

Modelagem BIM e otimização com algoritmo genético de coberturas treliçadas

Alexandre L. Bitencourt¹, Silvia Corbani¹, Juarez M. Franco²

¹Dept. de Estruturas, Universidade Federal do Rio de Janeiro

Av. Athos da Silveira Ramos, 149, bloco D, sala D-205, Cidade Universitária, 21941-972, Rio de Janeiro, RJ, Brasil

alebitencourt@poli.com.br, corbani@poli.com.br

²Dept. Arquitetura e Urbanismo, Universidade Federal Rural do Rio de Janeiro

BR 465 km 7, 23851-970, Seropédica, RJ, Brazil

juarezfranco@poli.ufrrj.br

Abstract. O advento da tecnologia Building Information Modeling (BIM) introduziu uma mudança de paradigma no desenvolvimento de projetos estruturais na engenharia civil. Os modelos computacionais em BIM são capazes de armazenar mais dados sobre elementos e sistemas do que as tecnologias CAD e CAE que a precederam. Em lugar das usuais simplificações em engenharia, as plataformas BIM integram ferramentas de programação, modelagem geométrica e simulação de desempenho em projetos de qualquer grau de complexidade, aumentando a produtividade e reduzindo a perda de informação entre diferentes etapas de trabalho. O maior volume de dados disponível torna mais difícil, em projetos de geometria incomum, que o projetista alcance soluções de alto desempenho sem o auxílio de ferramentas de Inteligência Artificial (IA). Dado o elevado custo computacional dos aplicativos BIM e das tecnologias IA em geral, faz-se necessária averiguar os limites e as potencialidades destas ferramentas no projeto de estruturas, face à capacidade de processamento dos computadores pessoais disponíveis atualmente. Este trabalho visa empregar o algoritmo genético NSGA-II (otimização multicritério com fronteira de Pareto) para determinar parâmetros de geometria e seção transversal dos perfis de uma cobertura de aço composta por vigas treliçadas planas. A implementação do algoritmo é realizada no ambiente Dynamo disponível no Revit, um programa BIM, enquanto os esforços solicitantes são determinados através do Robot Structural Analysis. São estudadas treliças com vão fixo e diferentes parâmetros de geometria, como, alturas, inclinações, número de diagonais e montantes, para avaliar a configuração que consome menos material. Um dos principais desafios envolvidos nesta tarefa é a proposição das funções de aptidão, que abarcam os objetivos a serem maximizados e as restrições de projeto que não podem ser violadas, de modo a informar automaticamente aos algoritmos de otimização os ajustes a serem efetuados no projeto. Embora as implementações realizadas nesta pesquisa indiquem que a programação nos sistemas BIM pode elevar o custo computacional das tarefas de otimização, este estudo empregando o algoritmo genético NSGA-II teve um consumo de aço menor quando comparado a um valor de referência que foi o consumo de aço de uma cobertura treliçada dimensionada conforme procedimento de escritórios de projeto. É possível diminuir o consumo de material neste valor de referência, mas o processo de redefinição dos parâmetros da estrutura e modelagem da mesma tomaria tempo do projetista.

Palavras-chaves: BIM; modelagem paramétrica; otimização; algoritmos genéticos; estruturas de aço.

1 Introdução

A computação tornou possível, a partir da segunda metade do século XX, a modelagem estrutural de sistemas espaciais em uma fração do tempo necessário para resolver os mesmos problemas por técnicas manuais, oferecendo ao projetista a possibilidade de representar uma gama maior de alternativas de projeto, fazer simulações sobre o comportamento destes sistemas e visualizar resultados de maneira interativa. Antes do advento das tecnologias BIM, as tarefas de geração geométrica, de análise e de levantamento de quantitativos era executada por diferentes ferramentas computacionais. A princípio, a baixa interoperabilidade entre elas implicava em perdas

de informação a cada etapa onde havia necessidade de migrar de uma plataforma para outra, gerando riscos de incompatibilidade entre decisões tomadas por equipes distintas. A centralização da informação sobre o projeto da edificação em um único arquivo, por um lado, reduz consideravelmente estes riscos. Por outro lado, a complexidade dos modelos BIM, aliada às atuais limitações de processamento, aumentam os custos computacionais de tarefas que, em plataformas mais simples, são execução relativamente simples. Como exemplo, tratamos neste trabalho da concepção de um projeto de estruturas de aço através da modelagem paramétrica, desenvolvido em um ambiente de programação com interface visual, com algoritmos genéticos de otimização. O objetivo deste artigo é investigar resultados que sugerem as potencialidades desta ferramenta e compreender suas limitações atuais.

Diversos autores têm contribuído para o amadurecimento desta linha de pesquisa. Korus, Salamak, Jasński [1] desenvolveram um estudo sobre projeto em BIM de ponte suspensa em arcos metálicos de seção tubular quadrada com modelagem paramétrica e cálculo estrutural na ferramenta Dynamo [3]. Dentre os diversos parâmetros, foram analisadas quatro disposições para os cabos que sustentam o tabuleiro. Os autores concluem que sua implementação resultou em custos computacionais de otimização considerados aceitáveis e nenhuma perda de capacidade de processamento associada a complexidade formal da ponte. Todavia, os autores destacam que diversas análises não foram consideradas em sua implementação, como efeitos de segunda ordem e o comportamento dinâmico das estruturas, o que pode ser inspiração para futuros trabalhos na área.

Kravanja et al. [2] fizeram uma análise de treliças de cobertura cuja forma permite a entrada de luz natural, que foi um dos seus objetivos. Desta forma, foi feita uma análise multidisciplinar, visto que foram analisados três objetivos distintos: incidência de luz solar, desempenho da estrutura em relação à esforços e deslocamento, e por último, o custo de material. O estudo é feito através de dois modelos distintos: análise de uma treliça biapoada que faz parte da cobertura e um modelo desta mesma cobertura, na qual a passagem de luz depende da altura desta treliça. Para solucionar este problema multi-objetivo, os autores empregaram o algoritmo genético NSGA-II (otimização multicritério com fronteira de Pareto) [5]. Em sua conclusão os autores alegam que o emprego destas técnicas pode auxiliar o projetista a encontrar soluções de alta performance, envolvendo diferentes áreas, ainda em estágios iniciais de projeto.

Neste trabalho, a estrutura analisada será uma treliça plana do tipo Pratt com duas águas, formada por perfis de seções tubulares retangulares e quadradas na qual será aplicado o algoritmo de otimização NSGA-II. Para comparar o consumo de aço, será utilizada uma estrutura com parâmetros fixos de acordo com literatura existente. Para realizar a modelagem paramétrica e para emprego do algoritmo genético (AG) de otimização, a rotina implementada neste trabalho considera o Estado-Limite Último (ELU) para a força axial de tração resistente de cálculo e a força axial de compressão resistente de cálculo e Estado-Limite de Serviço (ELS) no controle de deslocamentos. Esta rotina é uma ampliação da implementação de Vermeulen [3] por avaliar mais estados limites e por considerar outra função de minimização (função objetivo) no problema de otimização.

