

**UNIVERSIDADE DO VALE DO RIO DOS SINOS - UNISINOS
UNIDADE ACADÊMICA DE GRADUAÇÃO
CURSO DE ENGENHARIA CIVIL**

NICHOLAS ORTIZ PORTO

**ANÁLISE DO USO DE DESIGN GENERATIVO NAS FASES INICIAIS DO PROJETO
ARQUITETÔNICO**

São Leopoldo

2021

NICHOLAS ORTIZ PORTO

**ANÁLISE DO USO DE DESIGN GENERATIVO NAS FASES INICIAIS DO PROJETO
ARQUITETÔNICO**

Trabalho de Conclusão de Curso
apresentado como requisito parcial para
obtenção do título de Bacharel em
Engenharia Civil, pelo Curso de
Engenharia Civil da Universidade do Vale
do Rio dos Sinos - UNISINOS

Orientador: Prof. Dra. Andrea Parisi Kern

São Leopoldo

2021

AGRADECIMENTOS

Agradeço em totalidade o apoio da minha família, em especial aos meus pais, Aramis Dossena Porto e Eliara Ortiz Porto, e minha irmã, Nichole Ortiz Porto, por todo o apoio dado durante a minha formação pessoal e profissional, por serem conselheiros em momentos de dúvida e a minha inspiração.

Á minha namorada, Márcia Cunha dos Santos, por ser a minha companhia durante toda essa etapa e por sempre me motivar.

Á minha orientadora, Prof. Dra. Andrea Parisi Kern, por me acompanhar e orientar durante toda a elaboração deste trabalho.

A todos os professores do curso de Engenharia Civil, que foram de suma importância para minha formação acadêmica.

Aos colegas e amigos, por sempre me apoiarem durante a graduação.

RESUMO

Na etapa de estudo de concepção, o projeto arquitetônico tem como objetivo dar forma ao conceito desejado pelo projetista responsável e verificar como esse conceito responde à aplicação de diversos requisitos de projeto. No entanto, os prazos reduzidos para desenvolvimento dos projetos, impossibilita que diferentes tipos de ideias sejam explorados de uma forma mais profunda. Devido a essa necessidade, surge o método de design generativo, utilizando conhecimentos de outras áreas, como a ciência da computação, o design generativo usa técnica evolutiva para otimizar soluções de projeto de maneira iterativa, de forma, a melhorar o processo de interação homem-máquina, deixando para máquina, a responsabilidade de encontrar as melhores soluções para o problema aplicado e o projetista, responsável pelo processo criativo e refino dos resultados produzidos. Assim, esse projeto apresenta uma análise sobre a aplicação do método de design generativo visando à criação do projeto arquitetônico na etapa de estudo de concepção. Os resultados obtidos evidenciam que é possível aplicar o método de design generativo para gerar projetos arquitetônicos otimizados, visto que, em um período de 2h55, foram produzidos 120 soluções, ao longo de 40 gerações, otimizando simultaneamente quatro parâmetros de projeto. Por outro lado, ainda há algumas ressalvas sobre seu uso, principalmente, no que tange os conhecimentos necessários para aplicação do método, ainda muito dependente de programação, e na questão de produtividade das rotinas quando aplicado a um conjunto diversificado de casos reais.

Palavras-chave: BIM. Design Generativo. Projeto Arquitetônico.

LISTA DE FIGURAS

Figura 1 - Curva de MacLeamy	14
Figura 2 - Frente ótima de Pareto	18
Figura 3 - Etapas do algoritmo	20
Figura 4 - Processo de ordenação da população.....	21
Figura 5 - Processo de agrupamento em frentes	22
Figura 6 - Distância de multidão.....	23
Figura 7 - Etapa inicial do algoritmo	23
Figura 8 - Agrupamento nos picos de otimização	24
Figura 9 - Processo de união das características.....	25
Figura 10 - Processo de reprodução por mistura	25
Figura 11 - Exemplo de programação visual.....	27
Figura 12 - Diferentes taxas de ocupação no mesmo terreno.....	29
Figura 13 - Mudança do índice de aproveitamento relacionado à taxa de ocupação.....	29
Figura 14 - Curva do índice de compacidade pelo custo da construção	34
Figura 15 - Estrutura da rotina.....	40
Figura 16 - Performance com diferentes tamanhos de população	44
Figura 17 - Configuração no Dynamo	44
Figura 18 - Configuração no Grasshopper	45
Figura 19 - Nova estrutura da rotina.....	47
Figura 20 - Rotina e resultados Grasshopper.....	50
Figura 21 - Rotina e resultados Dynamo	51
Figura 22 - Base do terreno.....	52
Figura 23 - Terreno importado no software de programação visual	52
Figura 24 - TO e IA utilizados para construção da rotina	53
Figura 25 - Lógica para posição central da torre	54
Figura 26 - Rotina Dynamo	57
Figura 27 - Rotina Grasshopper	57
Figura 28 - Resultado do custo otimizado	59
Figura 29 - Resultado da TO otimizada.....	59
Figura 30 - Resultados gerados no Dynamo	60
Figura 31 - Soluções escolhidas	60
Figura 32 - Pontos para criação da geometria do pavimento base	63

Figura 33 - Superfície total	64
Figura 34 - Criação do contorno do pavimento base.....	65
Figura 35 - Pavimento base	66
Figura 36 - Resultado do sólido do prédio.....	67
Figura 37 - Pontos de análise solar.....	68
Figura 38 - Rotina final desenvolvida no Dynamo	69
Figura 39 - Resultados da rotina final.....	70
Figura 40 - Problemas no resultado gerado	73

LISTA DE QUADROS

Quadro 1 - Participação dos itens no custo total	32
Quadro 2 - Resumo das etapas	37
Quadro 3 - Comparativo entre as soluções escolhidas	61
Quadro 4 - Resumo dos resultados.....	71

LISTA DE SIGLAS

AIA	American Institute of Architects
CAD	Computer-aided Design
NSGA-II	Non-dominated Sorting Genetic Algorithm II
TO	Taxa de ocupação
IA	Índice de Aproveitamento
CUB	Custo Unitário Básico
SINAPI	Sistema Nacional de Pesquisa de Custos e Índices da Construção Civil
TCPO	Tabela de Composições de Preços para Orçamentos
ABNT	Associação Brasileira de Normas Técnicas
IC	Índice de Compacidade
IeC	Índice Econômico de Compacidade

SUMÁRIO

1 INTRODUÇÃO	10
1.1 OBJETIVOS	11
1.1.1 Objetivo Geral	11
1.1.2 Objetivos Específicos	11
1.2 JUSTIFICATIVA	12
2 FUNDAMENTAÇÃO TEÓRICA	13
2.1 ESTUDO DE CONCEPÇÃO	13
2.2 DESIGN COMPUTACIONAL	14
2.2.1 Design Paramétrico	15
2.2.2 Design Algorítmico	15
2.3 DESIGN GENERATIVO	16
2.3.1 Definição	16
2.3.2 Otimização Multiobjetivo	17
2.3.3 Algoritmo Genético	18
2.3.3.1 <i>Non-dominated Sorting Genetic Algorithm II - NSGA-II</i>	20
2.3.3.2 Galapagos	23
2.3.4 Programação Visual	26
2.4 PARÂMETROS DE PROJETO	28
2.4.1 Parâmetros legais	28
2.4.1.1 Taxa de ocupação	28
2.4.1.2 Índice de aproveitamento	29
2.4.2 Parâmetros de custo	30
2.4.4 Incidência Solar	34
3 METODOLOGIA	36
3.1 DELINEAMENTO DO TRABALHO	36
3.2 ETAPAS DE PESQUISA	38
3.2.1 Validação dos softwares	38
3.2.2 Criação das rotinas	39
3.2.2.1 Parâmetros de projeto	40
3.2.2.1.1 <i>Taxa de ocupação</i>	40
3.2.2.1.2 <i>Índice de aproveitamento</i>	41
3.2.2.1.3 <i>Custo</i>	42

3.2.2.2 Construção da programação visual	43
3.2.2.3 Configuração do algoritmo genético	43
3.2.3 Comparação entre os softwares	45
3.2.4 Incrementação de variáveis.....	46
3.2.4.1 Índice econômico de compacidade	47
3.2.4.2 Incidência solar	47
3.2.4.3 Modificações na rotina.....	48
3.2.5 Análise de resultados	48
4 RESULTADOS.....	49
4.1 VALIDAÇÃO DOS SOFTWARES	49
4.2 CONSTRUÇÃO DA ROTINA	51
4.2.1 Terreno	51
4.2.2 Posição da torre	53
4.2.3 Criação dos pavimentos	54
4.2.4 Custo	55
4.2.5 Rotinas	56
4.3 COMPARAÇÃO ENTRE OS SOFTWARES.....	57
4.4 INCREMENTAÇÃO DA ROTINA NO DYNAMO	62
4.4.1 Criação do pavimento base.....	62
4.4.2 Parâmetros de projeto	67
4.4.3 Rotina final.....	69
4.5 ANÁLISE DOS RESULTADOS	69
4.5.1 Limites de aplicação da metodologia	72
4.5.2 Avaliar a capacidade da aplicação desenvolvida.....	73
5 CONCLUSÃO	75
REFERÊNCIAS.....	77

1 INTRODUÇÃO

O estudo de concepção é uma etapa crucial no processo de desenvolvimento de um projeto, pois é nessa fase que decisões envolvendo o tipo da construção, a viabilidade funcional e econômica são tomadas, determinando o custo, a complexidade da construção e o tempo de entrega que o projeto terá. (EASTMAN et al., 2011).

Essa etapa de projeto, por obrigar o projetista a considerar simultaneamente os requisitos estéticos, funcionais e técnicos, é, usualmente, realizada em um processo cíclico de três etapas: esboço, avaliação e modificação. Para a criação e avaliação, geralmente os métodos utilizados são consultas a projetos de referência, realizando analogias a edifícios existentes ou de forma intuitiva, utilizando-se da experiência e conhecimento do projetista responsável. (EASTMAN et al., 2011; LAWSON, 2005).

Segundo Rohrmann (2019), mesmo tendo a necessidade de elaboração, exploração e avaliação de múltiplas alternativas, os prazos designados para as etapas de projeto são limitados, obrigando os projetistas a restringirem a quantidade de variáveis a serem testadas. Contudo, como aponta Bayer et al. (2017), ainda que haja essa necessidade, a principal ferramenta escolhida entre os projetistas para elaboração dos esboços tem sido o papel e a caneta, limitando assim, o potencial possível de ser alcançado.

O método de design generativo trata-se de uma alternativa, ou um método complementar, aos esboços à mão livre. Refere-se a uma abordagem que resolve problemas baseados em objetivos, dependente de um software de simulação para explorar as interações entre as variáveis e otimizá-las, usualmente, sendo realizado por algoritmos genéticos. (NAGY; VILLAGGI; BENJAMIN, 2018).

Como vantagem, destaca-se a possibilidade de realizar exaustivas modificações nas alternativas de projetos, explorando a criatividade do projetista. Isso se deve à velocidade de processamento dos computadores e ao processo de permutação automática das variáveis, resultando, em um grande número de distintas soluções de projetos arquitetônicos elaboradas. (FISCHER; HERR, 2001).

Entretanto, desenvolver um sistema de design generativo requer um grande esforço inicial, esbarrando em necessidades de programação, da correta compreensão do problema que deseja otimizar, da definição dos objetivos e

restrições através de parâmetros de maneira clara. Isso se deve ao fato de que o computador não tem uma compreensão inerente do problema com que se depara. Ele só pode aplicar as regras ensinadas. Assim sendo, ao não estruturar corretamente o algoritmo, restringi-se os possíveis resultados positivos. (DAVIS, 2013; ROHRMANN, 2019).

