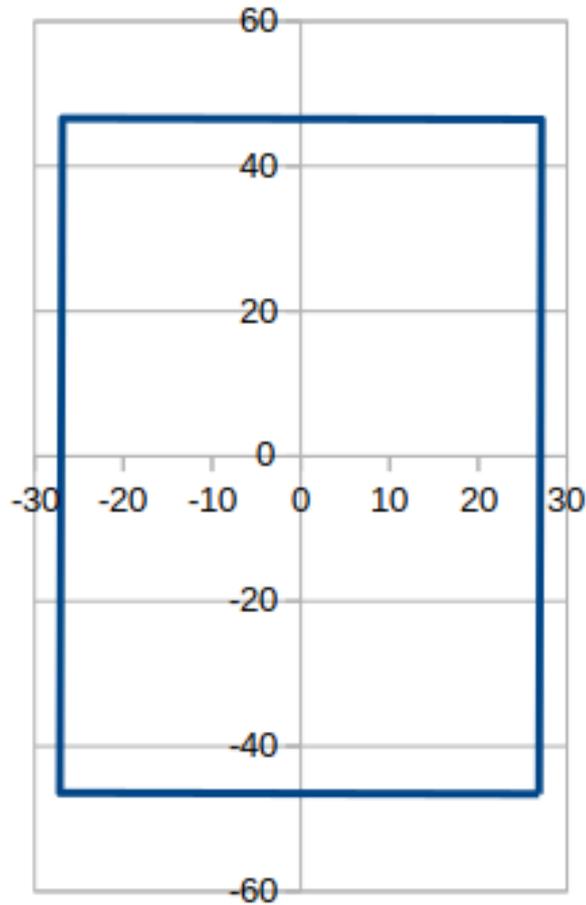


FIGURA 7.3 – Seção transversal resultante.

Cujos parâmetros são:

$$b = 93 \text{ cm}$$

$$h = 54 \text{ cm}$$

$$\theta = 89,82^\circ$$

$$d = 3,5 \text{ cm}$$

$$\lambda_1 = 0,0779 \text{ cm}$$

$$c = 3,04 \text{ R\$/cm}$$

Como esperado, a maior dimensão é a perpendicular ao eixo de aplicação do momento. Além disso, foi constatado que, sem contar o custo da forma ou a penalidade referente

ao coeficiente  $i$ , a seção ótima se torna uma linha muito fina, uma vez que isso aumenta o momento de inércia da seção relativo ao eixo  $x$ . No entanto, ao considerar o custo da forma, quanto mais próxima de um quadrado menor o perímetro e menor o custo com formas. Resultando na seção mostrada, com coeficiente  $i$  maior que 1 (seção quadrada) devido à aplicação de momento, mas sem se tornar uma seção muito fina.

O segundo carregamento foi:

$$(N, M_x, M_y) = (10000 \text{ kN}, 100000 \text{ kN.cm}, 100000 \text{ kN.cm})$$

A seção resultante foi a seguinte:

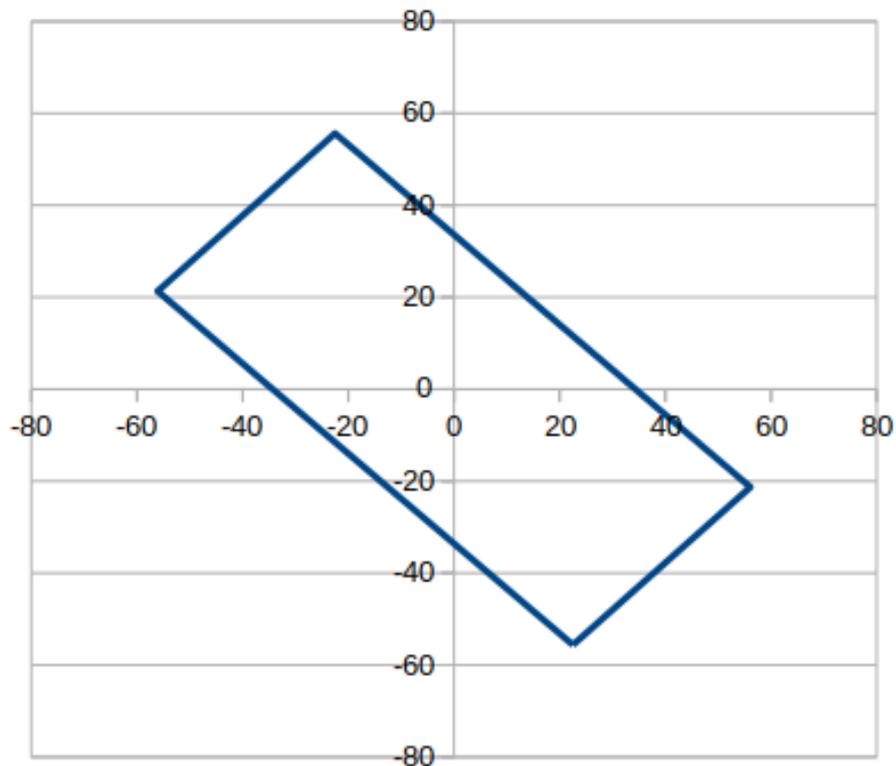


FIGURA 7.4 – Seção transversal resultante.

Cujos parâmetros são:

$$b = 48 \text{ cm}$$

$$h = 110 \text{ cm}$$

$$\theta = 45,62^\circ$$

$$d = 3,5 \text{ cm}$$

$$\lambda_1 = 0,0779 \text{ cm}$$

$$c = 3,38 \text{ R\$/cm}$$

Assim como no caso anterior, era esperado que a maior dimensão estivesse orientada perpendicularmente ao momento resultante e foi o que ocorreu na seção otimizada resultante. A seção agora possui um coeficiente  $i$  maior ( $110/48 > 93/54$ ) devido a um momento resultante maior.

Para o mesmo carregamento utilizado anteriormente, realizou-se a otimização genérica com apenas dois "braços". O resultado dessa otimização foi bastante semelhante à otimização anterior, resultando em um custo praticamente igual e certamente, com um número maior de épocas de evolução, as soluções dos dois métodos iriam convergir. A semelhança com a seção anterior pode ser percebida na seção esquematizada na figura abaixo:

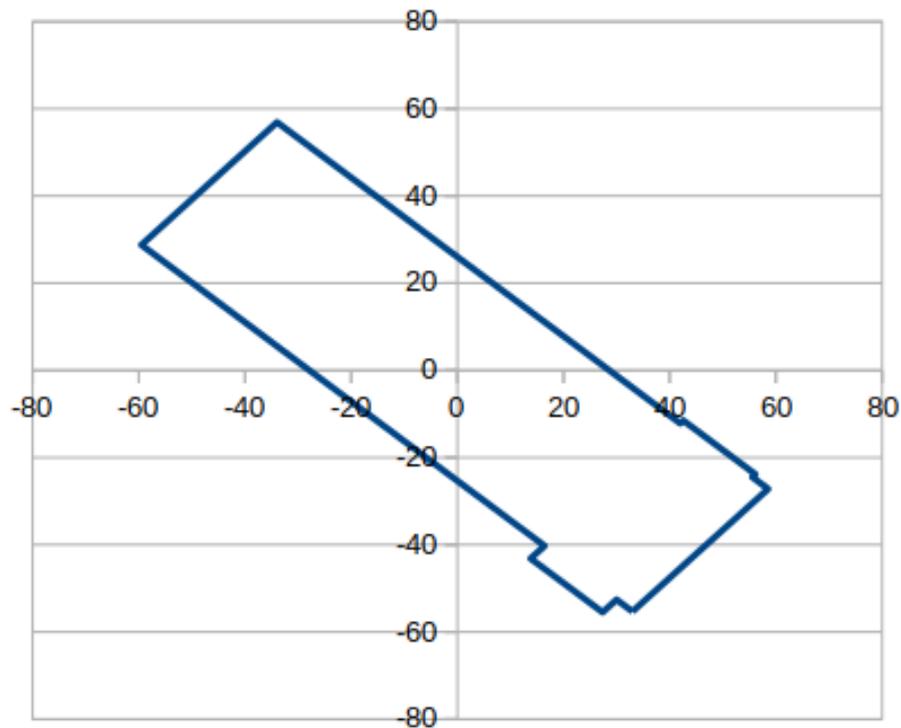


FIGURA 7.5 – Seção transversal resultante.