O programa Dynamo, que é um ambiente de programação visual distribuído pela Autodesk e associado ao programa Revit, foram utilizadas rotinas (chamadas de pacotes) desenvolvidas pela comunidade de usuários e disponíveis para *download* em uma biblioteca acessada no mesmo. O pacote Optimo foi utilizado para implementar o AG, o pacote *Structural Analysis* foi utilizado para criar o modelo computacional no programa de análise *Robot Structural Analysis* da empresa Autodesk. Por fim, foram implementados códigos em linguagem Python para obter informações adicionais às fornecidas pelo pacote *Structural Analysis* e também para definição da função objetivo que será utilizada no pacote Optimo. Na Figura 1 é ilustrada relação de compartilhamento de informações em um fluxograma com todos os programas e pacotes utilizados e sua funcionalidade no presente estudo. Na Figura 2 a descrição do problema estudado neste trabalho é esquematicamente mostrada.

Através deste estudo de caso, pretende-se avaliar o desempenho desta implementação em um ambiente BIM, contribuindo assim para o entendimento da fronteira em que tal tecnologia se encontra, sobretudo para usuários que não dispõem de elevado custo computacional ou uma equipe especializada para o emprego de tais técnicas. O computador utilizado neste estudo tem processador Ryzen 5 3600 6-Core Processor; memória instalada (RAM): 16 GB e sistema Operacional de 64 bits, processador com base em x64.

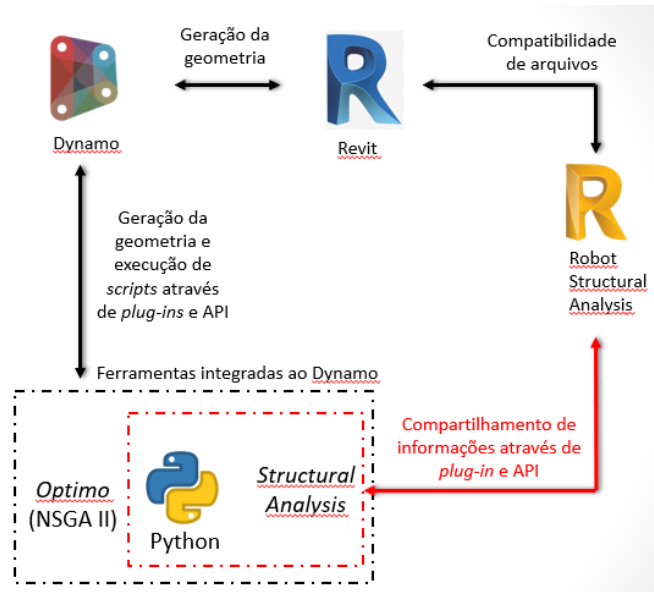


Figura 1. Programas empregados neste estudo.

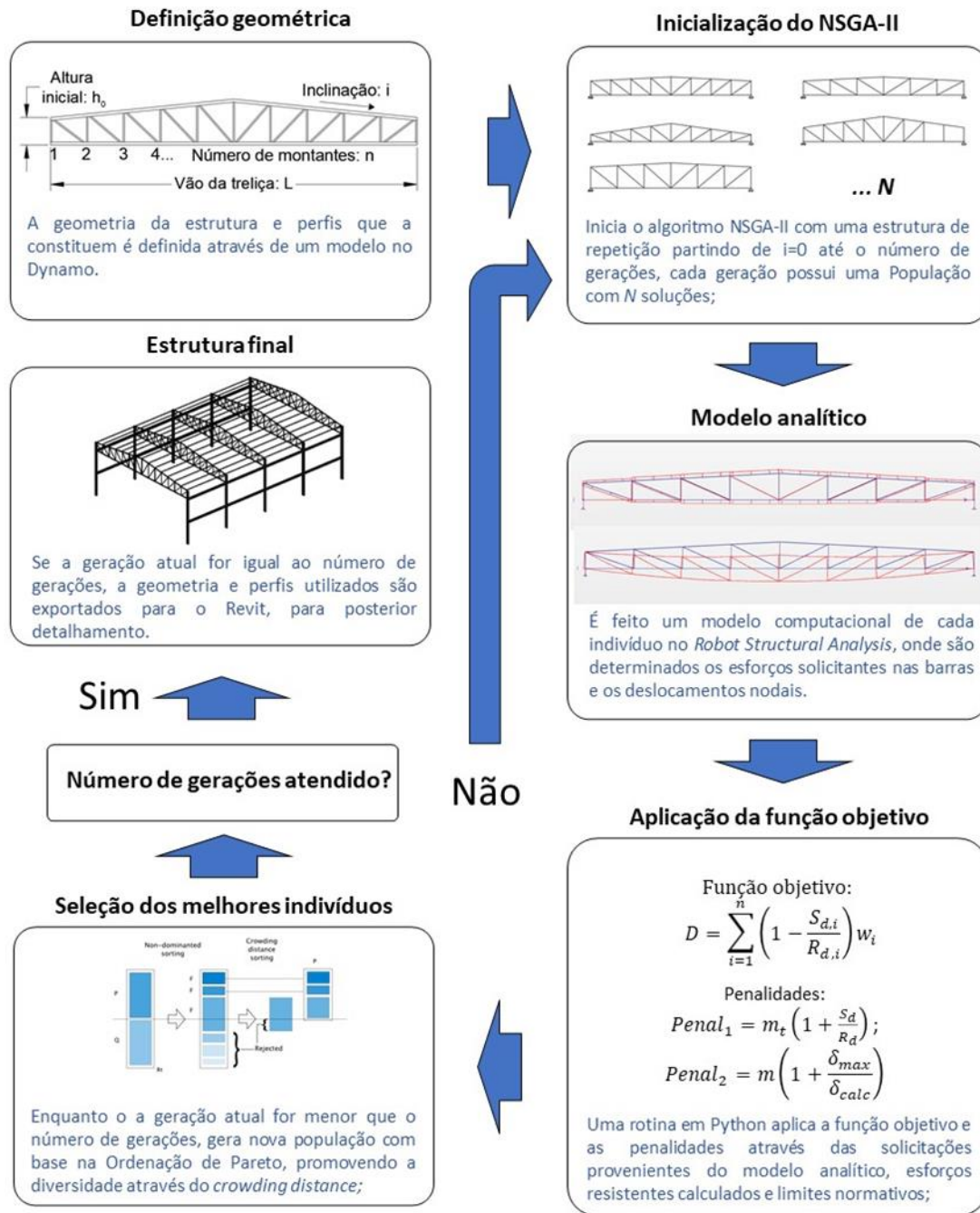


Figura 2. Descrição do problema.