Assim, os parâmetros de projeto tornam-se a referência para a qualidade dos resultados, utilizar-se de bons indicadores fará com que o sistema de design generativo possa entender melhor o problema que está sendo otimizado. Ainda, ao ser levados em conta os diversos requisitos que podem ser considerados no estudo de um projeto, há aqueles que são obrigatórios, pois se baseiam no atendimento de legislações, como os parâmetros de taxa de ocupação e índice de aproveitamento adotados por planos diretores no Brasil; há aqueles que regem a viabilidade de um projeto, visto que, necessitam atender critérios que estejam nos idealizados pelo mandante do projeto, como o custo de um empreendimento; e há, ainda, aqueles que buscam atender fundamentos de desempenho e de qualidade para o usuário, como o parâmetro de exposição à luz solar. (ABUSSAMRA, 2007; ESTADÃO, 2019; SAMPAIO, 2004)

Considerando os conceitos apresentados, este trabalho visa contribuir no campo de pesquisa acerca da aplicação de métodos computacionais no setor de arquitetura, engenharia e construção, com a finalidade de avaliação da aplicação do método de design generativo na etapa de estudo de concepção.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Analisar o método de design generativo voltado para a elaboração de projeto arquitetônico na etapa de estudo de concepção.

1.1.2 Objetivos Específicos

Os objetivos específicos são:

- a) comparar as opções de softwares personalizáveis existentes;
- b) identificar os limites de aplicação da metodologia;

- c) avaliar a capacidade da aplicação de responder aos requisitos de um processo de criação de um projeto arquitetônico na fase de estudo de concepção.

1.2 JUSTIFICATIVA

Iordanova et al. (2009) afirmam que os métodos paramétricos de modelagem enriquecem os processos de projeto. Esse avanço se deve ao fato de que os métodos paramétricos auxiliam os projetistas na geração e avaliação de diversas alternativas de projeto arquitetônico.

Contudo, a modelagem paramétrica apenas fornece os diferentes modelos de projeto, nesse processo não há a consideração dos critérios de desempenho, portanto, criou-se uma tendência recente de integrar ao método paramétrico às ferramentas de simulações, visto que, elas auxiliam na avaliação de desempenho dos projetos elaborados. (OXMAN, 2008).

Diante deste cenário, viu-se a necessidade de adoção de métodos que atendessem a essa demanda crescente, o design generativo é uma dessas soluções, utilizando-se de algoritmos genéticos e elementos paramétricos, permite com que seja explorada a geração de diversas soluções de projeto. (TURRIN; VON BUELOW; STOUFFS, 2011).

Entretanto, alertam Keough e Benjamin (2010), que esse processo mesmo gerando diversas soluções não produz um único resultado correto, necessita-se ainda da avaliação e do julgamento dos projetistas.

Por fim, por se tratar de uma metodologia adaptada recentemente para o setor de arquitetura, engenharia e construção, faltam evidências das vantagens e limitações da aplicação do design generativo no estudo de concepção, portanto, considerando a necessidade amostral, este estudo busca ainda verificar sobre as evidências apontadas sobre o uso do método de design generativo.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, é apresentada a fundamentação teórica sobre o tema abordado no trabalho, abrangendo os conceitos de estudo de concepção, design computacional, design generativo e parâmetros de projeto.

2.1 ESTUDO DE CONCEPÇÃO

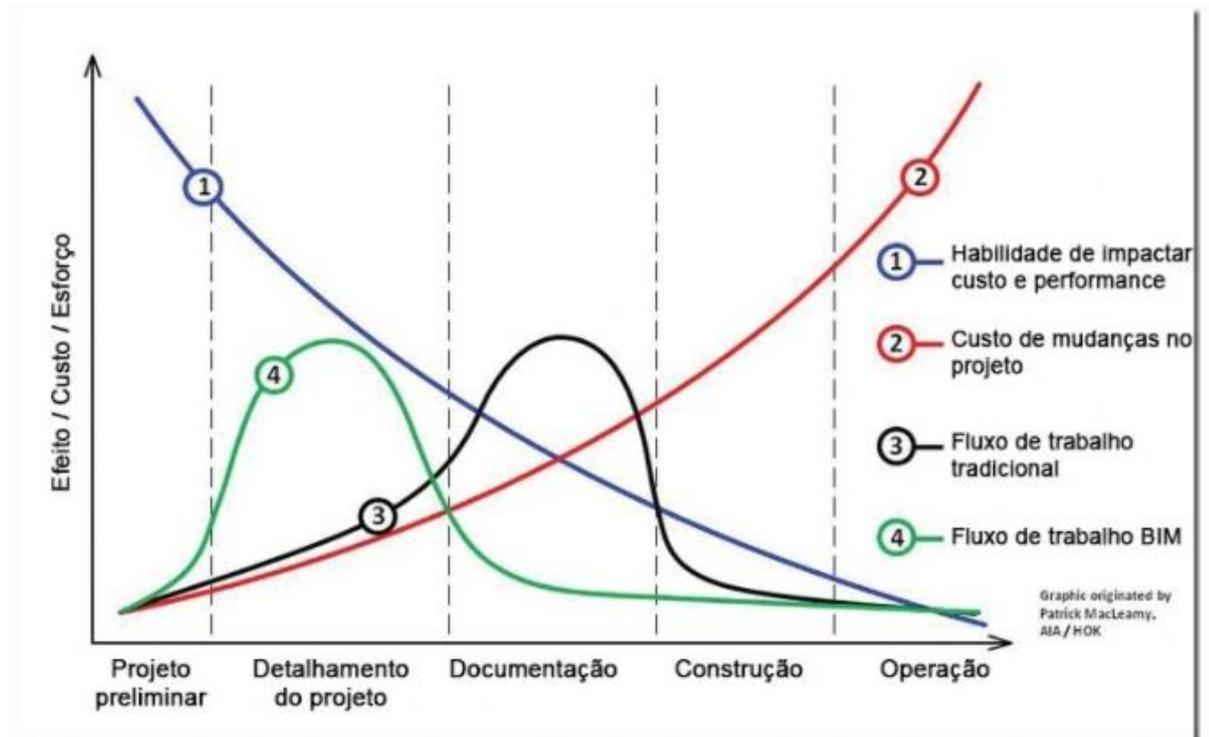
O estudo de concepção surge como uma etapa que busca incorporar um conjunto de questões que necessitam ser atendidas em uma resposta arquitetônica. Em outras palavras, a fase de concepção arquitetônica agrega diversos requisitos, como: conforto térmico, ventilação, viabilidade econômica e critérios funcionais e traduz tudo isso para um único resultado, o projeto conceito. (MEOTTI et al., 2015).

A etapa de concepção é a primeira fase no processo de criação do projeto arquitetônico. Seu principal objetivo é dar forma à ideia geral do projeto, buscando sintetizar todos os requisitos de projeto em um projeto conceito, agregando detalhes e informações suficientes para alimentar as etapas seguintes no processo de desenvolvimento arquitetônico. (MCCALL, 2018).

O primeiro passo no processo de criação do projeto conceito envolve definir os objetivos a serem atingidos e as restrições legislativas e técnicas. Depois disso, o processo torna-se iterativo com a exploração de múltiplas opções de soluções arquitetônicas, através da criação de esboços. Por fim, ao atingir o projeto que atenda aos objetivos e restrições inicialmente definidos, bem como, o conceito arquitetônico desejado pelo projetista, criam-se elementos visuais de representação da forma e volume, utilizando-se elementos de massa com representação genérica, seguindo assim, para as próximas etapas de criação focadas no refinamento do projeto conceito. (DAVIES, 2020).

Segundo o Instituto Americano de Arquitetos (2017), na sigla em inglês AIA, a etapa de concepção é crucial no processo de criação do projeto arquitetônico, exemplificado pela curva de MacLeamy, Figura 1, que representa como decisões tomadas no início de um projeto influenciam positivamente os resultados e reduzem os custos de alterações.

Figura 1 - Curva de MacLeamy



Contudo, aponta Schmitz (2017), em uma pesquisa realizada pela empresa PTC, que 67% dos projetistas participantes indicaram que do tempo total de um projeto, menos de um terço é gasto na fase de concepção e, além disso, 50% dos participantes indicaram que conseguem explorar no máximo três possibilidades de soluções arquitetônicas.

2.2 DESIGN COMPUTACIONAL

De acordo com Celani e Pupo (2008), o design computacional é um processo que visa o desenvolvimento de projetos por meios computacionais, através da incorporação do conceito paramétrico na geometria, buscando tornar o processo de projeto mais efetivo. A próxima etapa desse processo, apontam Shea, Aish e Gourtovaia (2005), é transformar o computador em um colaborador na geração de ideias e soluções para os projetos, utilizando-se de meios computacionais como a gramática de forma e os algoritmos genéticos.

2.2.1 Design Paramétrico

O design paramétrico consiste em desenvolver um projeto utilizando objetos que se relacionam entre si por meio de regras, carregadas como parâmetros nos objetos. Alterações realizadas nos parâmetros regeneram o elemento, criando novas formas. (EASTMAN et al., 2011). O termo parâmetro, originado da matemática, é definido como um valor constante para o caso considerado mas variável em casos diferentes, ou seja, é um valor mensurável que define um sistema ou determina seus limites. (HUDSON, 2010). Na arquitetura, os parâmetros são definidos como as dimensões da edificação ou os seus fatores de desempenho. (ELTAWEEL; SU, 2017). Assim sendo, em um processo paramétrico de projeto na arquitetura não utiliza-se apenas o conhecimento do projetista mas também define-se os parâmetros com suas regras e relações lógicas. (YU; GU; OSTWALD, 2013).

Originado em outros campos relacionados ao design, como a indústria aeroespacial e a automotiva, o design paramétrico começou a ser utilizado na arquitetura com a criação de novas ferramentas CAD (computer-aided design - ou desenho assistido por computador), no final dos anos 90. Ao design paramétrico são atribuídos vantagens quando comparado aos métodos tradicionais, como realizar a graficação de diferentes variações do projeto de forma mais rápida e ser mais adaptável a mudanças futuras, uma vez que seu processo de modelagem é baseado em regras não necessitando a reconstrução total do projeto. (NAGY et al., 2017a).

Como desvantagem, Aish e Woodbury (2005), apontam que a abordagem paramétrica demanda um maior esforço inicial do projetista, visto que, aumenta-se a exigência de atenção na modelagem dos elementos e o grau de complexidade das decisões.

2.2.2 Design Algorítmico

O design algorítmico é um processo iterativo, com o objetivo de resolver os problemas lógicos preestabelecidos através de repetidas operações, tendo como resultado um objeto. (UMAKOSHI, 2014). É um método que incorpora as teorias e métodos da ciência da computação no ambiente de projeto arquitetônico, transformando as condicionantes referentes aos programas arquitetônicos e aos códigos de construção em códigos computacionais, fazendo com que o arquiteto

deixe de manipular objetos para trabalhar com scripts algorítmicos. (TERZIDIS, 2002).

Portanto, a qualidade da aplicação do método está diretamente ligada com o domínio computacional e matemático do projetista. O conhecimento se deve ao fato da necessidade de entendimento de geometria para a criação das formas, da interpretação das condicionantes necessárias de serem aplicadas e à tradução de tudo que fora mencionado para códigos. (POTTMAN et al., 2007).

Uma ressalva, aponta Meredith (2008), é que a escolha final da solução arquitetônica, entre todas as opções geradas por meio de design computacional, deve ser realizada pelos projetistas, através de avaliações e processos críticos. Como um agente inconsciente como o computador está operando apenas algoritmos há a possibilidade de criação de soluções originais, resultados inesperados. Devido a isso, os computadores devem complementar o raciocínio humano e não assumir a posição de tomadores de decisões. (TERZIDIS, 2006).

2.3 DESIGN GENERATIVO

2.3.1 Definição

O design generativo é uma das subcategorias do processo de design algorítmico. (OXMAN, 2006). Esse método alinha conhecimentos da ciência da computação, como tecnologias de inteligência artificial, ao processo de criação de projetos. Utilizando-se de algoritmos metaheurísticos de busca é possível descobrir resultados de determinados problemas, ou seja, são aplicações que buscam resolver de forma genérica problemas de otimização. Por tratar cada problema de forma genérica, esse método tem o potencial de descobrir soluções inesperadas pelo projetista. (NAGY et al., 2017b).

Akella (2018) descreve o design generativo como uma réplica da abordagem evolucionária da natureza. Esse processo utiliza-se da computação em nuvem e de técnicas de aprendizado de máquinas para, a partir de seus objetivos de projeto, explorar todas as permutações possíveis de uma solução até encontrar a melhor.