Cujos parâmetros são:

$$b = 38 \text{ cm}$$

$$h = 125 \text{ cm}$$

$$\theta = 47,73^\circ$$

$$d = 3 \text{ cm}$$

$$\lambda_1 = 0,0782 \text{ cm}$$

$$\alpha_1 = 0,0322581$$

$$\alpha_2 = 0.151515$$

$$\beta_1 = 0,0312805$$

$$\beta_2 = 0,00684262$$

$$c = 3,38 \text{ R\$/cm}$$

O algoritmo, no entanto, demorou significativamente mais que no caso anterior, inviabilizando uma utilização rotineira do algoritmo de otimização. No entanto, estudos futuros poderiam utilizar o presente método para mapear seções ótimas e registrá-las em uma base de dados com auxílio de hardwares mais robustos para que o resultado possa ser consultado em complexidade de tempo  $O(1)$ , uma vez criada a base de dados.

Por fim, utilizou-se o mesmo carregamento anterior para a seção genérica completa, isto é, com 2 "braços". Assim como o caso anterior, a otimização demorou bastante, de forma inviável de se reproduzir rotineiramente. O resultado foi uma seção bem diferente da tradicional e significativamente mais barata, como pode ser visto abaixo:

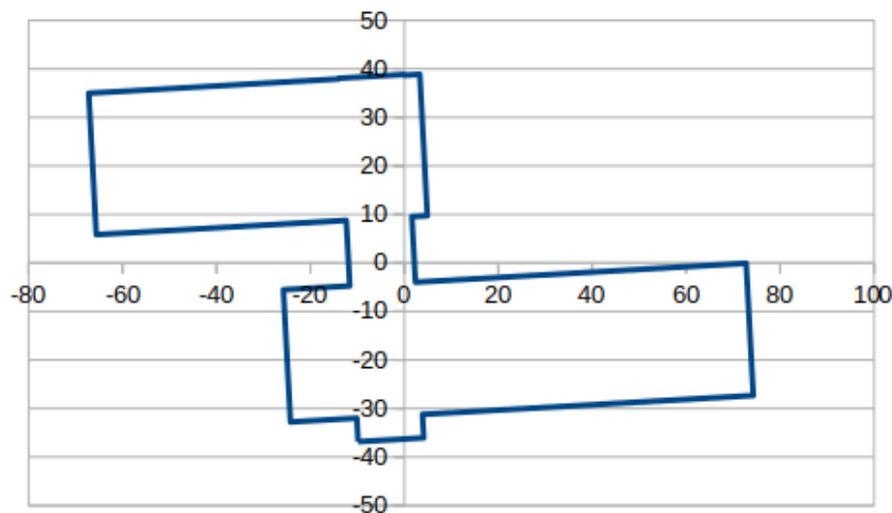


FIGURA 7.6 – Seção transversal resultante.

Cujos parâmetros são:

$$b = 14 \text{ cm}$$

$$h = 75 \text{ cm}$$

$$\theta = 3,17^\circ$$

$$d = 3,5 \text{ cm}$$

$$\lambda_1 = 0,0874 \text{ cm}$$

$$\alpha_1 = 0,0645161$$

$$\alpha_2 = 0,389052$$

$$\beta_1 = 0,188661$$

$$\beta_2 = 0,940371$$

$$\lambda_2 = 0,0782 \text{ cm}$$

$$\alpha_3 = 0.315738$$

$$\alpha_4 = 0.995112$$

$$\beta_3 = 0.710655$$

$$\beta_4 = 0.0439883$$

$$\lambda_3 = 0,0779 \text{ cm}$$

$$c = 2,728 \text{ R\$/cm}$$

Certamente, algum impeditivo arquitetônico poderia dificultar a execução de uma seção assim, ficando como sugestão para estudos futuros a imposição de limitações arquitetônicas na escolha das seções.

Além disso, o método de otimização com simulated annealing conseguiu reduzir bastante o custo da seção ao se utilizar uma seção genérica e o mapeamento de seções ótimas sugerido anteriormente pode ajudar o engenheiro na escolha de determinadas seções. No entanto, o aumento dos graus de liberdade em seções genéricas deixou o algoritmo muito demorado e o problema só tende a piorar em um caso de aplicação envolvendo diversos elementos estruturais.

Sendo assim, um estudo futuro importante seria o mapeamento citado, armazenando os resultados das otimizações sem precisar repetí-las toda vez. Ainda assim, recomenda-se incluir algumas restrições a mais na seção genérica de modo a resultarem seções arquitetonicamente viáveis (perceba que a seção da figura 7.6 é bastante peculiar) e diminuir o tempo de mapeamento.



# Referências

AFZAL, M.; LIU, Y.; CHENG, J. C.; GAN, V. J. Reinforced concrete structural design optimization: A critical review. **Journal of Cleaner Production**, Elsevier, v. 260, p. 120623, 2020.

ANWAR, N.; YAHYAH, M. Q. Parametric study of reinforced concrete column cross-section for strength and ductility. **Key Engineering Materials - KEY ENG MAT**, v. 400-402, p. 269–274, 01 2009.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6118: Projeto de estruturas de concreto - Procedimento**. Rio de Janeiro, 2014.

FERREIRO-CABELLO, J.; FRAILE-GARCIA, E.; ASCACIBAR, E.; ASCACIBAR, F. J. Metamodel-based design optimization of structural one-way slabs based on deep learning neural networks to reduce environmental impact. **Engineering Structures**, v. 155, p. 91–101, 01 2018.

FLETCHER, R.; REEVES, C. M. Function minimization by conjugate gradients. **The computer journal**, Oxford University Press, v. 7, n. 2, p. 149–154, 1964.

GARRETT, S. Chapter 13 - introductory numerical methods. *In*: GARRETT, S. (Ed.). **Introduction to Actuarial and Financial Mathematical Methods**. San Diego: Academic Press, 2015. p. 411–463. ISBN 978-0-12-800156-1. Available at: <https://www.sciencedirect.com/science/article/pii/B9780128001561000133>.

JUMBO, C. **Investigation of the Dissolution Reaction Kinetics and Mechanism of Water Interaction with Flat – Surface Ankerite using Hybrid QM/MM Simulation**. Thesis (Doutorado), 09 2011.

KOCER, F. Y.; ARORA, J. S. Design of prestressed concrete transmission poles: optimization approach. **Journal of Structural Engineering**, American Society of Civil Engineers, v. 122, n. 7, p. 804–814, 1996.

LEPŠ, M.; ŠEJNOHA, M. New approach to optimization of reinforced concrete beams. **Computers & structures**, Elsevier, v. 81, n. 18-19, p. 1957–1966, 2003.

MEDEIROS, G. A. **Programa para análise e dimensionamento da área de armadura de seções transversais de concreto armado submetidas a flexão oblíqua composta**. Trabalho de Conclusão de Curso — Instituto Tecnológico de Aeronáutica, São José dos Campos, 2004.