2 Problemas de otimização

A primeira etapa em AGs consiste na criação de um grupo de soluções aleatórias denominado População, Estas soluções são testadas pela função objetivo, sendo avaliado o desempenho de cada uma. Sobre esta população inicial são aplicados os operadores genéticos, que emulam mecanismos de evolução presentes na natureza: (i) *crossover* que consiste na troca de carga genética entre os indivíduos e (ii) *mutação*, que altera aleatoriamente o código genético das soluções. A aplicação destes operadores resulta em uma nova População. Como as soluções

de maior desempenho tendem a transmitir seu código para as gerações seguintes, Populações sucessivas tendem a apresentar desempenho médio superior àquelas que as precederam. A Figura 3 apresenta um fluxograma básico do funcionamento de um AG.

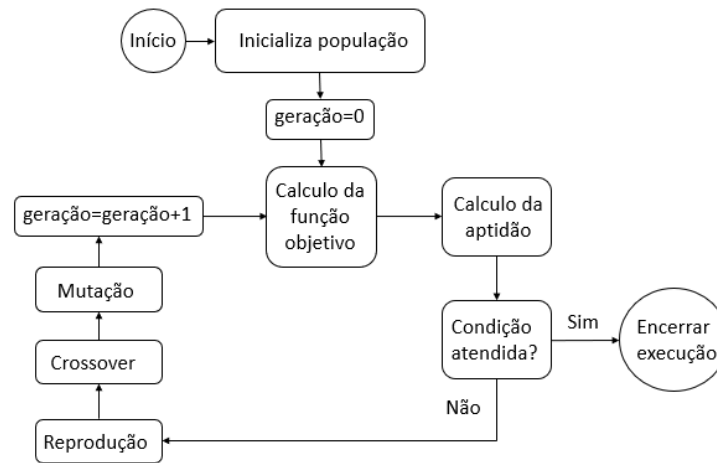


Figura 3. Fluxograma de um AG, adaptado de Deb [4].

O algoritmo escolhido para o desenvolvimento do presente trabalho foi o *Non-dominated Sorting Genetic Algorithm II* (NSGA-II), que é uma estratégia de otimização multiobjetivo amplamente utilizado. O modelo foi proposto por Deb et al. [5] baseado numa versão anterior do mesmo algoritmo (NSGA). A alteração consiste, de maneira resumida, na ordenação das soluções em cada iteração, da inclusão do elitismo na ordenação e, por fim, inclusão de um parâmetro para a preservação da diversidade da população (Deb et al. [5]).

No NSGA-II, os indivíduos (ou soluções para o problema) são ranqueados através do conceito de não-dominância e densidade de soluções apresentadas. O conceito de não-dominância tem por princípio que: se uma solução x_1 não é menos eficiente que uma solução x_2 em todos os objetivos e, concomitantemente, a solução x_1 é estritamente mais eficiente que x_2 em ao menos um objetivo, dizemos que x_1 domina x_2 ou que x_1 é não dominada por x_2 (Deb [4]). De forma a exemplificar temos a função $f(x)$, representada na Fig. 4, na qual há dois objetivos que se pretende. Adicionalmente, nos quadrantes restantes são demarcadas regiões onde não se pode estabelecer relação de não dominância com $f(x)$.

Dado um conjunto de soluções, pode-se comparar pares de soluções e classificá-las. Nesta classificação, haverá um conjunto distinto de soluções que não são dominadas pelas demais, este grupo contém soluções ótimas que se agrupam em uma fronteira, a qual denominamos Pareto-ótimo ou Fronteira de Pareto. Como pode-se deduzir, para problemas com infinitas soluções a fronteira de Pareto é contínua, entretanto, é conveniente tratá-la como um conjunto discreto de soluções, sobretudo nos algoritmos genéticos, que comparam soluções com o intuito de selecioná-las. Na Figura 4 é apresentado um conjunto de soluções de um problema hipotético, às quais destacam-se as soluções contidas no Pareto.

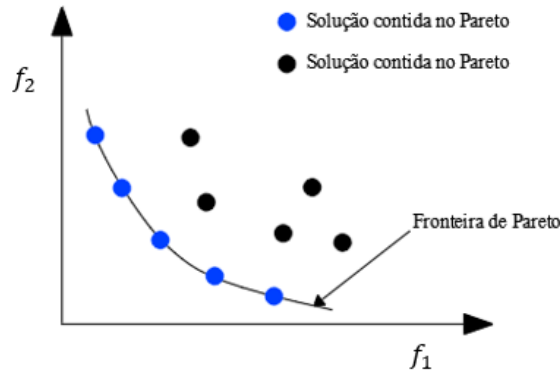


Figura 4. Representação da Fronteira de Pareto. Adaptado de Deantoni e Francato [6].

Há de se observar também que, quando há apenas um objetivo a ser analisado, o Pareto ótimo se torna uma reta na qual os indivíduos se aproximam da solução ótima sem transpor a restrição imposta. Ainda que o NSGA-II tenha por propósito a otimização multi-objetivo, esta análise em apenas um objetivo é viável e foi empregada no presente trabalho.

O algoritmo é iniciado com a criação randômica de um conjunto de soluções denominado população (P), nesta etapa também se inicia o contador de iterações do algoritmo. Cada indivíduo p é avaliado de acordo com sua não-dominância, ou seja, a característica de não ser dominado por outros indivíduos em um dado objetivo. São criados conjuntos S_p onde são alocados os indivíduos de acordo com sua não-dominância, sendo os conjuntos classificados com o número n_p , sendo este valor a quantidade de indivíduos em P que dominam os indivíduos alocados no grupo. Desta forma, os conjuntos são classificados, com o conjunto onde $n_p=0$ recebendo o $rank=1$, para $n_p=1$ temos $rank=2$ e assim sucessivamente. Adicionalmente, é criado um parâmetro denominado *crowding distance*, que mede a distância de cada indivíduo para seus vizinhos, em outras palavras, a densidade de soluções apresentadas. Por fim, os indivíduos são submetidos à seleção através de torneios, onde sua aptidão é medida através de sua posição no ranking e sua densidade (*crowding distance*), onde a maior diversidade é privilegiada. Após a seleção dos melhores indivíduos, são aplicados a estes os operadores genéticos que definirão uma nova população. Tal operação se repetirá, com o acréscimo do contador de iterações, até que se atinja o número de iterações desejado. Na Figura 5 é apresentado um pseudocódigo do algoritmo NSGA-II.