Atualmente, existem duas técnicas principais de aprendizado de máquinas utilizadas para execução do design algorítmico, são elas: a técnica de gramática de forma e a técnica evolutiva. A primeira técnica é baseada em expressões

matemáticas que utilizam regras transformacionais para gerar as formas, a segunda baseia-se em modelos evolutivos de geração natural aplicados para geração da forma. (OXMAN, 2006).

Porém, no design generativo utiliza-se uma abordagem adaptada de inteligência artificial, os algoritmos de busca. Essa adaptação se deve ao fato de que, usualmente, os modelos de aprendizado de máquinas são treinados através de funções diferenciáveis e cálculos de gradientes, que acabam por indicar o caminho até a melhor solução. Entretanto, os modelos paramétricos de design são compostos por elementos que utilizam-se de uma combinação de computação com operações geométricas, transformando-os em um problema que não pode ser treinado diretamente por funções diferenciais. A solução, então, é utilizar um sistema que testa o modelo de fora para dentro, aprendendo como ele funciona, ou seja, os algoritmos de busca não entendem como é o funcionamento interno do modelo, eles utilizam-se de processos iterativos, testando configurações e aprendendo a cada repetição o que funciona e o que não funciona. (NAGY, 2017).

2.3.2 Otimização Multiobjetivo

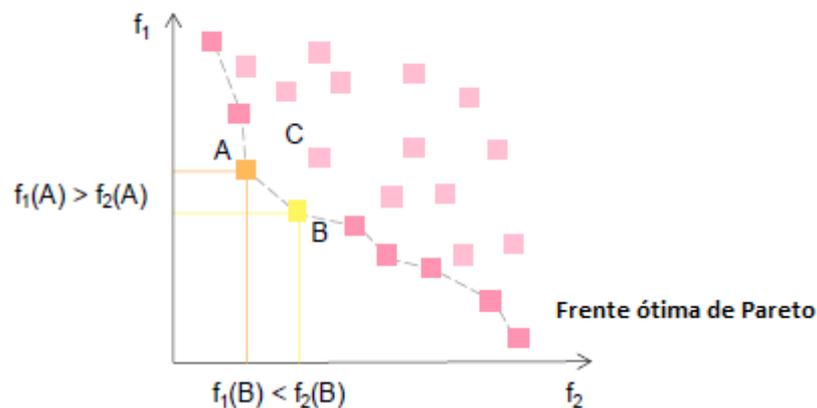
Por se tratar de uma abordagem algorítmica, a estruturação das condicionantes que serão aplicadas no design generativo devem ser pensadas previamente, contudo, sabe-se que problemas no setor de arquitetura, engenharia e construção, usualmente, exigem o atendimento de uma multiplicidade de requisitos que se contradizem por natureza. Por exemplo, minimizar o preço de venda de um apartamento é divergente do objetivo de aumentar a área útil por apartamento, devido a isso, ocorre a necessidade de aplicação de uma otimização multiobjetivo. (ROHRMANN, 2019).

Um problema de otimização multiobjetivo envolve a presença de dois ou mais objetivos conflitantes. Como cada objetivo pode ser maximizado ou minimizado e o problema pode conter uma série de restrições entre os objetivos, no problema de otimização multiobjetivo não há como existir uma única solução que otimize cada objetivo simultaneamente. O que ocorre nesse processo é a produção de *trade-offs* (escolhas), resultados conflitantes entre objetivos, ou seja, ao otimizar o problema em função de um dos objetivos acabará por comprometer a otimização de outro

objetivo. Portanto, o princípio da otimização é fornecer um conjunto de soluções que se encontram na frente ótima de Pareto. (DEB; KALYANMOY, 2001).

A frente ótima de Pareto é o conjunto de soluções potencialmente ótimas produzidas pela otimização de Pareto. É uma coleção de vetores que não podem ser dominados por outro, ou seja, na matemática, no contexto de um problema de otimização que busca maximizar todos os seus objetivos, um vetor x que apresente todos os componentes iguais ou superiores aos componentes de um vetor y , no caso da otimização de Pareto, o vetor x domina o vetor y . Portanto, para saber que um vetor é dominante ao outro, a relação de dominância precisa atender dois requisitos, o vetor x precisa ser tão bom quanto o vetor y para todos os objetivos e o vetor x tem que ser melhor que o vetor y em pelo menos um objetivo. (DRÉO et al., 2006). A Figura 2 exemplifica as soluções que atingem a frente ótima de Pareto.

Figura 2 - Frente ótima de Pareto



Fonte: Rohrmann (2019, p. 15, tradução nossa).

Como mostrados pela Figura 2, os pontos A e B não são dominados por nenhum outro ponto, por isso são classificados na frente ótima de Pareto, enquanto, o ponto C é dominado tanto por A como por B. Em suma, tem-se então, que o objetivo do design generativo é fornecer um conjunto de soluções convergentes à frente ótima de Pareto.

2.3.3 Algoritmo Genético

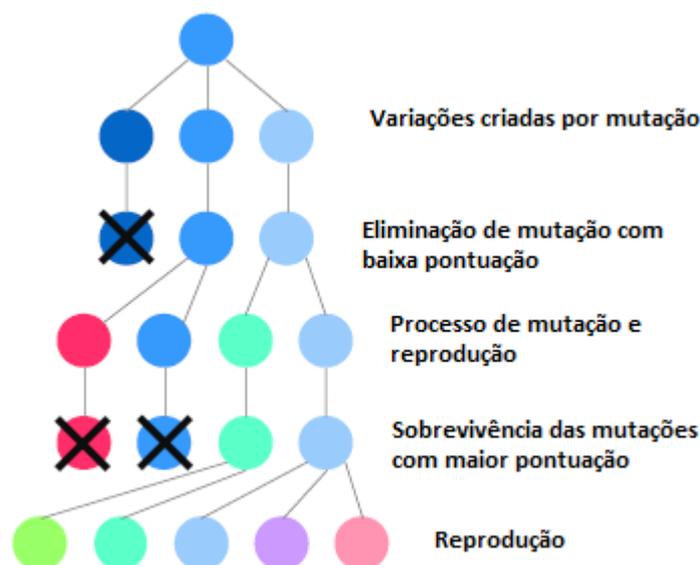
Holland (1992) foi o primeiro a cunhar o termo algoritmo genético e esse é o tipo específico de algoritmo de busca, usualmente, aplicado no design generativo. O

algoritmo funciona em quatro operações, imitando os conceitos da evolução darwiniana, são eles: a geração, seleção, reprodução e mutação. (NAGY, 2017).

A primeira etapa funciona com uma população inicial de indivíduos sendo criada aleatoriamente, ou seja, eles são da primeira geração. (ROHRMANN, 2019). Cada indivíduo recebe uma pontuação, baseada nos critérios estabelecidos no problema. Em seguida, ocorre a segunda etapa no qual os indivíduos com alta pontuação estão sujeitos a serem escolhidos para seguir para a reprodução, e os indivíduos com baixa pontuação tendem a “morrer”. (BEASLEY; BULL; MARTIN, 1993).

Na terceira etapa ocorre a escolha aleatória de dois indivíduos da primeira geração para criação de uma nova população, criando-se a população infantil da primeira geração. Uma característica das gerações seguintes à população pai é possuir uma maior proporção dos bons atributos da geração anterior. Portanto, por esse processo se tratar de um laço repetitivo, ao longo de muitas gerações, os melhores atributos acabam se reproduzindo entre si, garantindo que apenas os indivíduos com as maiores classificações formem as próximas gerações, convergindo para uma população de indivíduos ideais. O processo de mutação é aplicado tendo como objetivo manter a diversidade da população e diminuindo a chance de algum indivíduo com ótimos atributos não se reproduzir para as próximas gerações, esse processo é realizado através de entradas aleatórias nas gerações filho. (BEASLEY; BULL; MARTIN, 1993; DEB; KALYANMOY, 2001). Todo esse processo é exemplificado na Figura 3.

Figura 3 - Etapas do algoritmo



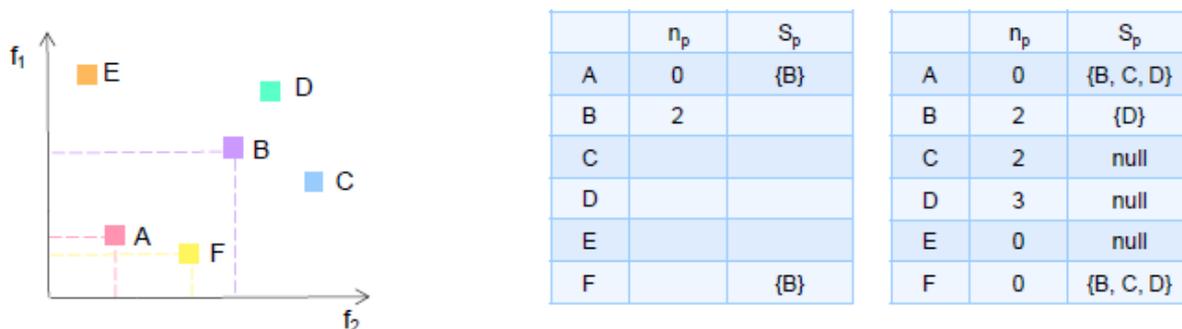
Fonte: Rohrmann (2019, p.11, tradução nossa).

2.3.3.1 Non-dominated Sorting Genetic Algorithm II - NSGA-II

Dentre os algoritmos para problemas de otimização multiobjetivo, o *Non-dominated Sorting Genetic Algorithm II* (NSGA-II), proposto por Deb et al. (1999), é o mais popular. O NSGA-II possui forte estratégia de elitismo, processo em que as populações filho e a população pai são classificadas juntamente, preservando as soluções com pontuações altas de etapas anteriores. Assim, ao avanço das próximas gerações apenas as boas soluções se reproduzirão, resultando em um conjunto que converge mais rapidamente para a frente ótima de Pareto, melhorando o desempenho dos algoritmos genéticos. (DEB et al., 2002).

O processo de classificação das populações pelo NSGA-II envolve dois procedimentos de ordenação: a ordenação da população em um ranking por não-dominância (*Fast Non-dominated Sorting Algorithm*) e a ordenação individual pela distância de multidão (*Crowding Distance Sorting*). A ordenação da população em um ranking por não-dominância serve para classificar soluções em grupos, ideal para avaliação de múltiplos objetivos. Em seu processo cada solução recebe um parâmetro n_p , número de soluções que dominam a solução em questão e S_p , as soluções que são dominadas pela solução em questão, esses parâmetros são preenchidos através de comparações, exemplificado pela Figura 4. (DEB et al., 2002).

Figura 4 - Processo de ordenação da população



Fonte: Rohrmann (2019).

Na Figura 4 é possível perceber que a solução “A” não é dominada por nenhuma outra solução, assim, seu parâmetro n_p é preenchido com 0, em contrapartida, a solução “B” é dominado por “A” e “F”, portanto, recebe o valor 2 em n_p , com “A” e “F” sendo dominantes a “B”. No parâmetro S_p é associada a informação do conjunto “B”, esse processo de avaliação ocorre para todas as soluções até que n_p e S_p estejam completos. Em seguida, começa um processo de iteração em que todas as soluções que possuem 0 em n_p , são classificadas como a primeira frente não dominada e o conjunto de soluções associadas a S_p dos elementos da primeira frente tem seus valores de n_p subtraídos por 1, depois desse processo, as soluções que possuírem $n_p = 0$ serão classificadas como segunda frente. Esse processo ocorre até que todas as soluções tenham uma frente definida, na Figura 4, “A” e “F” possuem valor 0 de n_p , logo, são da primeira frente não dominada, esses elementos possuem em seu S_p , o conjunto “B”, “C” e “D”, portanto, as soluções desse conjunto terão seus valores de n_p subtraídos por 2, um por estarem em S_p da solução “A” e um por estarem na solução “F”, devido a isso, “B” e “C” terão $n_p = 0$, conseqüentemente, serão agrupados como segunda frente não dominada. A Figura 5 ilustra essas iterações e o agrupamento das frentes. (DEB et al., 2002).