NETO, F. M.; PIMENTA, P. d. M. **Tópicos sobre a análise não-linear de pórticos planos de concreto armado**. Dissertation (Mestrado) — Universidade de São Paulo, 2000.

NG, A. **Gradient Descent - Coursera**. 2004. Available at:  
<https://www.coursera.org/lecture/machine-learning/gradient-descent-8SpIM>.

SHARAFI, P.; HADI, M.; TEH, L. H. A methodology for cost optimization of the layout design of multi-span reinforced concrete beams. 2013.

SHARAFI, P.; TEH, L. H.; HADI, M. N. Conceptual design optimization of rectilinear building frames: A knapsack problem approach. **Engineering Optimization**, Taylor & Francis, v. 47, n. 10, p. 1303–1323, 2015.

TOĞAN, V.; DALOĞLU, A. T. Optimization of 3d trusses with adaptive approach in genetic algorithms. **Engineering Structures**, Elsevier, v. 28, n. 7, p. 1019–1027, 2006.

YİLDİRİM, B. **Root Finding - Chapter 3**. Available at:  
<https://www.slideshare.net/batuhanyil23/es272-ch3a>.

# Apêndice A - Cálculo dos Esforços Resistentes da Distribuição Linear de Aço.

Seja uma distribuição linear (S) definida por um conjunto de pontos ordenados e uma espessura, dados os diagramas tensão deformação e o estado deformado  $(\varepsilon_0, k_x, k_y)$  da seção, é possível calcular os esforços resistentes devidos ao aço por uma integral de linha percorrendo a distribuição.

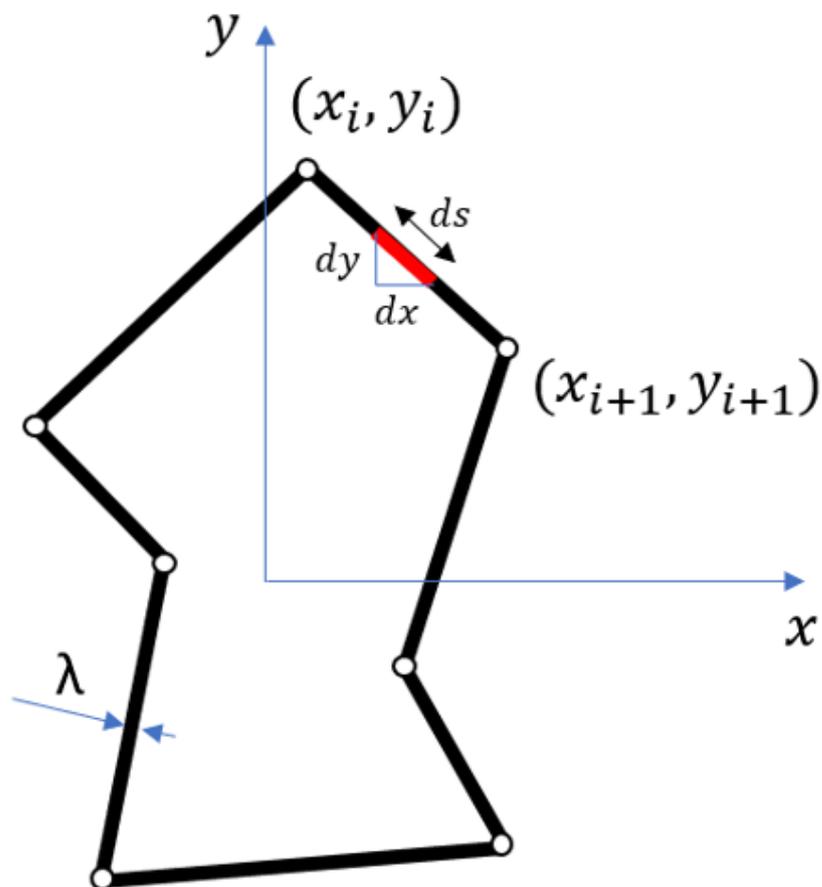


FIGURA A.1 – Distribuição linear de aço genérica (S).

Os esforços resistentes são dados por:

$$N_s = \oint_S \sigma_S(\varepsilon) ds$$

$$M_{sx} = - \oint_S \sigma_S(\varepsilon) y ds$$

$$M_{sy} = \oint_S \sigma_S(\varepsilon) x ds$$

Onde:

$$\varepsilon = \varepsilon_0 + k_y x - k_x y$$

Definindo:

$$\Delta x_i = x_{i+1} - x_i$$

$$\Delta y_i = y_{i+1} - y_i$$

$$\Delta \varepsilon_i = \varepsilon_{i+1} - \varepsilon_i$$

$$x = x_i + \beta \Delta x_i$$

$$y = y_i + \beta \Delta y_i$$

$$\varepsilon = \varepsilon_i + \beta \Delta \varepsilon_i$$

$$h_i = x_i \varepsilon_{i+1} - x_{i+1} \varepsilon_i$$

$$g_i = y_i \varepsilon_{i+1} - y_{i+1} \varepsilon_i$$

Sendo  $\beta$  um número de 0 a 1. Além disso, derivando as expressões acima:

$$dx = d\beta \Delta x_i$$

$$dy = d\beta \Delta y_i$$

$$d\varepsilon = d\beta \Delta \varepsilon_i$$

Ademais, por pitágoras:

$$ds^2 = dx^2 + dy^2$$

$$ds^2 = d\beta^2 (\Delta x_i^2 + \Delta y_i^2)$$

$$ds = d\beta \sqrt{\Delta x_i^2 + \Delta y_i^2}$$

Substituindo o  $d\beta$ :

$$ds = \frac{d\varepsilon}{\Delta\varepsilon_i} \sqrt{\Delta x_i^2 + \Delta y_i^2} \quad \Delta\varepsilon_i \neq 0$$

Feito isso, podemos realizar uma substituição de variáveis e integrar em função de  $\varepsilon$ . Desenvolveremos primeiramente a expressão para a força normal:

$$\begin{aligned} N_s &= \oint_S \sigma_s(\varepsilon) ds \\ N_s &= \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta\varepsilon_i} \int_{\varepsilon_i}^{\varepsilon_{i+1}} \sigma_s(\varepsilon) d\varepsilon \\ N_s &= \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta\varepsilon_i} \left( \int_{\varepsilon_i}^0 \sigma_s(\varepsilon) d\varepsilon + \int_0^{\varepsilon_{i+1}} \sigma_s(\varepsilon) d\varepsilon \right) \\ N_s &= \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta\varepsilon_i} \left( - \int_0^{\varepsilon_i} \sigma_s(\varepsilon) d\varepsilon + \int_0^{\varepsilon_{i+1}} \sigma_s(\varepsilon) d\varepsilon \right) \end{aligned}$$

Fazendo:

$$\int_0^x \sigma_s(\varepsilon) d\varepsilon = S_1(\varepsilon)$$

De acordo com o diagrama tensão-deformação do aço, temos:

$$S_1(\varepsilon) = \begin{cases} \left| \varepsilon f_{yd} - \frac{\varepsilon_{yd} f_{yd}}{2} \right| & |\varepsilon| > \varepsilon_{yd} \\ \frac{\varepsilon^2 f_{yd}}{2 \varepsilon_{yd}} & |\varepsilon| \leq \varepsilon_{yd} \end{cases}$$

Temos, por fim:

$$\begin{aligned} N_s &= \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta\varepsilon_i} (S_1(\varepsilon_{i+1}) - S_1(\varepsilon_i)) \\ N_s &= \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta\varepsilon_i} \Delta S_{1i} \end{aligned}$$

Caso  $\Delta\varepsilon_i = 0$  ( $\varepsilon_i = \varepsilon_{i+1}$ ) a deformação é a mesma em todos o intervalo entre os pontos  $i$  e  $i + 1$ . Nesse caso, a integral realizada é trivial:

$$N_{si} = \sigma_s(\varepsilon_i) \sqrt{\Delta x_i^2 + \Delta y_i^2} \quad \Delta\varepsilon_i = 0$$

Computacionalmente, é importante considerar uma tolerância máxima para o  $|\Delta\varepsilon_i|$  até a qual será utilizada a última expressão. Isso é importante pois divisões por valores muito próximos de zero podem ocasionar erros numéricos.