Pseudo código NSGA-II

```

1: Inicialize População
2: Gere N soluções aleatórias e insira em População
3: para i=1 até o número máximo de gerações faça
4:   Gere PopulaçãoProle de tamanho N
5:   Selecione os Pais de População
6:   Crie a Prole de Pais
7:   Aplique mutação em Prole
8:   Combine População e PopulaçãoProle em PopulaçãoAtual
9:   para todos indivíduos em PopulaçãoAtual faça
10:     Atribua rank baseado no Pareto do FastNonDominatedSort
11:   fim para
12:   Gere conjunto das soluções não-dominadas no Pareto
13:   Calcule o CrowdingDistance
14:   Selecione as N melhores soluções começando no melhor front
15: fim para
16: Retorne os resultados

```

Figura 5 – Pseudocódigo do NSGA-II. Adaptado de Lacerda et al. [7].

3 Estudo de caso

Com o objetivo de simplificar o problema, foram adotados apenas perfis tubulares, visto que as estruturas treliçadas industriais consistem em composições entre diferentes perfis e sua aplicação fugiria dos aspectos estudados neste trabalho. A estrutura consiste em uma treliça plana de duas águas, que sustenta a cobertura através de terças apoiadas em seus montantes. A geometria da estrutura é definida através dos seguintes parâmetros: vão da treliça, L ; menor altura da treliça, h_0 ; número de montantes, n ; e inclinação das águas da cobertura, i . Os parâmetros da estrutura são apresentados na Fig. 6.

Ainda que a tecnologia BIM permita a análise de estruturas espaciais, optou-se por uma estrutura na qual o comportamento é amplamente conhecido para permitir melhor avaliação dos resultados.

São também parâmetros da estrutura os perfis empregados nos banzos, diagonais e montantes e também a distância entre treliças

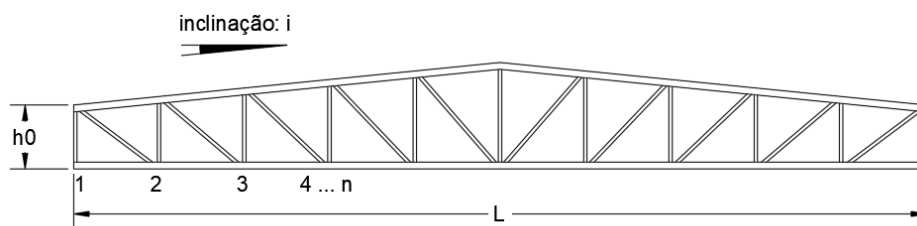


Figura 6. Parâmetros da estrutura.

Neste estudo, foram adotados apenas seções retangulares ou quadradas. Nas Tabelas 1 e 2 são resumidas as seções disponíveis como entrada de dados do código implementado no programa Dynamo. Foi considerado o emprego de perfis de aço de tensão de escoamento $f_y=350$ MPa e módulo de elasticidade $E=200$ GPa.

Tabela 1. Perfis tubulares quadrados utilizados nas diagonais e montantes.

| b [mm] | t [mm] | masa [kg/m] | Area [cm ²] |
|--------|--------|-------------|-------------------------|
| 51 | 3,2 | 4,52 | 5,80 |
| 64 | 3,2 | 5,77 | 7,41 |
| 64 | 4,8 | 8,27 | 10,6 |
| 76 | 3,2 | 7,04 | 9,03 |
| 89 | 3,2 | 8,23 | 10,6 |

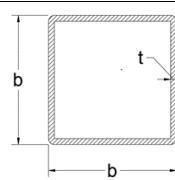
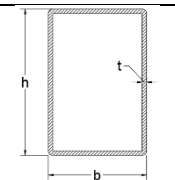


Tabela 2. Perfis tubulares retangulares utilizados nos banzos.

| hxb [mm] | t [mm] | masa [kg/m] | Area [cm ²] |
|----------|--------|-------------|-------------------------|
| 51x25 | 3,2 | 3,26 | 4,18 |
| 76x51 | 3,8 | 6,80 | 8,72 |
| 89x64 | 3,2 | 7,04 | 9,03 |
| 89x64 | 4,8 | 12,09 | 15,5 |
| 102x51 | 3,2 | 7,04 | 9,03 |



As ações na treliça são peso próprio da treliça, peso próprio da telha metálica tipo ondulada de 0,80 mm de espessura de 100 N/m² conforme NBR 6120:2019 [8], sobrecarga de 0,25 kN/m² de acordo com NBR8800:2018 [9] e ação de vento conforme NBR6123:1988 [10].

Conforme NBR8800:2008, são avaliadas três (3) combinações últimas normais das ações e uma combinação quase-permanente para o Estado Limite de serviço, que são resumidas na Tabela 3 em relação aos valores característicos com fatores de ponderação e combinações já consideradas.

Tabela 3. Resumo das combinações avaliadas.

| Identificação | Combinação |
|---------------|--------------------------|
| 1 | 1,25 PP + 1,5 SC |
| 2 | PP + 1,4 V ₀ |
| 3 | PP + 1,4 V ₉₀ |
| 4 | PP + 0,5 SC |

3.1 Função objetivo para minimização da massa

Podem ser adotadas diferentes abordagens na busca por otimizar uma estrutura, para o presente estudo adotou-se a minimização da massa da estrutura como função-objetivo que define o problema. Desta forma, o algoritmo NSGA-II irá classificar e privilegiar os indivíduos que apresentarem menor massa. A função-objetivo foi modelada tendo por base o desperdício de cada elemento de barra componente da estrutura, onde este desperdício é compreendido como a quantidade de material excedente na estrutura ao analisar estados limites últimos apenas, ou seja, a razão entre os esforços solicitantes e os esforços resistentes. A resistência dos elementos é determinada conforme NBR8800:2008 [9] considerando limite de escoamento com redução de magnitude caso ocorra flambagem global e/ou flambagem local, sendo o fator de redução associado à flambagem global calculado conforme a norma NBR16239:2013 [11]. Complementarmente, é imposta uma penalidade para satisfazer o estado limite de serviço de deformações excessivas.

Desta forma, temos a função de desperdício

$$D_i = \left(1 - \frac{S_{d,i}}{R_{d,i}}\right) w_i. \quad (1)$$

onde D_i é o desperdício do elemento de barra analisado; $S_{d,i}$ é o esforço solicitante no elemento de barra analisado; $R_{d,i}$ é o esforço resistente no elemento de barra analisado; w_i é a massa do elemento de barra analisado. O desperdício total da estrutura D é dado por

$$D = \sum_{i=1}^n \left(1 - \frac{S_{d,i}}{R_{d,i}}\right) w_i. \quad (2)$$

sendo n o número total de barras que compõe a estrutura. Para satisfazer a condição de segurança, os esforços solicitantes não podem superar os esforços resistentes, caso contrário ocorre a falha da estrutura. Quando há barras onde esta condição de segurança não é atendida, tem-se as seguintes penalidades.

$$P_1 = \sum_{i=1}^n w_i \left(1 + \frac{S_d}{R_d}\right) \quad (3)$$

$$P_2 = \sum_{i=1}^n w_i \left(\frac{\delta_d}{\delta_{lim}}\right) \quad (4)$$

P_1 e P_2 são penalidades associadas, respectivamente, ao estado limite último e estado limite de serviço de deformações excessivas ambos previstos em norma. δ_d é o deslocamento calculado e δ_{lim} é o deslocamento máximo estabelecido pela norma NBR8800:2008 [9].