Figura 5 - Processo de agrupamento em frentes

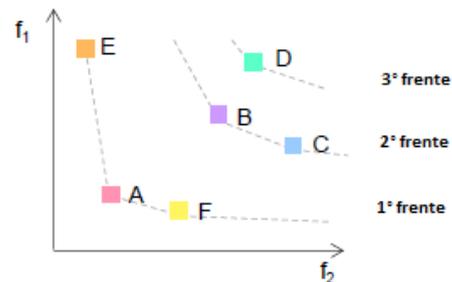
	n_p	S_p
A	0	{B, C, D}
B	2	{D}
C	2	null
D	3	null
E	0	null
F	0	{B, C, D}

→

	n_p	S_p
A	-	-
B	0	{D}
C	0	null
D	1	null
E	-	-
F	-	-

→

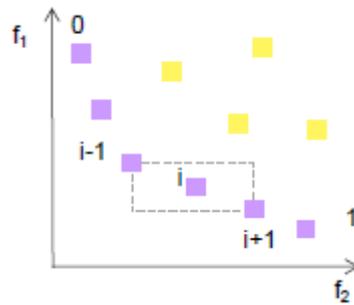
	n_p	S_p
A	-	-
B	-	-
C	-	-
D	0	null
E	-	-
F	-	-



Fonte: Rohrmann (2019, p.21, tradução nossa).

A ordenação individual pela distância de multidão é utilizada para que o algoritmo consiga manter um nível de variedade nas soluções encontradas, assim, ele irá preferir soluções em uma região menos densa. As distâncias de multidão são calculadas apenas entre as soluções de uma mesma frente não dominada. Esse valor é a distância média do lado do retângulo formado pelos pontos mais próximos à solução analisada, um ponto de cada lado da solução, como mostrado na Figura 6, após o cálculo, cada solução recebe dois atributos, um que identifica sua frente não dominada e outro que indica o valor da distância calculada. O algoritmo então define que as melhores soluções são aquelas que possuem menor valor no ordenamento da frente não dominada e caso o ordenamento seja igual, aquela solução que apresentar um valor maior de distância de multidão, pois se encontra em uma região menos populosa, será a solução com melhor pontuação. (DEB et al., 2002).

Figura 6 - Distância de multidão

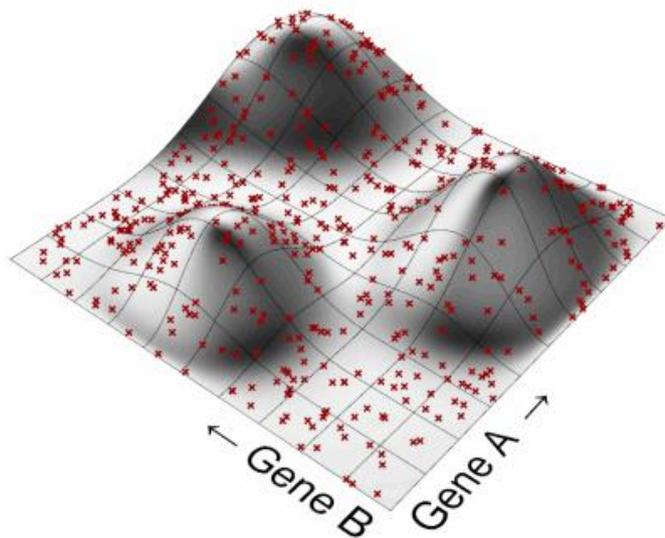


Fonte: Rohrmann (2019).

2.3.3.2 Galapagos

Desenvolvido por Rutten (2013), Galapagos, o algoritmo evolutivo incorporado ao software Grasshopper, possui características iguais aos denominados algoritmos genéticos, respeitando os conceitos apresentados da evolução darwiniana, sendo eles, as operações de geração, seleção, reprodução e mutação das soluções. De forma geral, o funcionamento do algoritmo executa os seguintes processos, de partida, as variáveis que serão otimizadas, denominadas também de genes, recebem diversos conjuntos de soluções, denominados de genomas, a Figura 7 ilustra essa etapa. (RUTTEN, 2010).

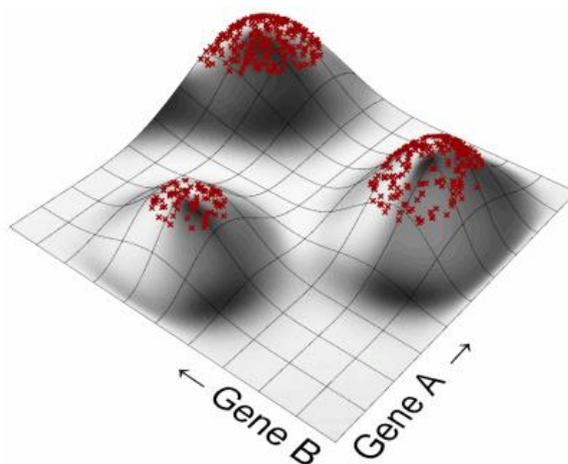
Figura 7 - Etapa inicial do algoritmo



Fonte: Rutten (2010).

Com os conjuntos criados, determina-se a pontuação dessas soluções, as maiores notas são atribuídas às soluções nos pontos mais altos, como mostra a Figura 7. Genomas com baixas pontuações são eliminados e as soluções sobreviventes então criarão as novas gerações, esses novos genomas criados possuirão pontuações relativas às pontuações obtidas por seus genomas pais, conseqüentemente, devido a esse processo iterativo, os genomas tenderão a se agrupar em torno dos picos de otimização até obter as soluções de maiores pontuações, esse agrupamento é mostrado na Figura 8. (RUTTEN, 2010).

Figura 8 - Agrupamento nos picos de otimização



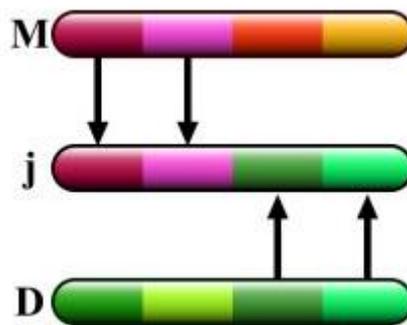
Fonte: Rutten (2010).

É dentro desse processo de otimização que as etapas de evolução agem. O processo de seleção serve para a escolha dos genomas pais que produzirão uma nova geração de soluções, de três possibilidades: A primeira é a seleção isotrópica na qual todas as soluções existentes podem ser escolhidas. Essa seleção serve, principalmente, para evitar que regiões de pontuação mais alta sejam alcançadas prematuramente, o que poderia acarretar na perda de genomas ótimos. Em contrapartida, o uso dessa seleção aumenta o tempo de processamento do algoritmo. A segunda seleção é a exclusiva, essa seleção permite apenas que as principais soluções criem novas gerações, resultando em gerações de genomas com boas pontuações, frutos da qualidade do descendente escolhido. Por fim, o último tipo de seleção é a tendenciosa, na qual soluções com maiores pontuações

possuem uma maior chance de reproduzir novas gerações. Essa seleção tende a ocupar espaços melhores de otimização gradualmente. (RUTTEN, 2010).

A etapa de reprodução tem como objetivo combinar as melhores qualidades dos genomas, criando assim, soluções cada vez melhores. A reprodução pode ocorrer através do processo de união das características dos genomas, assim, a solução filho herdar um número aleatório de qualidades de um dos genomas e os demais virão do segundo genoma pai, a Figura 9 exemplifica esse processo.

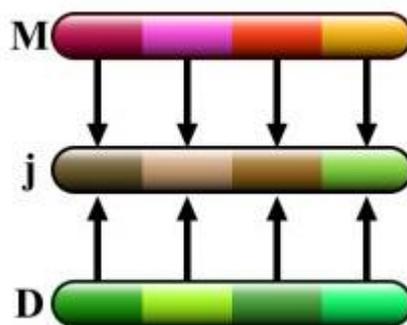
Figura 9 - Processo de união das características



Fonte: Rutten (2010).

Na Figura 9, dois genomas possuem quatro características cada, por aleatoriedade, na recombinação das qualidades para a solução filho, o genoma “M” doa duas características, e o genoma “D” as duas restantes. Na reprodução por mistura, o genoma filho receberá a média das características das soluções pais, como mostra a Figura 10. (RUTTEN, 2010).

Figura 10 - Processo de reprodução por mistura



Fonte: Rutten (2010).

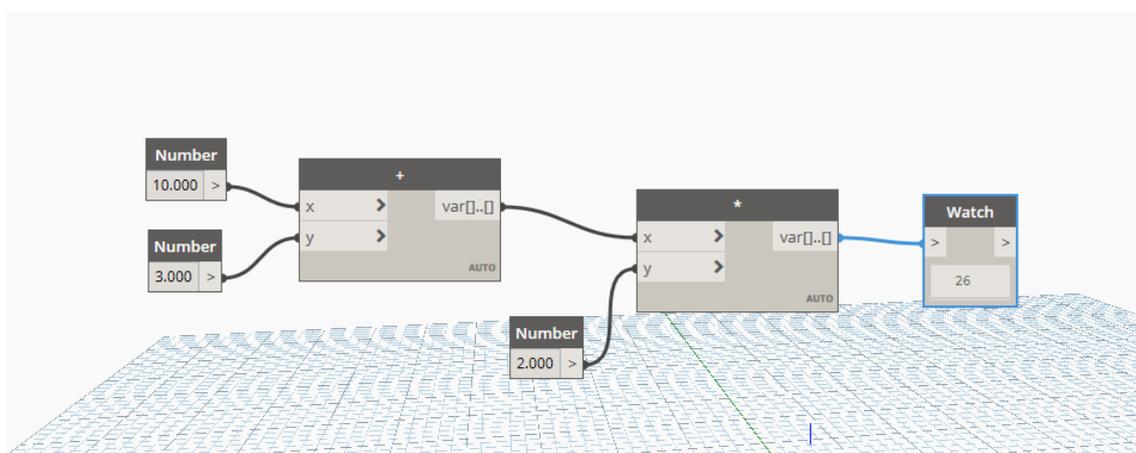
Ainda, na reprodução por mistura é possível adicionar uma variável de preferência, assim, aquele genoma que possuir a melhor pontuação terá significância maior na produção média das características do genoma filho. Entretanto, como mostrado, a etapa de reprodução não acrescenta nenhuma nova característica nas gerações filho, com isso, ao longo das gerações, tende-se a estabilizar os genomas em uma posição intermediária no plano de otimizações, diminuindo a diversidade de soluções. A fim de evitar isso, o processo de mutação é aplicado, o processo utilizado é chamado de mutação de ponto, ele consiste em alterar aleatoriamente o valor de uma das variáveis do genoma, permitindo assim, que as soluções explorem novas posições no plano de otimização. (RUTTEN, 2010). Segundo Sivanandam e Deepa (2008), deve-se ter atenção na definição da probabilidade em que a mutação ocorre, visto que, caso os genomas estejam sujeitos a pouca mutação, não haverá diversidade suficiente para avançar rumo ao pico no plano de otimização. Em compensação, caso a probabilidade esteja em um valor alto, as soluções estarão sujeitas à completa aleatoriedade, condição análoga à geração 0 de genomas na qual há soluções por todo plano de otimização, como mostrado pela Figura 7.

2.3.4 Programação Visual

O termo algoritmo pode ser descrito como um conjunto de instruções que seguem uma lógica básica contendo entrada, processamento e saída. É a descrição das etapas que serão realizadas passo a passo, como acontece em uma receita no qual os ingredientes são as entradas do algoritmo, o modo de preparo é o processamento e a saída é o resultado dessa receita. O ato de juntar todo esse conjunto de modo que o computador consiga ler e executar essas ações é denominado de programação de computadores. Contudo, uma das principais dificuldades da programação é escrever os algoritmos de forma que o computador possa entender. Para isso há diversas linguagens de programação e cada uma expressa suas instruções para o computador, diferente da outra. Devido a isso, surge a opção da programação visual, que tem como característica diminuir a barreira de entrada para todos os usuários pois utiliza-se de uma interface gráfica para construir as instruções a serem interpretadas pelo computador. (DYNAMOBIM, 2019).