Desenvolveremos agora as expressões para o momento fletor relativo ao eixo x:

$$M_{sx} = - \oint_S \sigma_S(\varepsilon) y ds$$

$$M_{sx} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon_i}^{\varepsilon_{i+1}} \sigma_s(\varepsilon) (y_i + \beta \Delta y_i) d\varepsilon$$

$$M_{sx} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon_i}^{\varepsilon_{i+1}} \sigma_s(\varepsilon) (y_i + \frac{\varepsilon - \varepsilon_i}{\Delta \varepsilon_i} \Delta y_i) d\varepsilon$$

$$M_{sx} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} \int_{\varepsilon_i}^{\varepsilon_{i+1}} \sigma_s(\varepsilon) (g_i + \varepsilon \Delta y_i) d\varepsilon$$

Fazendo:

$$\int_0^x \varepsilon \sigma_s(\varepsilon) d\varepsilon = K_1(x)$$

De acordo com o diagrama tensão-deformação do aço, temos:

$$K_1(x) = \begin{cases} -\frac{f_{yd}\varepsilon^2}{2} + \frac{f_{yd}\varepsilon_{yd}^2}{6} & x < -\varepsilon_{yd} \\ \frac{\varepsilon^3}{3} \frac{f_{yd}}{\varepsilon_{yd}} & |x| \leq \varepsilon_{yd} \\ \frac{f_{yd}\varepsilon^2}{2} - \frac{f_{yd}\varepsilon_{yd}^2}{6} & x > \varepsilon_{yd} \end{cases}$$

Temos, por fim:

$$M_{sx} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} \int_{\varepsilon_i}^{\varepsilon_{i+1}} (g_i \Delta S_{1i} + \Delta y_i \Delta K_{1i}) d\varepsilon$$

Caso  $\Delta\varepsilon_i = 0$  ( $\varepsilon_i = \varepsilon_{i+1}$ ), basta substituir a distribuição linear por uma força aplicada no centro geométrico da reta, isto é, no ponto médio:

$$M_{sxi} = - \sigma_s(\varepsilon_i) \sqrt{\Delta x_i^2 + \Delta y_i^2} \frac{(y_i + y_{i+1})}{2} \quad \Delta\varepsilon_i = 0$$

Para o cálculo do momento fletor em relação ao eixo y o cálculo é inteiramente análogo, resultando nas expressões:

$$M_{sy} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} \int_{\varepsilon_i}^{\varepsilon_{i+1}} (h_i \Delta S_{1i} + \Delta x_i \Delta K_{1i}) d\varepsilon$$

$$M_{sxi} = - \sigma_s(\varepsilon_i) \sqrt{\Delta x_i^2 + \Delta y_i^2} \frac{(x_i + x_{i+1})}{2} \quad \Delta \varepsilon_i = 0$$

Desta forma, definiram-se as expressões necessárias para encontrar os esforços resistentes devidos à distribuição linear de aço.

Perceba que basta iterar por cada reta da distribuição. Dessa forma, há uma grande variedade de aplicação para a distribuição linear, que neste estudo está sendo utilizada como uma simplificação para uma distribuição discreta uniforme. Pode-se, por exemplo, aproximar várias seções utilizadas em elementos puramente de aço.

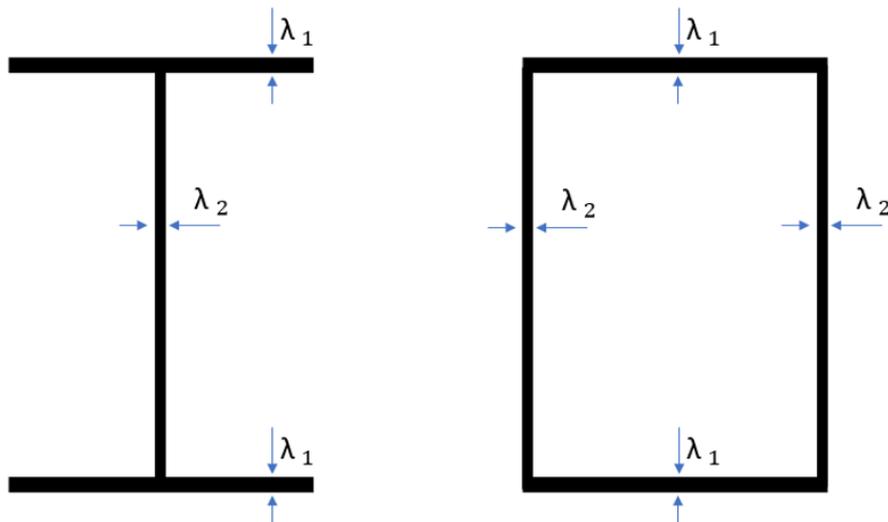


FIGURA A.2 – Seções I e caixão, muito comumente utilizadas em estruturas de aço, também podem ser calculadas como distribuições lineares de aço.



# Apêndice B - Verificação dos Esforços Resistentes da Distribuição Linear de Aço.

Como discutido anteriormente, a verificação das seções de concreto armado consiste em encontrar um estado de deformações  $(\varepsilon_0, k_x, k_y)$  no qual os esforços resistentes  $(N_R, M_{xR}, M_{yR})$  são iguais (ou, pelo menos, virtualmente iguais) aos esforços solicitantes aplicados  $(N_D, M_{xD}, M_{yD})$  para que a seção esteja equilibrada. Em seguida verifica-se se o estado de deformações atingido ultrapassa algum limite imposto pelo Estado Limite Último estabelecido por norma.

$(N_R, M_{xR}, M_{yR}) = f(\varepsilon_0, k_x, k_y)$  Cálculo dos Esforços Resistentes (Apêndices A e B)

$f(\varepsilon_0, k_x, k_y) - (N_D, M_{xD}, M_{yD}) = (0, 0, 0)$  Verificação das Seções (Apêndices C e D)

Em ambos os métodos utilizados para buscar esse estado de deformações (Newton-Raphson e Gradiente Descendente) parte-se do estado indeformado e dirige-se, com auxílio de Jacobianas ou gradientes, à raiz da função. Tanto os gradientes quanto as Jacobianas exigem o cálculo das derivadas parciais dos esforços resistentes com relação às variáveis dos estados de deformação. Calcularemos a seguir as nove, 3 esforços e 3 variáveis, derivadas parciais para o aço.