3.2 Aplicação do algoritmo genético

A aplicação do algoritmo genético neste trabalho foi realizada através do pacote *Optimo*, de livre acesso na biblioteca do Dynamo [3]. A implementação deste algoritmo inicia-se com a definição dos valores máximo e mínimo para cada parâmetro sujeito à otimização e a quantidade de indivíduos da população. Estes valores são os dados de entrada de uma função que gera a população inicial atribuindo valores aleatórios para cada parâmetro do

problema (Fig. 7). Na primeira iteração, também é gerada uma lista de valores nulos, que guarda a posição na qual serão armazenadas as soluções.

A partir da população inicial, a geometria da estrutura é definida pelo modelo paramétrico implementado no Dynamo. Através do pacote Structural Analysis são atribuídos os elementos do modelo estrutural, como os perfis utilizados, tipos de apoios e casos de combinações de carregamentos conforme NBR8800:2008 [9]. Uma função exporta os dados fornecidos para o programa Robot Structural Analysis, que faz a análise da estrutura com comportamento linear-elástico do material, e tem como valores de retorno os esforços solicitantes em cada barra e a flecha para verificar o Estado Limite de Serviço.

A determinação dos esforços resistentes dos elementos é feita através de uma rotina em Python, na qual as barras são dimensionadas para o Estado-Limite Último. Os esforços resistentes calculados, bem como os resultados da análise obtidos (como esforços solicitantes e deslocamento para o Estado-Limite de Serviço), são dados de entrada de outra rotina em Python que calcula a função objetivo conforme as eq. (1) e (2) e penalidades conforme as eq. (3) e (4).

Os resultados obtidos para esta função objetivo são atribuídos à população (sendo estes os Pais), então é aplicada uma nova função para reincorporar estes valores à rotina do pacote *Optimo*. Então uma função recebe a lista de soluções e gera a *Prole* utilizando algoritmo genético. Uma função é novamente empregada para atribuir a *Prole* à população. Por fim é aplicado o conceito da Fronteira de Pareto [5] para ordenar os resultados que formarão a próxima população.

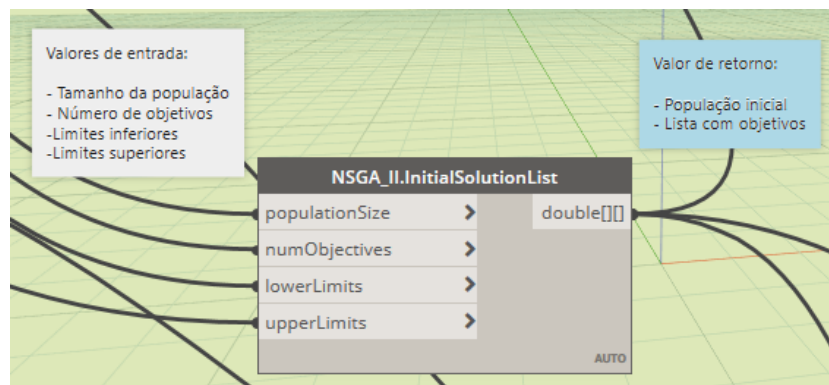


Figura 7 – Função que cria a população inicial.

3.3 RESULTADOS

Com base na Fig.6, foram feitas análises para o vão, L , de 16,00 metros e adotou um espaçamento entre treliças de 5,00 metros. Os perfis utilizados para compor os elementos da treliça estão nas Tab.1 e 2.

3.3.1 Análise através de estrutura parametrizada e otimização

A análise foi feita com os seguintes parâmetros mínimos e máximos, respectivamente, altura da treliça, h_0 , 0,30 metros e 1,10 metros; inclinação, i , 5° e 15° ; número de montantes, n , 7 e 15, sendo adotados apenas números ímpares. A Figura 8apresenta a evolução do peso total e do desperdício conforme novos indivíduos são gerados.