A construção do algoritmo na programação visual passa pela estrutura de nós que é montada. Nós são os blocos lógicos que possuem uma função predeterminada, e através do relacionamento desses nós é que as instruções são executadas. Por exemplo um nó que tem a função de somar dois números, necessita como entrada dois números produzidos por nós anteriores a ele e produz como saída o número resultado da soma que pode ser aproveitado pelos próximos nós. (DYNAMOBIM, 2019). A Figura 11 exemplifica esse relacionamento entre nós.

Figura 11 - Exemplo de programação visual



Fonte: Dynamobim (2019).

Na Figura 11 é possível observar como o nó “Number” produz os números que o nó de soma “+” irá utilizar, na sequência das instruções, o nó de multiplicação “*” utiliza-se do resultado do nó de soma mais a entrada de um terceiro nó “Number” para produzir o resultado final da operação, visualizado no último nó “Watch”.

Como exemplo de softwares que operam por programação visual, o Dynamo é um software livre e independente que pode ser utilizado em conexão com o Autodesk Revit. É uma ferramenta de programação visual que oferece a possibilidade de criação de rotinas, nomenclatura utilizada para os algoritmos criados nesse ambiente, visando uma ampla variedade de aplicações, como processamento de dados até gerações de geometria. Uma dessas opções é o uso para estudo de otimizações no qual o Dynamo integra-se ao Refinery, projeto de design generativo da Autodesk, e utiliza o algoritmo genético NSGA-II do Refinery para explorar as opções de projetos desenvolvidos (DYNAMOBIM, 2019).

Outra opção de software de programação visual voltado para o mercado de arquitetura, engenharia e construção é o Grasshopper. O Grasshopper é um

software integrado ao Rhino que pode ser conectado ao Revit através de uma aplicação de terceiros, o Rhino.Inside. O Grasshopper possui função similar ao Dynamo, contudo, para estudo de otimizações de design generativo, o Grasshopper tem seus próprios nós desenvolvidos para esse fim, o Galapagos, que possui seu próprio algoritmo desenvolvido por Rutten (2013). (MEDINA, 2021; RUTTEN, 2013).

2.4 PARÂMETROS DE PROJETO

Definir os objetivos a serem atingidos é o que definirá o resultado do projeto gerado. Esses objetivos tendem a ser subjetivos ao projetista, como estética ou racionais (como lucro), contudo, independente aos tipos escolhidos, eles necessitam ser transformados em indicadores mensuráveis permitindo assim a comparação entre as opções de projeto geradas. (ROHRMANN, 2019).

A seguir são discutidos alguns parâmetros que baseiam o desenvolvimento de projetos, como os referentes à legislação e aprovação de projetos nas cidades brasileiras, custo e orientação solar.

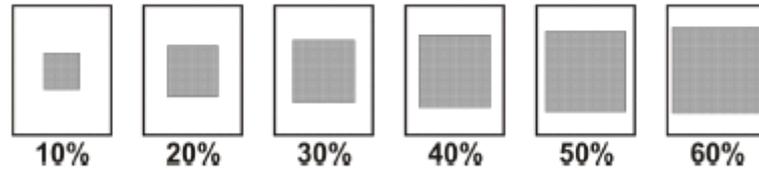
2.4.1 Parâmetros legais

2.1.4.1 Taxa de ocupação

Dentre o conjunto de normativas empregado por zoneamentos e planos diretores no Brasil, dois dos mais utilizados são a taxa de ocupação (TO) e o índice de aproveitamento (IA), coeficientes destinados ao controle da volumetria das edificações no terreno e ao controle das densidades construtivas, respectivamente. (SÃO PAULO (SP), 2014).

A taxa de ocupação é a relação entre a projeção da edificação e a área do terreno, normalmente, representado em porcentagem, por exemplo, uma projeção de 300 m² em um lote de 600 m² corresponde a uma TO de 50%. (ABUSSAMRA, 2007). A Figura 12 apresenta diferentes taxas de ocupação para a mesma área.

Figura 12 - Diferentes taxas de ocupação no mesmo terreno



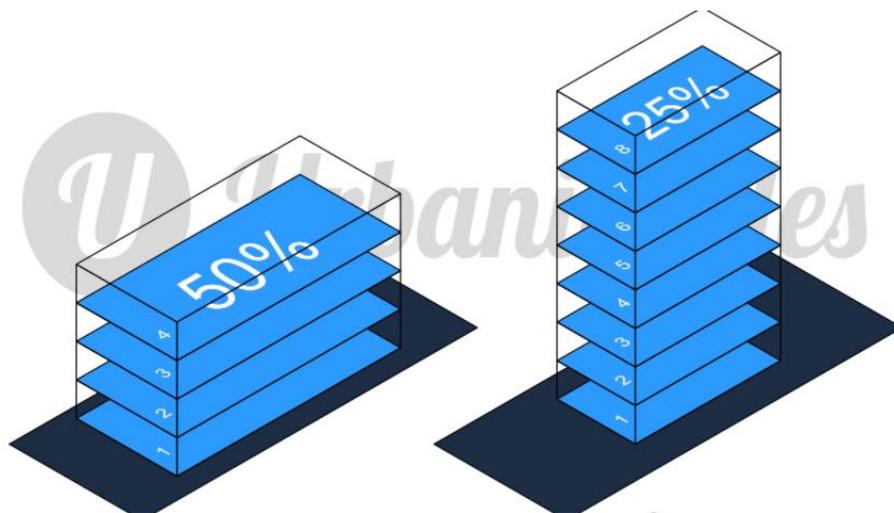
Fonte: Barros (2014)

2.1.4.2 Índice de aproveitamento

O índice de aproveitamento é a relação entre a área total construída e a área do lote, esse valor representa a área total que poderá ser construída no terreno. Exemplificando, a área total construída da edificação é de 900 m² divididos em 3 pavimentos de 300 m², em um terreno de 600 m², corresponde a uma IA de 1,5. (ABUSSAMRA, 2007).

A relação de ambos os parâmetros na aplicação de um projeto se dá na forma de quanto maior a TO utilizada, menor será a altura da edificação, criando assim, diversas possibilidades de projeto para o mesmo terreno, como exemplificado pela Figura 13.

Figura 13 - Mudança do índice de aproveitamento relacionado à taxa de ocupação



Fonte: Saboya (2020).

2.4.2 Parâmetros de custo

Segundo Queiroz (2001), orçamentação é o processo que fornece uma previsão do custo da obra antes do seu início. O orçamento é um dos pontos de partida para muitos serviços da construção civil, visto que, o custo de um empreendimento define a viabilidade financeira daquela obra. (ANDRADE; SOUZA, 2003). Sampaio (2004) aponta que a escolha pelo tipo de orçamento está relacionado à finalidade da estimativa e seu grau de precisão, os três principais tipos utilizados: (a) orçamento paramétrico; (b) orçamento discriminado; e (c) orçamento operacional.

O orçamento paramétrico é indicado para a etapa de estudos de viabilidade no qual os projetos estão em sua fase inicial. Consiste na aplicação de indicadores baseados em dados históricos de custos, de comparativos com projetos similares ou de índices de custos praticados pelo mercado. A metodologia mais simplificada utilizada é a multiplicação da área equivalente de construção por um custo unitário baseado no CUB (Custo Unitário Básico), divulgado pelo Sindicato da Indústria da Construção Civil de cada estado (NBR 12.721/2006). (GONZÁLEZ, 2008; QUEIROZ, 2001; ABNT, 2006).

O orçamento discriminado tem como objetivo calcular os custos totais da construção, considerando a quantidade de serviços com respectivas composições de custos (levantamento de insumos como materiais e mão de obra). Os preços unitários utilizados nessa etapa podem ser constituídos a partir de uma pesquisa de preços dos insumos dos fornecedores ou através de uma base de composição de custo genérica, como o SINAPI, distribuído pela CAIXA ou a Tabela de Composições de Preços para Orçamentos (TCPO) da PINI.

Por realizar uma discriminação dos serviços e atividades a serem realizados em obra e estes estarem associados ao custo unitário de execução, esse tipo de orçamento possui um grau maior de precisão quando comparado ao orçamento paramétrico. Contudo, existe um alta variabilidade de fatores e detalhes que podem imputar erro ao orçamento, dado a desatender variáveis como a dificuldade de execução e o local da obra, fatores que acabam influenciando nos custos da mão de obra. (CABRAL, 1988; SAMPAIO, 2004; GONZALEZ, 2008).

O orçamento operacional tem seu uso focado ao planejamento da obra, sendo assim, seus custos são modelados conforme sequência cronológica das

atividades no canteiro. A abordagem desse tipo de orçamento consiste em dividir a obra em centros de custos conforme as operações necessárias para a execução da obra, tendo como resultado, o cálculo do consumo de materiais, ao montante utiliza-se ainda constantes de consumo por serviço e a mão de obra é orçada a partir da composição e duração das equipes no canteiro.

Sua principal diferença frente ao orçamento discriminado é que no operacional considera-se os processos de produção, a partir da programação de atividades do planejamento, atingindo uma estimativa de custo detalhada, ao contrário do orçamento discriminado que é elaborado considerando apenas o produto final. Essa abordagem permite a identificação de fatores que possuem influência no custo do empreendimento, como o prazo da obra, tamanho da equipe gerencial, a utilização de equipamentos e seus tempos de permanência. (ANDRADE, 1996; CABRAL, 1988; HEINECK, 1986 apud KERN, 2005).

Conforme González (2008) é de suma importância estudar o custo de novas alternativas durante o desenvolvimento de um projeto, Mascaró (2010) mostra que no ponto de vista macro, os custos sofrem influências das relações geométricas do edifício. Devido a isso, alterações na forma do empreendimento impactam no custo final da obra, criando alternativas para a decisão do projetista.

A estrutura de um edifício pode ser vista como um conjunto de planos horizontais com interseção de planos verticais, contando ainda com espaços de circulação horizontais e verticais. Alterações no modo em que esses planos interagem resultam em alternativas de projeto, principalmente do ponto de vista econômico, quando se há o aumento de vão entre um dos planos acarreta maiores custos para o outro plano. (MASCARÓ, 2010).

Mascaró (2010) apresenta que há diferenças de custos entre os planos. Aqueles que envolvem o edifício são mais caros que os equivalentes internos, os planos verticais de fachadas chegam a ser de três a cinco vezes mais caros que seus pares internos enquanto os planos horizontais externos chegam a ser 20% ou 30% mais caros que os internos. A Quadro 1 apresenta o resultado de um levantamento de custos feito pelo autor, posteriormente, compilado segundo os planos funcionais do edifício e as instalações, indicando a participação de cada item no custo de uma obra.

Quadro 1 - Participação dos itens no custo total

Classificação do elemento	Composição	Participação (%)
Elementos que formam os planos horizontais	Parte horizontal da estrutura e das fundações, telhado, pisos e parte horizontal dos revestimentos e da pintura.	26,79
Elementos que formam os planos verticais	Parte vertical da estrutura e das fundações, alvenarias, aberturas, revestimentos interno e externo, parte vertical da pintura.	44,84
Instalações	Elétrica, telefônica, hidráulica, gás, louças e metais e elevador.	24,33
Instalações provisórias, limpeza da obra e outros trabalhos não considerados		4,02

Fonte: Adaptado de Mascaró (2010, p.45).

Referente aos percentuais expostos no Quadro 1, Mascaró (2010) conclui que visando à redução de custos, o estudo de alternativas deve ser focado nos elementos que formam os planos verticais, que representam aproximadamente 45% do custo total da obra, enquanto, para os planos horizontais têm-se opções reduzidas de alternativas, visto que, 67% do custo total desses planos são destinados à estrutura que são, em sua grande maioria, de concreto.

Mayer (2012) endossa a ideia afirmando que ao manter constante o tipo e qualidade do material empregado, o custo da edificação poderá ser alterado em função da sua forma, sendo a forma relacionada a sua compactidade. Logo, quanto mais compacta a construção, menor o seu custo. Definido por Rosso (1978), o índice de compactidade (IC), equação (1), é a relação percentual que se estabelece entre o perímetro de um círculo de igual área do projeto e o perímetro de paredes exteriores do projeto.

$$IC = \frac{2\sqrt{A_p \cdot \pi}}{P_p} \times 100 \quad (1)$$

Em que,

IC = índice de compacidade;

A_p = superfície do projeto;

P_p = perímetro das paredes exteriores do projeto.