1. Força Normal ( $N_{sR}$ ) :

1. (a) Derivada parcial relativa à deformação no CG ( $\varepsilon_0$ ):

$$N_s = \oint_S \sigma_s(\varepsilon) ds$$

$$\frac{\partial N_s}{\partial \varepsilon_0} = \oint_S \frac{\partial \sigma_s(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial \varepsilon_0} ds$$

Fazendo:

$$\frac{\partial \sigma_s(\varepsilon)}{\partial \varepsilon} = D_s(\varepsilon)$$

Além disso:

$$\varepsilon = \varepsilon_0 + k_y x - k_x y \rightarrow \frac{\partial \varepsilon}{\partial \varepsilon_0} = 1$$

Logo:

$$\frac{\partial N_s}{\partial \varepsilon_0} = \oint_S D_s(\varepsilon) ds$$

Desenvolvendo conforme a figura A.2 e as definições do apêndice A, temos:

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial N_s}{\partial \varepsilon_0} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon}^{\varepsilon_{i+1}} D_s(\varepsilon) d\varepsilon$$

$$\frac{\partial N_s}{\partial \varepsilon_0} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \Delta \sigma_i$$

$$\Delta \varepsilon_i = 0 :$$

$$\frac{\partial N_s}{\partial \varepsilon_0} = \sum_{i=1}^n \sqrt{\Delta x_i^2 + \Delta y_i^2} D_{s_i}$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo x ( $k_x$ ):

$$\frac{\partial N_s}{\partial k_x} = \oint_S \frac{\partial \sigma_s(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial k_x} ds$$

$$\varepsilon = \varepsilon_0 + k_y x - k_x y \rightarrow \frac{\partial \varepsilon}{\partial k_x} = -y$$

$$\frac{\partial N_s}{\partial k_x} = - \oint_S D_s(\varepsilon) y ds$$

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon}^{\varepsilon_{i+1}} D_s(\varepsilon) y d\varepsilon$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon}^{\varepsilon_{i+1}} D_s(\varepsilon) \left( y_i + \frac{\varepsilon - \varepsilon_i}{\Delta \varepsilon_i} \Delta y_i \right) d\varepsilon$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) (g_i + \varepsilon \Delta y_i) d\varepsilon$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} (g_i \Delta \sigma_i + \Delta y_i (\Delta(\sigma \varepsilon)_i - \Delta S_{1i}))$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} (g_i \Delta \sigma_i + \Delta y_i \Delta(\sigma \varepsilon)_i - \Delta y_i \Delta S_{1i})$$

$$\Delta \varepsilon_i = 0 :$$

$$\frac{\partial N_s}{\partial k_x} = - \sum_{i=1}^n D_{si} \sqrt{\Delta x_i^2 + \Delta y_i^2} \left( y_i + \frac{\Delta y_i}{2} \right)$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo y ( $k_y$ ):

Analogamente ao caso anterior:

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial N_s}{\partial k_y} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^2} (h_i \Delta \sigma_i + \Delta x_i \Delta(\sigma \varepsilon)_i - \Delta x_i \Delta S_{1i})$$

$$\Delta \varepsilon_i = 0 :$$

$$\frac{\partial N_s}{\partial k_y} = \sum_{i=1}^n D_{si} \sqrt{\Delta x_i^2 + \Delta y_i^2} \left( x_i + \frac{\Delta x_i}{2} \right)$$

1. Momento Fletor Relativo ao Eixo x ( $M_{xR}$ ) :

1. (a) Derivada parcial relativa à deformação no CG ( $\varepsilon_0$ ):

$$M_{sx} = - \oint_S \sigma_S(\varepsilon) y ds$$

$$\frac{\partial M_{sx}}{\partial \varepsilon_0} = - \oint_S \frac{\partial \sigma_S(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial \varepsilon_0} y ds$$

$$\frac{\partial M_{sx}}{\partial \varepsilon_0} = - \oint_S D_s(\varepsilon) y ds = \frac{\partial N_s}{\partial k_x}$$

Sendo assim, as expressões são as mesmas que as já calculadas no caso anterior:

$$\frac{\partial M_{sx}}{\partial \varepsilon_0} = \frac{\partial N_s}{\partial k_x}$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo x ( $k_x$ ):

$$\frac{\partial M_{sx}}{\partial k_x} = - \oint_S \frac{\partial \sigma_S(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial k_x} y ds$$

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial M_{sx}}{\partial k_x} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) y^2 d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_x} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) (g_i + \varepsilon \Delta y_i)^2 d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_x} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) (g_i^2 + 2g_i \varepsilon \Delta y_i + \varepsilon^2 \Delta y_i^2) d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_x} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} (g_i^2 \Delta \sigma_i + 2g_i \Delta y_i (\Delta(\sigma\varepsilon)_i - \Delta S_{1i}) + \Delta y_i^2 (\Delta(\sigma\varepsilon^2)_i - 2\Delta K_{1i}))$$

$$\Delta \varepsilon_i = 0 :$$

$$\frac{\partial M_{sx}}{\partial k_x} = \sum_{i=1}^n D_{Si} \sqrt{\Delta x_i^2 + \Delta y_i^2} \left( y_i^2 + y_i \Delta y_i + \frac{\Delta y_i^2}{3} \right)$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo y ( $k_y$ ):

$$\frac{\partial M_{sx}}{\partial k_y} = - \oint_S \frac{\partial \sigma_S(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial k_y} y ds$$

$$\varepsilon = \varepsilon_0 + k_y x - k_x y \rightarrow \frac{\partial \varepsilon}{\partial k_y} = x$$

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial M_{sx}}{\partial k_y} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) x y d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_y} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) (g_i + \varepsilon \Delta y_i) (h_i + \varepsilon \Delta x_i) d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_y} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} \int_{\varepsilon_i}^{\varepsilon_{i+1}} D_s(\varepsilon) (g_i h_i + g_i \varepsilon \Delta x_i + h_i \varepsilon \Delta y_i + \varepsilon^2 \Delta y_i \Delta x_i) d\varepsilon$$

$$\frac{\partial M_{sx}}{\partial k_y} = - \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} (g_i h_i \Delta \sigma_i + (g_i \Delta x_i + h_i \Delta y_i) (\Delta(\sigma\varepsilon)_i - \Delta S_{1i}) + \Delta x_i \Delta y_i (\Delta(\sigma\varepsilon^2)_i - 2\Delta S_{2i}))$$

$\Delta \varepsilon_i \neq 0 :$

$$\frac{\partial M_{sx}}{\partial k_y} = - \sum_{i=1}^n D_{Si} \sqrt{\Delta x_i^2 + \Delta y_i^2} \left( x_i y_i + \frac{x_i \Delta y_i + y_i \Delta x_i}{2} + \frac{\Delta x_i \Delta y_i}{3} \right)$$

1. Momento Fletor Relativo ao Eixo y ( $M_{yR}$ ) :

1. (a) Derivada parcial relativa à deformação no CG ( $\varepsilon_0$ ):

$$M_{sy} = \oint_S \sigma_S(\varepsilon) x ds$$

$$\frac{\partial M_{sy}}{\partial \varepsilon_0} = \oint_S \frac{\partial \sigma_S(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial \varepsilon_0} x ds$$

$$\frac{\partial M_{sy}}{\partial \varepsilon_0} = \oint_S D_s(\varepsilon) x ds = \frac{\partial N_s}{\partial k_y}$$

Sendo assim, as expressões são as mesmas que as já calculadas no caso anterior:

$$\frac{\partial M_{sy}}{\partial \varepsilon_0} = \frac{\partial N_s}{\partial k_y}$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo x ( $k_x$ ):

$$M_{sy} = \oint_S \sigma_S(\varepsilon) x ds$$

$$\frac{\partial M_{sy}}{\partial k_x} = \oint_S \frac{\partial \sigma_S(\varepsilon)}{\partial \varepsilon} \frac{\partial \varepsilon}{\partial k_x} x ds$$

$$\frac{\partial M_{sy}}{\partial k_x} = - \oint_S D_s(\varepsilon) y x ds = \frac{\partial M_{sx}}{\partial k_y}$$

Sendo assim, as expressões são as mesmas que as já calculadas no caso anterior:

$$\frac{\partial M_{sy}}{\partial k_x} = \frac{\partial M_{sx}}{\partial k_y}$$

1. (a) Derivada parcial relativa à inclinação do plano em torno do eixo y ( $k_y$ ):