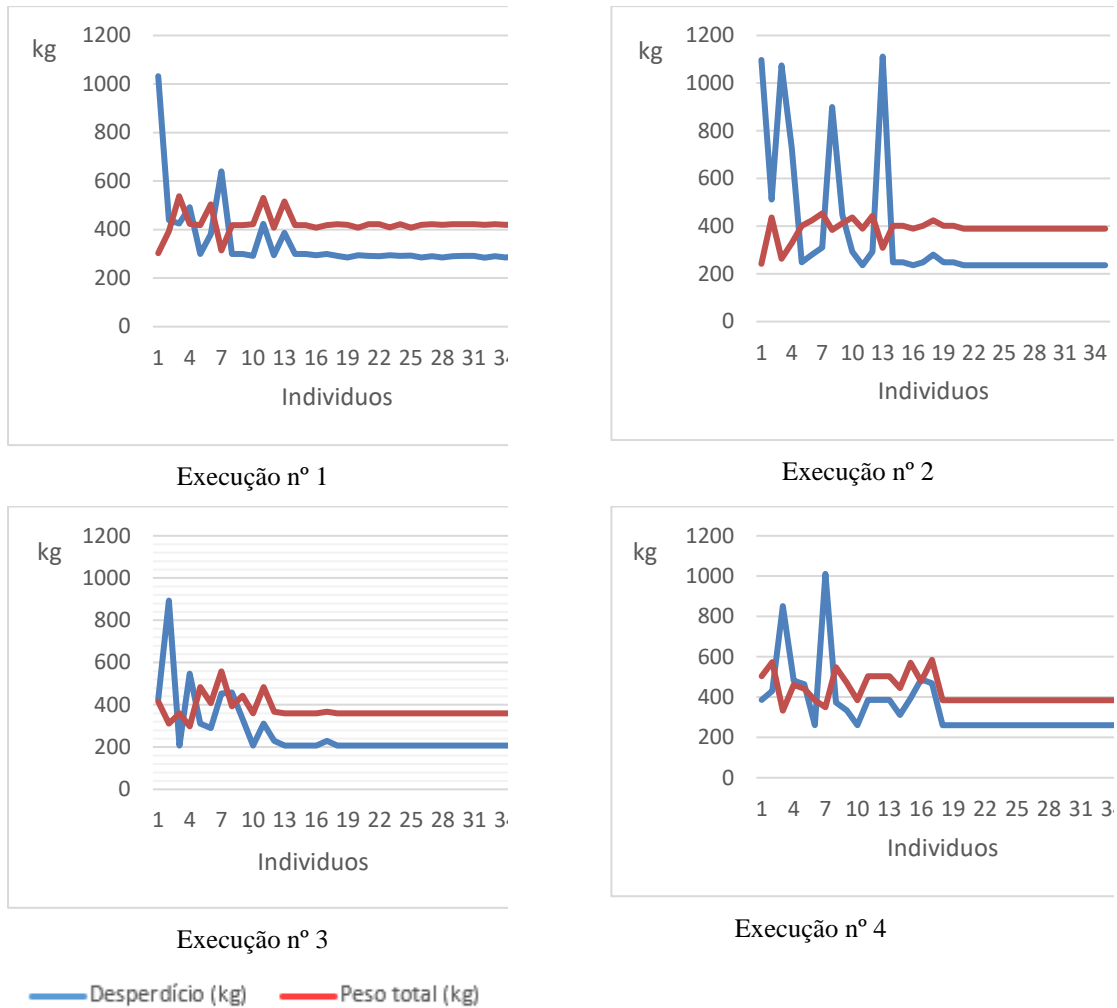


Figura 8. Gráficos de Desperdício e Peso Total x Individuos para quatro execuções da rotina.

A treliça mais leve foi obtida através da execução nº4 e tem os seguintes parâmetros: $h_0 = 0,624 m$, $n = 7$, $i = 10^\circ$, perfis (89x64x3,2) para os banzos, (51x51x3,2) para os montantes e (64x64x3,2) para as diagonais. Foram obtidos os seguintes resultados:

- Massa total da treliça – 359 kg
- Desperdício de material – 207 kg
- Relação Desperdício/Peso - 0,57
- Deslocamento máximo (ELS) – 5,30 mm

As Tabelas 4 e 5 apresentam o resumo dos esforços solicitantes máximos relaciona com respectivas combinações de ação. As Figuras 9 a 12 apresentam os esforços solicitantes para as combinações apresentadas na Tab. 3. Para Figuras 9, 10 e 11, os diagramas em verde são esforços de tração e os diagramas em vermelho são esforços de compressão) e deformação para a combinação do ELS é ilustrada na fig. 12.

Tabela 4. Esforços normais solicitantes de tração e combinações para análise paramétrica

| Elemento | Esforço Solicitante (kN) | Combinação |
|----------------|--------------------------|-------------------|
| Banzo superior | 45,30 | 1: 1,25PP + 1,5SC |

| | | |
|----------------|-------|-------------------|
| Banzo inferior | 54,72 | 2: PP + 1,4V0 |
| Diagonal | 48,64 | 2: PP + 1,4V0 |
| Montante | 10,40 | 1: 1,25PP + 1,5SC |

Tabela 5. Esforços normais solicitantes de tração e combinações para análise paramétrica

| Elemento | Esforço Solicitante (kN) | Combinação |
|----------------|--------------------------|-------------------|
| Banzo superior | -51,49 | 3: PP + 1,4V90 |
| Banzo inferior | -51,49 | 1: 1,25PP + 1,5SC |
| Diagonal | -12,13 | 2: PP + 1,4V0 |
| Montante | -12,00 | 3: PP + 1,4V90 |

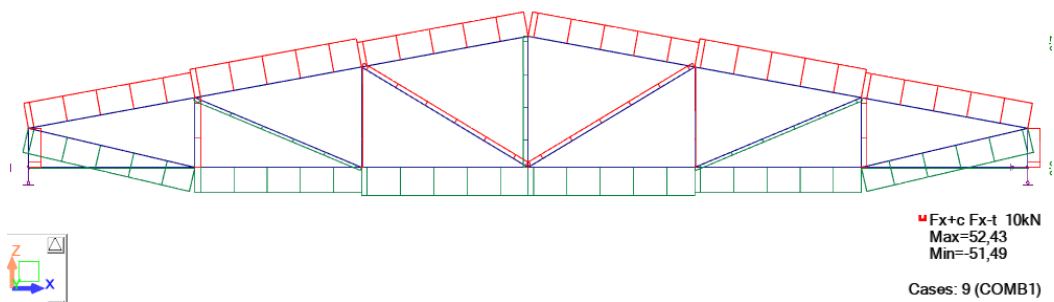


Figura 9. Diagrama de esforços normais para Combinação 1, para análise paramétrica.

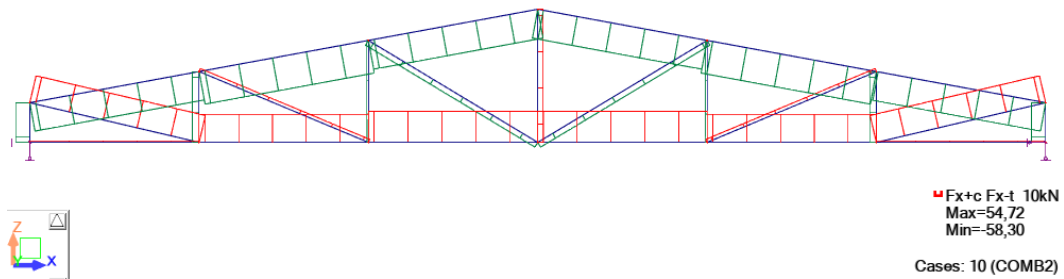


Figura 10. Diagrama de esforços normais para Combinação 2, para análise paramétrica.

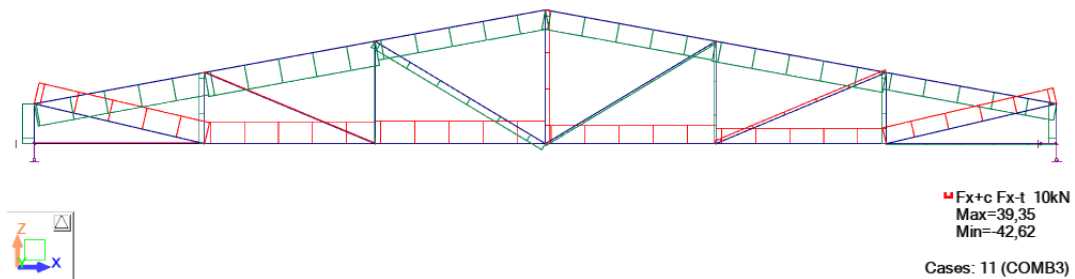


Figura 11. Diagrama de esforços normais para Combinação 3, para análise paramétrica.