Mascaró (2010) apontou que o índice de compacidade não indicava com exatidão o custo das fachadas, visto que, arestas e curvas implicam aumento do custo. Devido a isso, o IC foi readequado para Índice Econômico de Compacidade (IeC), equações (2) e (3), incorporando o número de arestas e perímetros curvos de fachadas na equação.

$$IeC = \frac{2\sqrt{A_p \cdot \pi}}{P_{ep}} \times 100 \quad (2)$$

Em que,

IeC = índice econômico de compacidade;

A_p = superfície do projeto;

P_{ep} = perímetro econômico do projeto.

$$P_{ep} = P_{pr} + 1,5P_{pc} + \frac{nA}{2} \quad (3)$$

Em que,

P_{ep} = perímetro econômico do projeto;

P_{pr} = perímetro das paredes exteriores retas;

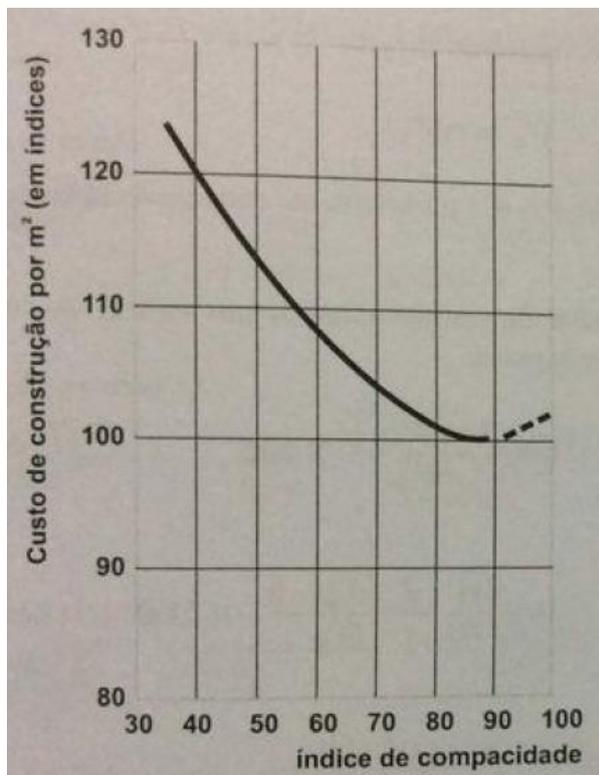
P_{pc} = perímetro das paredes exteriores curvas;

nA = número de arestas das fachadas.

O valor máximo para o índice é de 100%, contudo, esse valor só é atingido com o edifício em formato circular. Porém, arestas curvas têm acréscimo no custo, portanto, o valor ótimo do índice é atingido quando se tem a geometria quadrada do

prédio, no valor de 88,6%, conseqüentemente, é nesse valor que se obtém o custo mínimo de obra. (MASCARÓ, 2010). A Figura 14 mostra a variação do custo de construção conforme o incremento no valor do índice.

Figura 14 - Curva do índice de compacidade pelo custo da construção



Fonte: Mascaró (2010, p.57)

Mascaró (2010) conclui que devido ao achatamento da curva na região de valor ótimo, 88,6%, não há a necessidade de se atingir esse valor para um edifício econômico, o importante é obter um valor dentro dessa região que resultará numa provável perda de compacidade, e um pequeno incremento no custo, mas permitirá ganhos nas condições de habitabilidade.

2.4.4 Incidência Solar

A quantidade de luz natural que um imóvel recebe está diretamente relacionada com a sua orientação solar. A baixa exposição à luz solar afeta a qualidade de vida dos usuários, podendo acarretar em problemas psicológicos, como falta de disposição; perda de produtividade e depressão. Também afeta o conforto térmico da edificação, pois ambientes de permanência prolongada, como

quarto; salas e cozinha, necessitam ser ambientes com temperatura agradáveis para períodos longos de uso e devido à qualidade antisséptica da luz solar. A baixa exposição pode causar o surgimento de mofo nos ambientes. (ANTUNES, 2020; ESTADÃO, 2019). Segundo o Creci-SP, apartamentos com boa incidência da luz solar valorizam até 10% mais do que aqueles que não possuem boa orientação solar.

O sol nasce voltado à direção Leste e se põe voltado à direção Oeste, o Brasil, por possuir em quase sua totalidade território no hemisfério sul, recebe a maior insolação diária nas faces voltadas ao Norte, em contrapartida, as faces voltadas ao Sul recebem a menor quantidade de luz solar entre todas as fachadas, as faces à Leste recebem a insolação na parte da manhã e o Oeste na parte da tarde. Ainda, há o efeito de inércia térmica no qual o calor é retido ao longo do dia e durante a noite, ele é liberado. Assim, aqueles ambientes voltados ao lado que recebe sol pela manhã terão temperaturas mais amenas que ambientes que recebem o sol da tarde.

3 METODOLOGIA

Este capítulo descreve o método de pesquisa utilizado para atingir o objetivo do trabalho. Como estratégia de pesquisa foi realizado um estudo de caso, utilizando-se de duas rotinas, construídas com auxílio de programação visual, nos softwares Dynamo e Grasshopper para analisar a aplicação do design generativo voltado para elaboração de projeto arquitetônico na etapa de estudo de concepção.

As rotinas utilizaram como base para construção da edificação, um terreno hipotético, localizado em Novo Hamburgo, no bairro Jardim Mauá. O terreno possui um formato não regular, no intuito de criar diversidades ao algoritmo no posicionamento da edificação.

Na primeira parte do trabalho, de comparação entre os softwares, a geometria da edificação a ser construída foi limitada ao formato retângulo, assim, nivelando o tipo de soluções que poderiam ser gerados pelos softwares, ao realizar o comparativo, as diferenças seriam aparentes apenas nos critérios escolhidos, sendo: velocidade de processamento, qualidade dos resultados e otimização de variáveis, pois, para efeitos de comparação, caso o formato para criação da edificação fosse livre, não haveria um mesmo denominador entre os diferentes algoritmos. Desse modo, no comparativo, isso poderia produzir uma vantagem “oculta” para algum dos softwares.

Os parâmetros de projetos analisados na primeira parte do estudo foram referentes à taxa de ocupação, índice de aproveitamento, considerando os valores apontados pelo plano diretor da cidade de Novo Hamburgo referente ao setor SM3, setor que abrange a região em que o terreno estava localizado, e custo.

Na segunda parte, incrementou-se a rotina do software que apresentou melhor desempenho, dessa vez, buscando testar os limites da aplicação, mudou-se a lógica de criação da edificação, dando mais liberdade para escolha da geometria e foram adicionados dois novos parâmetros de projeto para análise, orientação solar e índice de compacidade.

3.1 DELINEAMENTO DO TRABALHO

A pesquisa desenvolvida neste trabalho foi dividida em cinco etapas, de modo a clarificar todos os processos envolvidos na construção deste trabalho, sendo elas:

validação dos softwares utilizados; criação das rotinas; comparação entre os softwares; incrementação de variáveis; e análise dos resultados. O Quadro 2 mostra as etapas, questões de pesquisa que basearam cada uma e respectivos resultados.

Quadro 2 - Resumo das etapas

Etapa	Objeto de estudo e objetivo	Resultados
Validação dos softwares	Objeto de estudo: problema matemático de otimização; Objetivo: verificar os algoritmos do Dynamo e Grasshopper.	Otimizações produzidas por Dynamo e Grasshopper e resultado do cálculo manual.
Criação das rotinas	Objeto de estudo: edificação produzida pelo método de design generativo; Objetivo: produzir soluções de projeto arquitetônico otimizando parâmetros de projeto.	Soluções de projeto arquitetônico produzidos pelo método de design generativo através de rotinas de programação visual.
Comparação entre os softwares	Objeto de estudo: resultados produzidos pelos softwares Dynamo e Grasshopper; Objetivo: avaliar os softwares para design generativo, utilizando os critérios de comparação: velocidade de processamento, qualidade dos resultados encontrados e quantidade de variáveis avaliadas.	Definição do software de melhor desempenho para o estudo de caso do trabalho.
Incrementação de variáveis	Objeto de estudo: edificação produzida pelo método de design generativo; Objetivo: testar a aplicação adicionando dois novos parâmetros de projetos para serem otimizados.	Soluções de projeto arquitetônico produzidos pelo método de design generativo através de uma rotina de programação visual.
Análise dos resultados	Objeto de estudo: resultados gerados pelo software que apresentou melhor	Análise dos resultados produzidos pela aplicação do método de design

	desempenho; Objetivo: analisar a aplicação do método de design generativo voltado para projetos arquitetônicos na fase de estudo de concepção.	generativo.
--	---	-------------

Fonte: Elaborado pelo autor.

A primeira etapa consistiu em validar os algoritmos genéticos utilizados no Dynamo e Grasshopper, foi utilizado um problema matemático que pudesse ter seu valor de otimização obtido através de cálculo manual.

Na segunda etapa, a criação das rotinas, desenvolveu-se toda a programação visual necessária para execução do design generativo, após a conclusão dessa etapa, realizou-se a comparação entre os softwares, a fim de, estimar qual era melhor para a aplicação que estava sendo desenvolvida.

Na quarta etapa, foi realizado incremento de variáveis na programação visual do software escolhido na terceira etapa, por fim, a quinta, e última, etapa, consistiu nas análises realizadas nos resultados gerados pelo processo de design generativo.

3.2 ETAPAS DE PESQUISA

A seguir a realização das cinco etapas serão detalhadas.

3.2.1 Validação dos softwares

Antes de iniciar a construção das rotinas que seriam utilizadas para construção dos projetos arquitetônicos, foi decidido verificar os algoritmos genéticos que seriam utilizados, isso se deve ao fato, de que os algoritmos foram implementados por terceiros, portanto, não se tinha acesso as linhas de código para possíveis conferências, e na continuidade do trabalho, os problemas de otimização iriam escalar para um grau de complexidade maior, gerando assim, uma impossibilidade de atestar se as soluções encontradas eram as mais otimizadas possíveis. Devido a isso, optou-se por provar que os algoritmos genéticos utilizados eram capazes de encontrar a melhor solução de otimização, essa validação ocorreu através de um problema de otimização matemático, que fora resolvido por cálculo

manual, e depois comparado com os resultados obtidos pelos softwares Dynamo e Grasshopper.

No caso das rotinas criadas, elas eram rotinas auxiliares, não tendo relação com as rotinas criadas nas próximas etapas do trabalho, elas foram criadas a parte, com intuito, apenas, de verificação dos algoritmos genéticos. Por fim, o problema matemático escolhido para resolução foi: uma lata cilíndrica aberta no topo deve conter 500 cm^3 de líquido. Encontrar a altura e o raio que minimizam a quantidade de material necessário para confeccionar a lata.

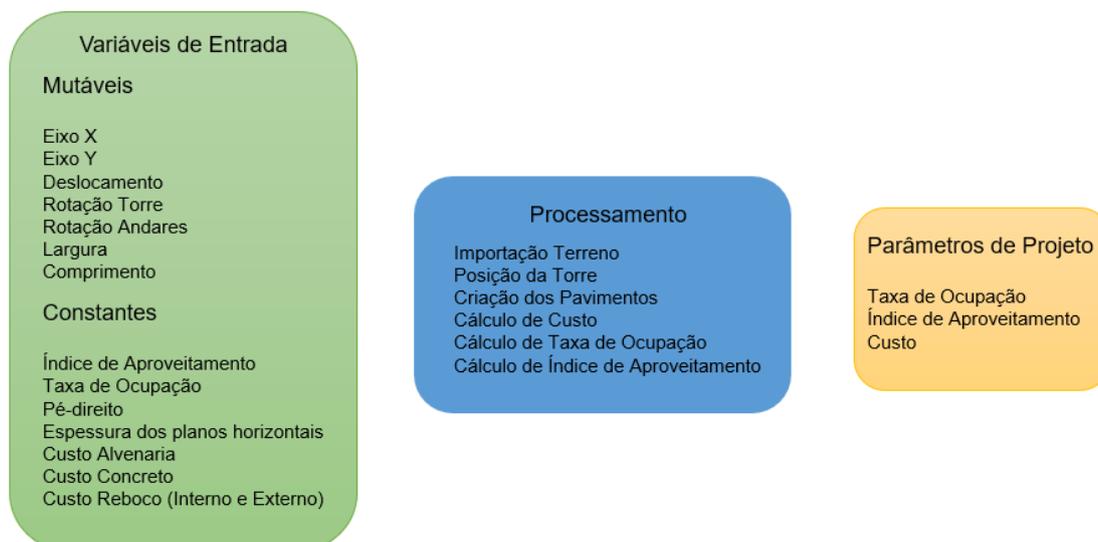
3.2.2 Criação das rotinas

Antes de iniciar a construção das rotinas, foi necessário estruturar quais informações eram relevantes nessa etapa do trabalho. Visando ao produto final que essa etapa deveria ter para atender os objetivos traçados, a primeira informação pensada para a rotina foram quais parâmetros de projeto seriam utilizados, decidindo então, pela utilização dos parâmetros: taxa de ocupação, índice de aproveitamento e custo.