Analogamente ao caso  $\frac{\partial M_{sx}}{\partial k_x}$ :

$$\Delta \varepsilon_i \neq 0 :$$

$$\frac{\partial M_{sy}}{\partial k_y} = \sum_{i=1}^n \frac{\sqrt{\Delta x_i^2 + \Delta y_i^2}}{\Delta \varepsilon_i^3} (h_i^2 \Delta \sigma_i + 2h_i \Delta x_i (\Delta (\sigma \varepsilon)_i - \Delta S_{1i}) + \Delta x_i^2 (\Delta (\sigma \varepsilon^2)_i - 2\Delta K_{1i}))$$

$$\Delta \varepsilon_i = 0 :$$

$$\frac{\partial M_{sy}}{\partial k_y} = \sum_{i=1}^n D_{Si} \sqrt{\Delta x_i^2 + \Delta y_i^2} \left( x_i + x_i \Delta x_i + \frac{\Delta x_i^2}{3} \right)$$

Temos, finalmente, as 9 expressões das derivadas parciais que serão utilizadas nos métodos numéricos Newton-Raphson e gradiente descendente. Definidas em função das propriedades materiais, geométricas e estado de deformação. Perceba que os termos simétricos da matriz Jacobiana são iguais, sendo assim, a matriz Jacobiana e só é necessário calcular 6 dos 9 termos.

$$J = \begin{pmatrix} \frac{\partial N_s}{\partial \varepsilon_0} & \frac{\partial N_s}{\partial k_x} & \frac{\partial N_s}{\partial k_y} \\ & \frac{\partial M_{sx}}{\partial k_x} & \frac{\partial M_{sx}}{\partial k_y} \\ sim. & & \frac{\partial M_{sy}}{\partial k_y} \end{pmatrix}$$



# Apêndice C - Código implementado em C++.

Todo o modelamento descrito ao longo do presente trabalho foi implementado inicialmente no MATLAB (exceto a parte de otimização utilizando Simulated Annealing). No entanto, as otimizações realizadas no MATLAB estavam demorando bastante para serem concluídas. Quando a seção não converge, o algoritmo costuma realizar milhares de iterações antes de desistir de encontrar uma raiz para a verificação da seção, nesses casos cada otimização demorava cerca de 30 segundos.

Como o método simulated annealing precisa trabalhar com centenas de seções, realizando milhares de iterações até concluir a otimização, esse tempo inviabilizaria o processo. Sendo assim, decidiu-se implementar o mesmo modelo de verificação em linguagem C++, diminuindo drasticamente o tempo de execução. Mesmo nos casos em que a seção não convergia, a execução dura menos que um segundo.

O algoritmo utilizado implementou a classe seção de concreto armado, composta por pontos de concreto e aço como listas ligadas, bem como as propriedades dos materiais. Foram implementadas também todas as funções necessárias, dentro e fora da classe, para realizar os cálculos necessários para as verificações.

Além disso, toda a otimização com o simulated annealing foi também realizada no C++, porém não foi implementada nenhuma ferramenta de visualização nessa linguagem. As visualizações criadas foram tratamentos manuais de dados resultantes das otimizações cujo código fornecia em formato texto.

A seguir, será mostrado parte do código relativa a parte da implementação (a maior parte foi omitida, podendo ser encontrada no suplemento do trabalho) da classe seção de concreto armado e estruturas de dados auxiliares, além disso ao final é mostrada a função na qual é implementado o método AUSA (bestChromosome()):

```
struct point {  
    long double x;  
    long double y;
```

```
    long double steelArea;
    long double linearSteelThickness;
    struct point *nextPoint;
};

struct stresses {
    long double x = 0;
    long double y = 0;
    long double normalStress;
    long double xBendingMoment;
    long double yBendingMoment;
    bool safe;
};

struct strainState {
    long double x = 0.0;
    long double y = 0.0;
    long double e0; // strain in the origin
    long double kx; // variation of strain per distance from x axis
    long double ky; // variation of strain per distance from y axis
    bool safe;
};

class polygon {
private:
public:
    point points[200];
    int numPoints = 0;

    void pushPoint(point newPoint){
        points[numPoints] = newPoint;
        numPoints++;
    }

    long double minX() {
        long double minimum = __FLT_MAX__;
        for (int i = 0; i < numPoints; i++) {
            if (points[i].x < minimum) {
```

```
        minimum = points[i].x;
    }
}
return minimum;
}

long double maxX() {
    \ \ omitted
}

long double minY() {
    \ \ omitted
}

long double maxY() {
    \ \ omitted
}
};

class crossSection {
private:
    polygon concreteSection;
    polygon steelLinearDist;

public:
    long double fyk = 50.0; //featureYieldKnown
    long double fck = 4.0; //featureCompressionKnown
    long double E = 21000.0; //youngModulus
    long double r = 0.00001; //tolerance
    long double p = 0.00000001;
    long double t = 0.000000000001;

    void defineConcreteSection(polygon section){
        concreteSection = section;
    }

    void defineSteelLinearDist(polygon linearDist) {
        steelLinearDist = linearDist;
    }
}
```

```
long double Ac() {
    long double soma = 0;
    for (int i = 0; i < concreteSection.numPoints; i++) {
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xI = concreteSection.points[i].x;
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yI = concreteSection.points[i].y;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double aI = xI*yIplus1 - xIplus1*yI;
        soma += aI;
    }
    return soma / 2.0;
}

long double Sy() {
    long double soma = 0.0;
    for (int i = 0; i < concreteSection.numPoints; i++) {
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xI = concreteSection.points[i].x;
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yI = concreteSection.points[i].y;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double aI = xI*yIplus1 - xIplus1*yI;
        soma += aI * (xI + xIplus1);
    }
    return soma / 6.0;
}
```

```
long double Sx() {
    \N omitted
}

long double Iyy() {
    long double soma = 0.0;
    for (int i = 0; i < concreteSection.numPoints; i++) {
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xI = concreteSection.points[i].x;
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yI = concreteSection.points[i].y;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double aI = xI*yIplus1 - xIplus1*yI;
        soma += aI * (pow(xI,2) + xI*xIplus1 + pow(xIplus1,2));
    }
    return soma / 12.0;
}

long double Ixy() {
    \N omitted
}

long double Ixx() {
    \N omitted
}

long double NcFOC(strainState e0kxky) {
    long double sigmacd = stressCompressDesign(fck);
    long double e0 = e0kxky.e0;
    long double kx = e0kxky.kx;
    long double ky = e0kxky.ky;
    long double soma = 0.0;
    if (abs(kx)<r && abs(ky)<r) {
        return sigmac(e0,sigmacd,n(fck),ec2(fck))*Ac();
    }
}
```

```
} else if (abs(kx) >= r) {
    for (int i = 0; i < concreteSection.numPoints; i++) {
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xI = concreteSection.points[i].x;
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yI = concreteSection.points[i].y;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double epsI = e0 + ky*xI - kx*yI;
        long double epsIplus1 = e0 + ky*xIplus1 - kx*yIplus1;
        soma += (xIplus1 - xI) * f1i(epsIplus1, epsI, fck, r);
    }
    return (1.0 / kx) * soma;
} else {
    for (int i = 0; i < concreteSection.numPoints; i++) {
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xI = concreteSection.points[i].x;
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yI = concreteSection.points[i].y;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double epsI = e0 + ky*xI - kx*yI;
        long double epsIplus1 = e0 + ky*xIplus1 - kx*yIplus1;
        soma += (yIplus1 - yI) * f1i(epsIplus1, epsI, fck, r);
    }
    return (1.0 / ky) * soma;
}
}
```

```
long double McxFOC(strainState e0kxky) {
    \ \ omitted
}

long double McyFOC(strainState e0kxky) {
    \ \ omitted
}

long double NsFOC(strainState e0kxky) {
    \ \ omitted
}

long double MsxFOC(strainState e0kxky) {
    \ \ omitted
}

long double MsyFOC(strainState e0kxky) {
    \ \ omitted
}

long double NtotFOC(strainState e0kxky) {
    long double Nc = NcFOC(e0kxky);
    long double Ns = NsFOC(e0kxky);
    return Nc + Ns;
}

long double MxtotFOC(strainState e0kxky) {
    long double Mcx = McxFOC(e0kxky);
    long double Msx = MsxFOC(e0kxky);
    return Mcx + Msx;
}

long double MytotFOC(strainState e0kxky) {
    long double Mcy = McyFOC(e0kxky);
    long double Msy = MsyFOC(e0kxky);
    return Mcy + Msy;
}

long double** Jc(strainState e0kxky) {
```

```
long double** sum2D;
sum2D = new long double*[3];
for (int i = 0; i < 3; i++) {
    sum2D[i] = new long double[3];
    for (int j = 0; j < 3; j++) {
        sum2D[i][j] = 0;
    }
}

long double e0 = e0kxky.e0;
long double kx = e0kxky.kx;
long double ky = e0kxky.ky;

if (abs(kx) > r) {
    for (int i = 0; i < concreteSection.numPoints; i++) {
        long double xI = concreteSection.points[i].x;
        long double yI = concreteSection.points[i].y;
        long double epsI = e0 + ky * xI - kx * yI;

        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {
            Iplus1 = 0;
        }
        long double xIplus1 = concreteSection.points[Iplus1].x;
        long double yIplus1 = concreteSection.points[Iplus1].y;
        long double epsIplus1 = e0 + ky * xIplus1 - kx * yIplus1;

        long double dx = xIplus1 - xI;

        long double f6if = f6i(epsIplus1, epsI, fck, r);
        long double f7if = f7i(epsIplus1, epsI, fck, yI,
yIplus1, r);
        long double f8if = f8i(epsIplus1, epsI, fck, xI,
xIplus1, r);
        long double f9if = f9i(epsIplus1, epsI, fck, yI,
yIplus1, r);
        long double f10if = f10i(epsIplus1, epsI, fck, yI,
```

```

        yIplus1, xI, xIplus1, r);
    long double f11if = f11i(epsIplus1, epsI, fck, xI,
        xIplus1, r);

    long double sum[3][3] = {{dx*f6if, -dx*f7if, dx*f8if},
        {-dx*f7if, dx*f9if, -dx*f10if},
        {dx*f8if, -dx*f10if, dx*f11if}};

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            sum2D[i][j] += sum[i][j];
        }
    }

}

long double Ncf = NcFOC(e0kxky);
long double Mcxf = McxFOC(e0kxky);
long double Mcyf = McyFOC(e0kxky);

long double sub[3][3] = {{0, Ncf, 0},
    {Ncf, 2*Mcxf, Mcyf},
    {0, Mcyf, 0}};

for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 3; j++) {
        sum2D[i][j] = (1.0/kx) * (sum2D[i][j] - sub[i][j]);
    }
}

return sum2D;

} else if (abs(ky) > r) {
    for (int i = 0; i < concreteSection.numPoints; i++) {
        long double xI = concreteSection.points[i].x;
        long double yI = concreteSection.points[i].y;
        long double epsI = e0 + ky * xI - kx * yI;
        int Iplus1;
        if (i != concreteSection.numPoints - 1) {
            Iplus1 = i + 1;
        } else {