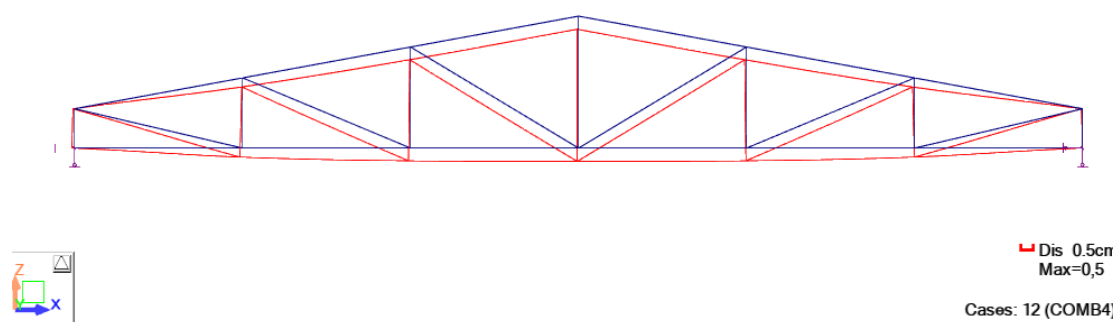


Figura 12. Estrutura deformada para Combinação 4, para análise paramétrica.

Através destes resultados foi possível observar que em todas as execuções houve convergência dos resultados. Também sendo possível observar situações onde os indivíduos foram penalizados, ocasionando em elevado desperdício, como era esperado com a aplicação das penalidades. Ao final, a estrutura obtida de menor massa possuía um aproveitamento de aço próximo aos 60%, sendo um valor satisfatório, considerando-se que não foram adotados grupos de elementos diferenciados para regiões onde as solicitações são reconhecidamente baixas.

3.3.2 Análise de estrutura através de métodos convencionais

Para comparar os resultados obtidos neste estudo, foi realizado um dimensionamento conforme Rebello [12], que recomenda razão (h/L) entre 0,10 e 0,20 para treliças do tipo Pratt, sendo h a maior altura da treliça. Desta forma a altura da treliça estaria confinada entre os valores $h = 1,60 m$ e $h = 3,20 m$. Optou-se por adotar o valor intermediário $h = 2,40 m$. A inclinação adotada foi de 10° , resultando em uma altura inicial h_0 de $1,00 m$.

Adotando-se os perfis apresentados nas tabelas 1 e 2 para o dimensionamento, foi adotada para os banzos a seção (76x51x3,8) e para as diagonais e montantes a seção (51x51x3,2). As tabelas 6 e 7 apresentam os esforços solicitantes e resistentes máximos e suas respectivas combinações. As Figuras 13 a 16 apresentam os diagramas de esforços normais para as combinações de ELU (em verde esforços de tração e em vermelho esforços de compressão) e a estrutura deformada para a combinação do ELS, conforme apresentado previamente na Tabela 3.

- Massa total da treliça – 432 kg
- Deslocamento máximo (ELS) – 3,60 mm

Tabela 6. Esforços normais solicitantes e resistentes de tração e combinações para análise convencional.

| Elemento | Esforço Resistente [kN] | Esforço Solicitante [kN] | Solicitante/ Resistente | Combinação |
|----------------|-------------------------|--------------------------|-------------------------|-------------------|
| Banzo superior | 277,45 | 40,09 | 0,145 | 1: 1,25PP + 1,5SC |
| Banzo inferior | 277,45 | 44,07 | 0,157 | 2: PP + 1,4V0 |
| Diagonal | 184,54 | 30,73 | 0,166 | 2: PP + 1,4V0 |
| Montante | 184,54 | 17,19 | 0,093 | 1: 1,25PP + 1,5SC |

Tabela 7. Esforços normais solicitantes e resistentes de compressão e combinações para análise convencional

| Elemento | Esforço Resistente [kN] | Esforço Solicitante [kN] | Solicitante/ Resistente | Combinação |
|----------------|-------------------------|--------------------------|-------------------------|-------------------|
| Banzo superior | -226,5 | -45,75 | 0,197 | 1: 1,25PP + 1,5SC |
| Banzo inferior | -226,5 | -42,41 | 0,187 | 2: PP + 1,4VO |
| Diagonal | -114,25 | -28,76 | 0,199 | 2: PP + 1,4VO |
| Montante | -153,63 | -18,75 | 0,122 | 1: 1,25PP + 1,5SC |

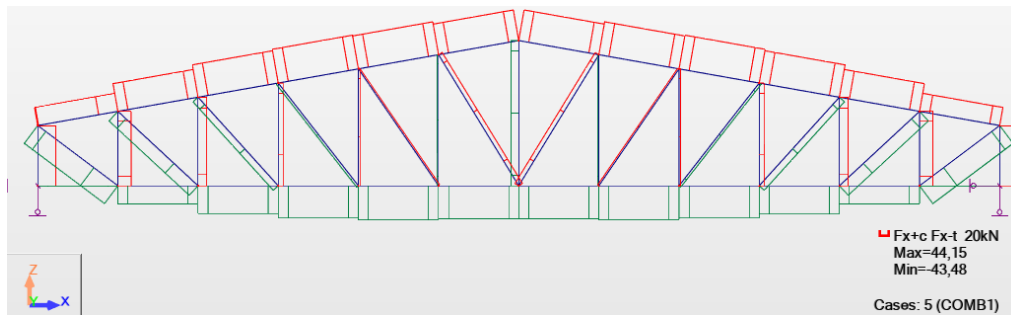


Figura 13. Diagrama de esforços normais para Combinação 1, para análise convencional.

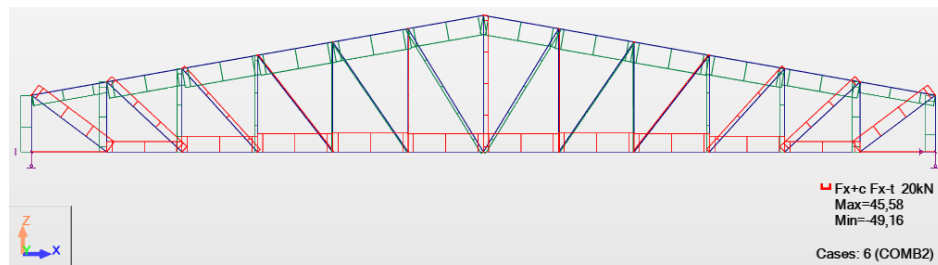


Figura 14. Diagrama de esforços normais para Combinação 2, para análise convencional.