Em seguida, estruturou-se quais seriam as variáveis de entrada da rotina, que foram divididas em duas categorias: as constantes, que teriam seus valores fixos, necessárias para realização de algum processo ou cálculo dentro da rotina, e as mutáveis, que poderia ter seus valores alterados, permitindo assim, que o algoritmo realizasse iterações no processo de otimização.

A terceira parte da estrutura da rotina era a de processamento, uma etapa intermediária entre as duas já citadas, que realizava toda a lógica da rotina, envolvendo o método de criação da edificação, tratamento das variáveis de entrada, e cálculos para obtenção dos resultados dos parâmetros de projeto que seriam otimizados. A Figura 15 compila a estrutura da rotina, citada anteriormente.

Figura 15 - Estrutura da rotina



Fonte: Elaborado pelo autor.

3.2.2.1 Parâmetros de projeto

Após definir quais parâmetros de projeto seriam utilizados, foi preciso se certificar que os resultados gerados pelas variáveis de índice de aproveitamento, taxa de ocupação e de custo fossem quantificáveis, pois é, unicamente, utilizando-se de valores mensuráveis que o algoritmo genético consegue realizar as otimizações.

3.2.2.1.1 Taxa de ocupação

A TO é a relação entre a área total construída e a área do terreno, na prática, esse coeficiente é utilizado para se descobrir a área de projeção máxima construída que pode ser utilizada, através do produto entre o coeficiente, informado por norma de cada cidade, e a área total do terreno.

Em um projeto, usualmente, tende-se a aproveitar ao máximo do valor permitido pela taxa de ocupação, contudo, no processo de design generativo é necessário conectar o objetivo a um valor quantificável para poder otimizá-lo. Devido a isso, neste trabalho, a taxa de ocupação foi otimizada através da subtração entre o valor limite total e o valor gerado pela projeção do prédio, visando, então, minimizar esse valor obtido, assim, quanto menor for o resultado da subtração, mais se aproveitou do valor total permitido pela TO. A equação (4) a seguir demonstra como a pontuação da taxa de ocupação foi realizado.

$$PTO = (TO * AT) - APT \quad (4)$$

Em que,

PTO = pontuação da taxa de ocupação;

AT = área do terreno;

APT = área da projeção total.

Como esse processo de subtração pode gerar um valor negativo, caso a área de projeção construída pela rotina seja maior que o permitido pela TO, necessitou-se adicionar uma condicional de penalidade, assim, ao obter um valor negativo na equação, a pontuação resultante seria de 100000, conflitando com o objetivo de minimização dessa pontuação.

3.2.2.1.2 Índice de aproveitamento

O IA é a relação entre a área total construída e a área do lote, no trabalho, esse objetivo não teve nenhum valor atribuído visando a otimização, o índice foi utilizado para descobrir quantos andares o empreendimento poderia ter, atendendo a normativa. Primeiro, calculou-se a área total que poderia ser utilizada nesse projeto, esse valor é o resultado entre o produto do índice de aproveitamento informado pelo plano diretor de cada cidade e a área total do terreno, a divisão desse resultado pela área de um pavimento, resultou no número de pavimentos possíveis. A equação (5) exemplifica o cálculo.

$$NPav = \frac{(IA * AT)}{APav} \quad (5)$$

Em que,

NPav = número de pavimentos;

IA = índice de aproveitamento;

AT = área do terreno;

APav = área do pavimento.

3.2.2.1.3 Custo

Para a fase de concepção, o tipo de orçamento mais indicado seria o orçamento paramétrico, utilizando os valores do CUB pela área total, contudo, o CUB possui diversas classificações de preços dependente dos padrões de acabamentos, tipo do empreendimento e quantidade de pavimentos, isso abriria um leque de variantes que não estavam envolvidas na montagem das rotinas, por exemplo, a decisão do padrão de acabamento não é uma variável possível de ser quantificável, dificultando o entendimento do algoritmo. Assim, para criação da rotina, optou-se por utilizar uma versão adaptada do orçamento discriminado adotando o conceito de planos horizontais e verticais de Mascaró (2010), visto que, por ser um estudo de concepção não há a disponibilidade de todos os projetos finalizados.

Para os planos horizontais, o custo foi realizado através do produto do volume do plano pelo preço da composição retirado do SINAPI, nos planos verticais, o cálculo para o custo é o mesmo, exceto, que foi utilizado a área dos planos verticais, por fim, foi somado ambos os valores, resultando num custo geral do projeto. A equação (6) exemplifica o cálculo.

$$CT = (VPH * PCV) + (APV * PCA) \quad (6)$$

Em que,

CT = custo total;

VPH = volume dos planos horizontais;

PCV = preço da composição volumétrica;

APV = área dos planos verticais;

PCA = preço da composição por área.

Como o resultado da soma é um valor quantificável foi possível adotar a otimização nesse resultado, assim, o objetivo foi minimizar o valor gerado pois quanto menor o custo do projeto, maior a possibilidade de ser viável financeiramente.

3.2.2.2 Construção da programação visual

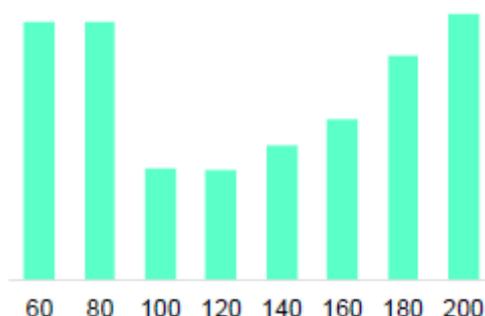
Após organizar todos os pontos importantes para o correto funcionamento da rotina, a parte final dessa etapa, foi implementar, tanto no Dynamo como no Grasshopper, toda a estrutura explicada anteriormente, para isso, utilizou-se do método de programação visual, e seus conjuntos de nós e recursos disponíveis em cada software.

3.2.2.3 Configuração do algoritmo genético

Com as rotinas construídas, para obtenção dos resultados, foi preciso rodar o algoritmo genético, contudo, há um conflito na escolha do tamanho da população e no número de gerações ao se configurar o algoritmo genético para rodar o problema de otimização, grandes populações fazem com que se tenha uma maior demora para a convergência para a frente ótima de Pareto mas gerarão melhores soluções, com populações menores, o processamento é mais rápido, pois não há a necessidade de eliminar tantos genomas com pontuações intermediárias, que acabam por atrasar a chegada ao pico da curva de otimização, contudo, poderá acarretar em uma menor pontuação das populações finais. Todavia, o método para determinar os melhores valores para o tamanho da população e para o número de gerações levaria a um outro processo de otimização, entretanto, esse processo demanda um alto nível de processamento computacional, a alternativa é através de experimentação, devido a isso, na configuração de ambos os algoritmos genéticos será utilizado os valores encontrados por Rohrmann (2019). (GOTSHALL; RYLANDER, 2002; ROHRMANN, 2019).

No seu estudo Rohrmann (2019), realiza medições de performance com oito configurações de tamanhos de população diferentes, chegando ao resultado de melhor performance com o valor de 120, como mostra a Figura 16.

Figura 16 - Performance com diferentes tamanhos de população



Fonte: Rohrmann (2019, p.76).

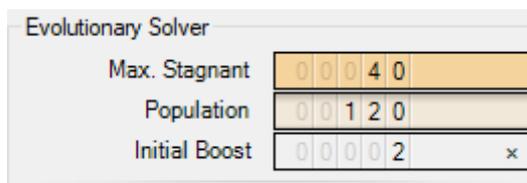
Ainda, com a sequência dos testes, Rohrmann (2019) obtém que até a geração 40, a pontuação das soluções oscila ao longo das gerações mas de forma geral, melhora gradualmente, contudo, após a geração 40, a melhora das soluções tende a se estabilizar, mantendo essa característica até a geração 50, valor que foi o último testado. Assim, qualquer valor escolhido entre 40 e 50 produziria resultados semelhantes, devido a isso, obteve-se por utilizar o valor de 40 para o número de gerações, visando o tempo de processamento um pouco mais rápido que os demais valores. Tem-se então, a configuração no Dynamo, como mostrado pela Figura 17, *Population Size* com valor de 120, *Generations* com 40 e *Seed*, um parâmetro para gerar a aleatoriedade dos valores produzidos na primeira geração, com valor 1, para o Grasshopper, como mostrado pela Figura 18, tem 40 para *Max. Stagnant*, nomenclatura utilizada para as gerações, *Population* com valor de 120, e o parâmetro que realiza a mesma função que o *Seed* do Dynamo, o *Initial Boost*, com valor 2.

Figura 17 - Configuração no Dynamo

Generation Settings	
Population Size	120
Generations	40
Seed	1

Fonte: Elaborado pelo autor.

Figura 18 - Configuração no Grasshopper



Fonte: Elaborado pelo autor.

Com o algoritmo configurado foi possível rodar a otimização, produzindo assim, os dados utilizados na comparação realizado na etapa seguinte.

3.2.3 Comparação entre os softwares

Para avaliar qual software apresenta melhores características visando ao prosseguimento do trabalho, foram escolhidos três critérios, sendo eles: a duração do processamento das rotinas para obtenção dos resultados; a qualidade dos resultados produzidos; e a capacidade do software de otimizar variáveis.

O critério de velocidade de processamento foi considerado devido questões de usabilidade, ao utilizar um sistema de otimização por algoritmo genético, mesmo que o sistema produza os melhores resultados, torna-se inviável a aplicação dele, se o mesmo demora muito tempo para obter os resultados. Rohrmann (2019) cita que os prazos designados para as etapas de projeto são limitados, assim, o benefício que o algoritmo genético busca trazer, a possibilidade de testar diversas soluções em pouco tempo, tornaria impraticável, caso os prazos para produção de resultados fossem demasiadamente longas. Assim, para avaliar esse critério, foi feita a marcação do tempo de início e fim que os softwares ficaram rodando, com o objetivo de tentar produzir o resultado que refletisse apenas o tempo de processamento dos algoritmos, todas as variáveis externas que poderiam produzir alguma diferença foram equiparadas. Assim, foi utilizado o mesmo computador para rodar as duas rotinas, e por se tratarem de softwares diferentes, ambos foram integrados para dentro do Autodesk Revit, o Grasshopper com o plugin Rhino.Inside e o Dynamo já sendo um plugin para o Revit, desse modo, os elementos produzidos possuiriam o mesmo motor gráfico para serem criados, igualando o tempo de processamento nessa parte.

Nagy (2017) aponta que a forma como o algoritmo entende o problema é através de um processo iterativo, testando possibilidades, assim, o que o algoritmo

consegue interpretar são os valores de entrada que estão sendo manipulados por ele, e a partir dos valores que resultam na saída, é que o algoritmo aprende sobre o problema. Portanto, isso pode gerar situações em que as soluções encontradas sejam totalmente irrealistas, resultados impossíveis de serem concretizados, devido a isso, o segundo critério adotado para avaliar os softwares foi sobre a qualidade dos resultados produzidos. A avaliação desse critério foi totalmente baseada na opinião do autor deste trabalho, através do racional de que as soluções geradas eram ou não possíveis de serem adotadas em um projeto real.