```

```

        Iplus1 = 0;
    }
    long double xIplus1 = concreteSection.points[Iplus1].x;
    long double yIplus1 = concreteSection.points[Iplus1].y;
    long double epsIplus1 = e0 + ky * xIplus1 - kx * yIplus1;

    long double dy = yIplus1 - yI;

    long double f6if = f6i(epsIplus1, epsI, fck, r);
    long double f7if = f7i(epsIplus1, epsI, fck, yI,
    yIplus1, r);
    long double f8if = f8i(epsIplus1, epsI, fck, xI,
    xIplus1, r);
    long double f9if = f9i(epsIplus1, epsI, fck, yI,
    yIplus1, r);
    long double f10if = f10i(epsIplus1, epsI, fck, yI,
    yIplus1, xI, xIplus1, r);
    long double f11if = f11i(epsIplus1, epsI, fck, xI,
    xIplus1, r);

    long double sum[3][3] = {{dy*f6if, -dy*f7if, dy*f8if},
                            {-dy*f7if, dy*f9if, -dy*f10if},
                            {dy*f8if, -dy*f10if, dy*f11if}};

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            sum2D[i][j] += sum[i][j];
        }
    }

}

long double Ncf = NcFOC(e0kxky);
long double Mcxf = McxFOC(e0kxky);
long double Mcyf = McyFOC(e0kxky);

long double sub[3][3] = {{0, 0, Ncf},
                        {0, 0, Mcxf},
                        {Ncf, Mcxf, 2*Mcyf}};

```

```

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            sum2D[i][j] = (1.0/ky) * (sum2D[i][j] - sub[i][j]);
        }
    }
    return sum2D;
} else {
    long double dc = Dc(e0, fck);
    long double result[3][3] = {{dc*Ac(), dc*Sx(), dc*Sy()},
                                {dc*Sx(), dc*Ixx(), -dc*Ixy()},
                                {dc*Sy(), -dc*Ixy(), dc*Iyy()}};

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            sum2D[i][j] = result[i][j];
        }
    }
    return sum2D;
}
}

long double** Js(strainState e0kxky) {
    \N omitted
}

strainState e0kxky (stresses nMxMy) {
    long double Nd = nMxMy.normalStress;
    long double Mxd = nMxMy.xBendingMoment - Nd * nMxMy.y;
    long double Myd = nMxMy.yBendingMoment + Nd * nMxMy.x;
    long double sigmacd = stressCompressDesign(fck);

    strainState e0kxky = {0.0, 0.0, 0.0, 0.0, 0.0};

    bool over = false;
    int counter = 0;
    long double f_checkpoint = __FLT_MAX__;

    while (!over) {

        long double Nc = NcFOC(e0kxky);