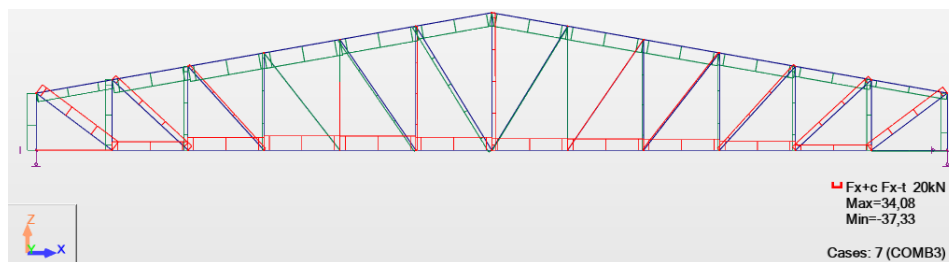


Figura 15. Diagrama de esforços normais para Combinação 3, para análise convencional.

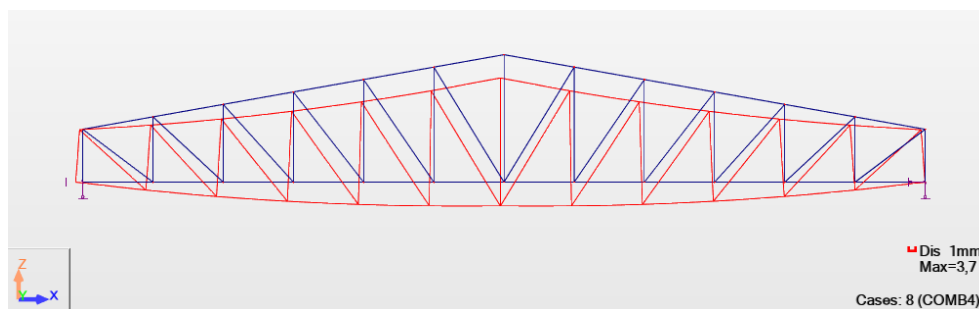


Figura 16. Estrutura deformada para Combinação 4, para análise convencional.

Observa-se neste trabalho que, em oposição ao estudo com otimização, a análise convencional pode apresentar resultados com menor aproveitamento dos elementos. Ainda que em um projeto real possam ser adotadas variações para buscar melhores soluções, o processo de redefinição dos parâmetros da estrutura e modelagem da mesma tomaria tempo do projetista. Este trabalho teve como proposta a análise de uma estrutura em que o projetista tem clareza da distribuição de esforços na mesma e consequências nesta distribuição caso ocorram

4 Conclusão

Este estudo teve como principal objetivo avaliar o emprego da modelagem paramétrica de estruturas aliada à otimização de estruturas por meio de algoritmos genéticos, tendo como principal ponto de partida o emprego desta técnica em um ambiente BIM. Para tal, foi utilizado o programa Dynamo e o programa de cálculo estrutural Robot Structural Analysis, ambos distribuídos pela Autodesk, e com adição de rotinas em Python.

Através do estudo realizado foi possível observar diversos empecilhos referentes à execução do método proposto, sobretudo referente à comunicação entre os diferentes programas, o que produziu a necessidade da utilização de “pacotes” desenvolvidos por terceiros e que não se teve acesso ao código fonte. Esta comunicação deficitária provocou um aumento do custo computacional conforme cada estrutura era calculada, impossibilitando o cálculo para uma grande quantidade de indivíduos. Apesar dos problemas relatados, foi feita uma análise dentro do que as limitações permitiram. Nesta análise foi possível observar a convergência dos resultados para soluções de menor desperdício, o que possivelmente resultaria em soluções otimizadas caso fosse estendido para um número significativo de indivíduos por população e gerações.

Para futuros trabalhos, sugere-se analisar a aplicação das mesmas técnicas aqui propostas em diversos programas disponíveis comercialmente, preferencialmente em ambiente BIM, com o objetivo de determinar a aplicabilidade da mesma e possíveis empecilhos.

Outra possível abordagem é a integração entre ferramentas já existentes em programas de cálculo estrutural para a otimização, como por exemplo a ferramenta de verificação de estruturas “*Steel Members Design*” do Robot Structural Analysis e algoritmos de otimização.

Por fim, as técnicas aqui apresentadas podem ainda ser analisadas frente a análises de maior complexidade, a citar os efeitos de segunda ordem ou comportamento dinâmico da estrutura. Tendo em vista que tais análises por si já podem representar um alto custo computacional, a depender da estrutura analisada, seria de grande interesse compreender a viabilidade das mesmas através de algoritmos que requerem repetição de resultados.

Declaração de autoria. Os autores confirmam que são os únicos responsáveis pela autoria deste trabalho, e que todo o material que foi incluído como parte do presente trabalho é propriedade (e autoria) dos autores, ou tem a permissão dos proprietários para ser incluído aqui.

References

- [1] K. Kourus, M. Salamak e M. Jasiński. *Optimization of geometric parameters of arch bridges using visual programming FEM components and genetic algorithm*. Engineering Structures, v. 241, Mar, 2021.

- [2] S. Kravanja et al. *Optimal design of single-story steel building structures based on parametric MINLP optimization*. *Journal of Constructional Steel Research*, v. 81, p. 86–103, 2013.
- [3] D. Vermeulen, *Dynam(o)ite Your Design for Engineers*. Disponível em: <<https://www.autodesk.com/autodesk-university/class/Dynamoite-Your-Design-Engineers-2015>>. Acesso em: 5 jul. 2021.
- [4] K. Deb. *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, Ltd., 2001. v. 1
- [5] K. Deb, et al. *A fast and elitist multiobjective genetic algorithm: NSGA-II*. *IEEE Transactions on Evolutionary Computation*, v. 6, n. 2, p. 182–197, 2002.
- [6] V. B. Deantoni e A. L. Francato, *Contribuição aos processos de tomada de decisão no setor de saneamento com a aplicação da Teoria dos Jogos (TJ)*. *Revista DAE*, v. 67, n. 219, p. 41–54, 2019.
- [7] H. Lacerda e A. R. S. Feitosa and W. P. Santos and F. R. Cordeiro. *Redução de Energia em Casas Inteligentes utilizando uma Abordagem Multiobjetivo baseada nos Perfis de Uso de Equipamentos*. Jul 2016.
- [8] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, (2019). ABNT NBR 6120 – *Ações para o cálculo de estruturas de edificações*. Rio de Janeiro, Brasil.
- [9] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, (2008). ABNT NBR 8800 – *Projeto de estruturas de aço e de estruturas mistas de aço e concreto de edifícios*. Rio de Janeiro, Brasil.
- [10] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, (1988). ABNT NBR 6123 – *Forças Devidas ao Vento em Edificações*. Rio de Janeiro, Brasil.
- [11] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, (2013). ABNT NBR 16239 – *Projeto de estruturas de aço e de estruturas mistas de aço e concreto de edificações com perfis tubulares*. Rio de Janeiro, Brasil.
- [12] Y. Rebello. *A concepção estrutural e a arquitetura*. Editora Zigurate, 2000.