```

```
long double Mcx = McxFOC(e0kxky);
long double Mcy = McyFOC(e0kxky);

long double Ns = NsFOC(e0kxky);
long double Msx = MsxFOC(e0kxky);
long double Msy = MsyFOC(e0kxky);

long double Nt = Ns + Nc;
long double Mxt = Mcx + Msx;
long double Myt = Mcy + Msy;

long double Ac_ = this->Ac();

long double maxY = - __FLT_MAX__;
long double minY = __FLT_MAX__;

for (int i = 0; i < concreteSection.numPoints; i++) {
    long double yI = concreteSection.points[i].y;
    if (yI > maxY) {
        maxY = yI;
    }
    if (yI < minY) {
        minY = yI;
    }
}

long double h = maxY - minY;
long double f = sqrt(pow(((Nd-Nt)/(sigmacd*Ac_)),2)
                    +pow(((Mxd-Mxt)/(sigmacd*Ac_*h)),2)
                    +pow(((Myd-Myt)/(sigmacd*Ac_*h)),2));

if (f <= p) {
    over = true;
    bool broke = brokeSection(e0kxky);
    if (broke) {
        e0kxky.safe = false;
        return e0kxky;
    } else {
```

```
        e0kxky.safe = true;
        return e0kxky;
    }
} else {

    long double** Jcf = Jc(e0kxky);
    long double** Jsf = Js(e0kxky);

    long double Jt[3][3] {};

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            Jt[i][j] = Jsf[i][j] + Jcf[i][j];
        }
    }

    for (int i = 0; i < 3; i++) {
        delete Jcf[i];
        delete Jsf[i];
    }
    delete [] Jcf;
    delete [] Jsf;

    long double G = determinant(Jt, 3.0);

    long double dStresses[3] {};
    dStresses[0] = Nd - Nt;
    dStresses[1] = Mxd - Mxt;
    dStresses[2] = Myd - Myt;

    if (abs(G) > t) {

        long double Dx = determinant(
            Jt, 3.0, true, dStresses, 0.0);
        long double delE0 = Dx / G;
        long double Dy = determinant(
            Jt, 3.0, true, dStresses, 1.0);
        long double delKx = Dy / G;
        long double Dz = determinant(
```

```
        Jt, 3.0, true, dStresses, 2.0);
    long double delKy = Dz / G;

    e0kxky.e0 += delE0;
    e0kxky.kx += delKx;
    e0kxky.ky += delKy;

    } else {
        over = true;
        e0kxky.safe = false;
        return e0kxky;
    }

}

counter++;

if (counter % 1000 == 0) {
    if (f > 0.9*f_checkpoint) {
        over = true;
        e0kxky.safe = false;
        return e0kxky;
    }
    f_checkpoint = f;

    if (counter > 100000) {
        over = true;
        e0kxky.safe = false;
        return e0kxky;
    }
}

}

}

long double* maxminStrainsConcreteSteel(
    strainState e0kxky, long double *extremes) {
    \\ omitted
}
```

```
bool brokeSection(strainState e0kxky) {
    \N omitted
};

string bestChromosome(stresses appliedStresses, int typeSection) {

    string bestOfAll;
    double minCostOfAll = __FLT_MAX__;
    int sizePopulation = 100;
    string* randomPopulation;
    randomPopulation = new string[sizePopulation];
    double minCost = __FLT_MAX__;
    int minIndex = NULL;

    for (int i = 0; i < sizePopulation; i++) {
        bool aliveMember = false;
        double cost = __FLT_MAX__;
        while (!aliveMember) {
            randomPopulation[i] = generateBinaryString(
                typeSection*55 - 10);
            aliveMember = offspringSurvived(
                randomPopulation[i], appliedStresses, typeSection);
            cost = rectBasedCostPerLength(
                randomPopulation[i], typeSection);
        }
        if (cost < minCost) {
            minCost = cost;
            minIndex = i;
            minCostOfAll = minCost;
            bestOfAll = randomPopulation[i];
        }
    }

    double maxTemperature = 0.4;
    double temperature = maxTemperature;
    double temperatureDecay = 0.9;
    int epoch = 1;
```

```
int counter = 0;
int successCrossovers = 0;
int counterMax = 1000;
int successCrossoversMax = 100;

while (epoch < 10000) {
    counter = 0;
    successCrossovers = 0;
    counterMax = 100;
    successCrossoversMax = 100;

    while (counter < counterMax &&
           successCrossovers < successCrossoversMax) {
        counter += 1;
        performOperation(randomPopulation, 0.1, sizePopulation,
                        appliedStresses, temperature, typeSection);
    }
    temperature *= temperature * temperatureDecay;

    minIndex = minCostIndex(
        randomPopulation, sizePopulation, typeSection);
    minCost = rectBasedCostPerLength(
        randomPopulation[minIndex], typeSection);

    if (minCost < minCostOfAll) {
        minCostOfAll = minCost;
        bestOfAll = randomPopulation[minIndex];
    }
    epoch += 1;

    if ((epoch % 1000) == 0) {
        temperature *= 0.9;
    }

}

return bestOfAll;
}
```

## FOLHA DE REGISTRO DO DOCUMENTO

1. CLASSIFICAÇÃO/TIPO TC	2. DATA 22 de novembro de 2021	3. DOCUMENTO Nº DCTA/ITA/TC-101/2021	4. Nº DE PÁGINAS 105
5. TÍTULO E SUBTÍTULO: Aplicação do método augmented simulated annealing para otimização de seções de concreto armado sob flexão oblíqua			
6. AUTOR(ES): <b>Gabriel Mendes Cabral Gondim</b>			
7. INSTITUIÇÃO(ÕES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÕES): Instituto Tecnológico de Aeronáutica – ITA			
8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR: Concreto armado, Seção genérica, Flexão oblíqua, Otimização, <i>Augmented simulated annealing</i>			
9. PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO: Concreto armado; Estruturas de concreto; Recozimento simulador (matemática); Otimização; Calculo estrutural; Engenharia estrutural; Engenharia civil.			
10. APRESENTAÇÃO: <span style="float: right;"><input checked="" type="checkbox"/> Nacional    <input type="checkbox"/> Internacional</span> ITA, São José dos Campos. Curso de Graduação em Engenharia Civil-Aeronáutica. Orientador: Prof. Dr. Francisco Alex Correia Monteiro; co-orientador: Prof. Dr. Sérgio Gustavo Ferreira Cordeiro. Publicado em 2021.			
11. RESUMO: <p>A concepção estrutural de uma edificação hoje ainda depende muito da criatividade e experiência do engenheiro responsável. Cabe a ele escolher o posicionamento de pilares, vigas e fundações que se adequem ao projeto arquitetônico e que resultem em uma estrutura barata e confiável. Feita a concepção estrutural, é prática comum utilizar softwares estruturais para que estes otimizem a utilização de aço dentro dos elementos. No entanto, ao otimizar apenas o aço, mantendo fixos os parâmetros de posição, forma e orientação das seções, é pouco provável que o mínimo global seja encontrado. O que se encontra é o mínimo para aquela configuração estrutural específica.</p> <p>Pensando nisso, o presente trabalho implementa um método de otimização que leva em consideração todos os fatores mencionados acima, visando um resultado mais próximo de um mínimo global de custo. Uma seção genérica foi idealizada de modo que esta seção pode se particularizar nas seções mais utilizadas da engenharia civil, exceto seção circular. No entanto, os graus de liberdade aumentam bastante, levando à simplificação das armaduras com uma distribuição linear, para resolução em tempo hábil. Portanto, o modelo apresentado aqui se trata de um metamodelo que pretende, inicialmente, auxiliar o engenheiro a encontrar uma concepção estrutural otimizada. Porém não retorna um projeto detalhado pronto para a execução.</p> <p>Durante a otimização, deve-se sempre verificar se as seções suportam os esforços solicitantes. Para isso foi implementado um modelo computacional para cálculo de esforços resistentes e verificação de seções sob a ótica do ELU (Estado Limite Último) previsto pela norma técnica brasileira NBR-6118. A modelo utilizado foi baseado no trabalho do (NETO; PIMENTA, 2000), com adição de distribuições lineares de aço e estudo da aplicação do método do gradiente conjugado para a verificação das seções.</p> <p>A inclusão da distribuição linear de aço possibilitou ainda o estudo de seções não só de concreto armado, mas também seções mistas ou seções de aço, para as quais foram gerados gráficos das envoltórias de segurança de deformações e esforços.</p> <p>Por fim, o modelo implementado foi utilizado em um processo de otimização com o método Augmented Simulated Annealing (AUSA) de acordo com o estudo (LEPŠ; ŠEJNOHA, 2003). Os parâmetros da seção genérica idealizada foram expressos em uma string de binários, denominada cromossomo, que define um indivíduo em uma população. Esses cromossomos passam por mutações e reproduções, sendo selecionados os indivíduos com menor função custo e que resistem aos esforços solicitantes. Após um número predeterminado de iterações, o algoritmo retorna o indivíduo válido mais barato já visto.</p>			
12. GRAU DE SIGILO: <input checked="" type="checkbox"/> <b>OSTENSIVO</b> <input type="checkbox"/> <b>RESERVADO</b> <input type="checkbox"/> <b>SECRETO</b>			