O terceiro, e último critério considerado na avaliação nesta etapa, foi a quantidade de variáveis que o software é capaz de otimizar simultaneamente. Deb e Kalyanmoy (2001) mencionam de como a otimização de múltiplas variáveis é um processo de ceda, em que para obter ganho em uma variável acaba comprometendo o valor de outra variável, por isso, quanto mais variáveis são avaliadas simultaneamente, maior será o tempo de processamento para realizar esse processo de ganhos e perdas até convergir todas as variáveis para a frente ótima de Pareto. A avaliação desse critério foi analisar se o algoritmo conseguia rodar as otimizações com mais de uma variável ao mesmo tempo, ou não.

3.2.4 Incrementação de variáveis

Com a etapa de comparação concluída, o trabalho seguiu com a adição de novos parâmetros de projeto e modificações de algumas lógicas implementadas na fase de processamento da rotina, no software que apresentou melhor desempenho, buscando testar os limites da aplicação.

Assim, a nova rotina constou com os acréscimos dos parâmetros: índice econômico de compacidade e incidência solar, e as modificações realizadas na lógica utilizada para criação da geometria da edificação, devido a isso, a lista de variáveis de entrada sofreu alterações, como: a retirada das variáveis mutáveis de rotação da torre, rotação dos andares, largura e comprimento, a adição de seis pares de novas variáveis para controle das coordenadas dos pontos no eixo X e no eixo Y e a adição de novos valores constantes, latitude e longitude da localização do terreno e data de início e fim para o estudo solar.

Em consequência dessas alterações, a estrutura da rotina foi alterada, conforme mostra a Figura 19.

Figura 19 - Nova estrutura da rotina



Fonte: Elaborado pelo autor.

3.2.4.1 Índice econômico de compacidade

Sendo um complemento para o parâmetro já utilizado de custo, a escolha pelo índice econômico de compacidade, deve-se ao fato, de que seu resultado está relacionado à forma do prédio, buscando assim, observar como o algoritmo vai se relacionar em relação a geometria do prédio, na tentativa de otimizar essa variável.

Para o cálculo desse parâmetro, foi utilizada a equação (2) apresentada por Mascaró (2010) para o índice econômico de compacidade, como esse índice tem relação com o custo do empreendimento, e como citado por Mascaró (2010), quanto maior o valor do índice, menor será o custo, tendo seu valor ótimo em 88,6%, o objetivo definido para essa variável no algoritmo será de maximizar o valor gerado pela equação de IeC .

3.2.4.2 Incidência solar

Como apontado por Antunes (2020), a baixa exposição à luz solar pode acarretar em diversos problemas, afetando assim, a qualidade de vida das pessoas, devido a isso, a segunda, e última, variável adicionada para otimização nesta etapa do trabalho, foi a incidência solar que o prédio está exposto.

O cálculo para a incidência solar realizado foi uma média ponderada entre a exposição solar que cada plano vertical do prédio está exposto pela área da

superfície, como mostra a equação (7). Com esse valor foi possível verificar a exposição solar média em que o prédio estava exposto, devido a isso, configurou-se a variável de incidência solar no algoritmo para maximizar o valor gerado como resultado.

$$\bar{x}_{Is} = \frac{\sum_{k=0}^n (Is_k * As_k)}{\sum_{k=0}^n As_k} \quad (7)$$

Em que,

\bar{x}_{Is} = média ponderada da incidência solar no prédio;

Is = valor da incidência solar por superfície;

As = área da superfície.

3.2.4.3 Modificações na rotina

As modificações propostas, principalmente, a relacionada na construção da forma da edificação, teve como objetivo, dar mais liberdade de criação para o algoritmo, na etapa anterior o formato foi restringido, a fim de, equiparar softwares distintos, para que a comparação fosse mais igualitária, assim, as diferenças surgiriam apenas nos critérios analisados, sem que algum fator desconhecido influenciasse os resultados.

Desse modo, com a nova estrutura da rotina definida, o seguimento dessa etapa foi a implementação de todas as alterações, utilizando, novamente, dos recursos da programação visual.

3.2.5 Análise de resultados

Com a rotina finalizada, foi possível partir para a quinta, e última, etapa do trabalho, essa etapa consistiu, primeiramente, na avaliação dos resultados produzidos, utilizando-se dos mesmos critérios adotados na etapa de comparação dos softwares, em seguida, foi realizado uma análise, utilizando todos os resultados produzidos, com foco em responder os objetivos definidos neste trabalho.

4 RESULTADOS

4.1 VALIDAÇÃO DOS SOFTWARES

O ponto de partida para construção da rotina em ambos os softwares foi construir a base do cilindro, de forma que essa base fosse paramétrica ao raio do círculo. Assim, o raio serviria como uma variável de entrada para os processos de iteração de otimização. A outra variável a ser otimizada era a altura, contudo, ela não poderia ser classificada como variável de entrada, pois há uma restrição no volume que o cilindro pode ter, logo, a altura teve que ser construída de forma que ficasse relacionada ao raio. Por fim, com ambas as variáveis definidas, realizou-se a extrusão da base, criando assim, um sólido, e como o problema pedia que o cilindro não tivesse a superfície do topo, necessitou-se realizar a extração apenas das superfícies da base e do corpo do cilindro, obtendo com isso, a quantidade de material que precisaria ser minimizada, no mais, foi rodar a otimização do software e checar os resultados.

Na parte do cálculo manual, utilizando-se da fórmula do volume de um cilindro, foi calculada a relação que o raio possuía com a altura do objeto, respeitando a restrição de volume informado no problema original. Na sequência, com as fórmulas das áreas do corpo do cilindro e da base, obteve-se a equação que representa a quantidade de material da lata, ao realizar a primeira derivada dessa função, busca-se o valor ótimo do resultado igualando a zero, após ambos os processos, foi calculada a equação resultante, encontrando assim, os valores ótimos para o raio e para a altura, as equações (8), (9) e (10) são as utilizadas no cálculo manual.

$$Volume = 500 \text{ cm}^3 = \pi * raio^2 * altura \quad (8)$$

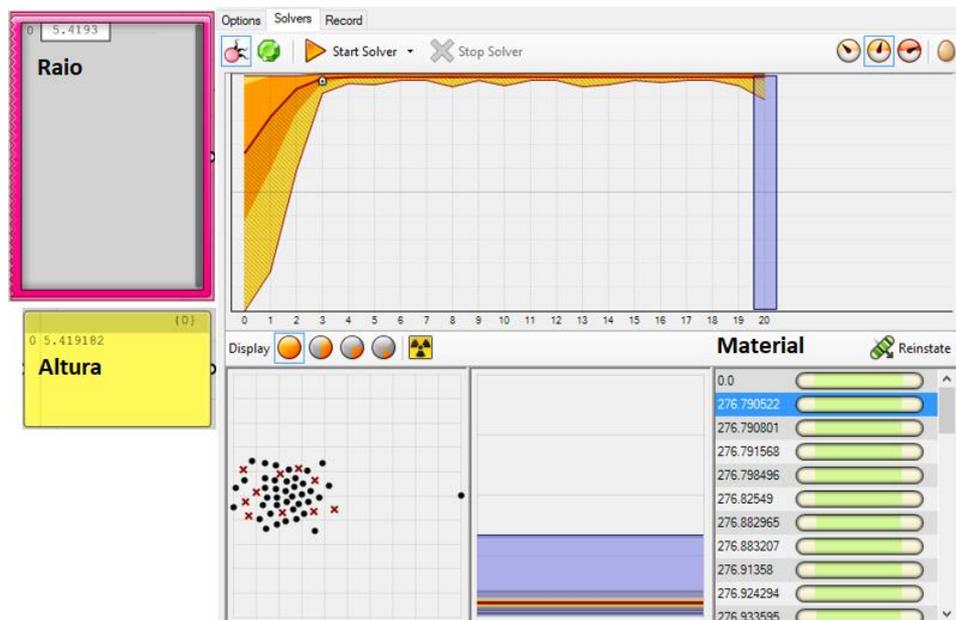
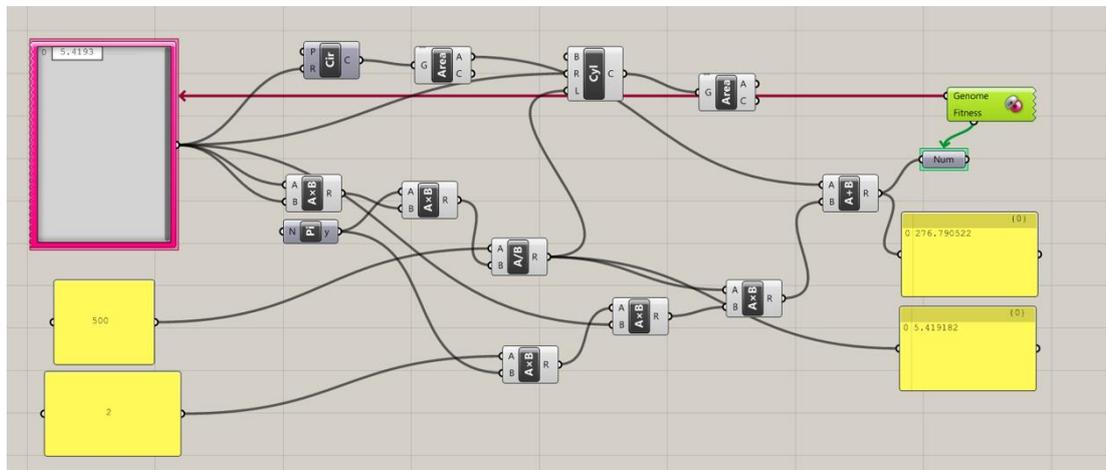
$$Material = 2 * \pi * raio * altura + \pi * raio^2 \quad (9)$$

$$M'(raio) = \frac{-1000}{raio^2} + 2 * \pi * raio = 0 \quad (10)$$

A partir das equações (8), (9) e (10), o cálculo manual para conferência do problema de otimização citado anteriormente, resultou nos seguintes valores, as duas variáveis de entrada, raio e altura, precisam ter 5,419 cm para que se consiga confeccionar a lata com a menor quantidade de material possível, sendo então, o valor ótimo de material de 276,76 cm².

O resultado da verificação através do software Grasshopper, bem como, a rotina montada para chegar a otimização do problema estão expostos na Figura 20.

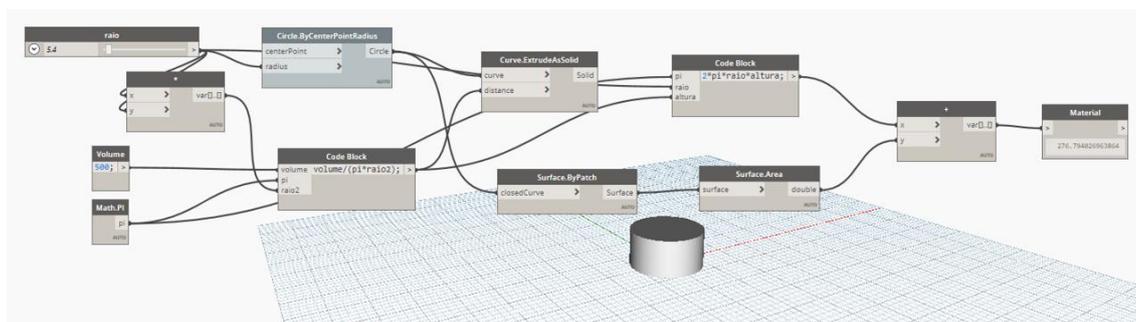
Figura 20 - Rotina e resultados Grasshopper



Fonte: Elaborado pelo autor.

Já para o caso da verificação utilizando o Dynamo, tanto o resultado como a rotina estão ilustradas na Figura 21.

Figura 21 - Rotina e resultados Dynamo



Outputs	
Altura	5.420
Material	276.791
Inputs	
Raio	5.419

Fonte: Elaborado pelo autor.

Como os resultados apresentados pelo Dynamo e pelo Grasshopper se equivalem ao obtido no cálculo manual, logo, concluiu-se a verificação dos algoritmos genéticos utilizados nesse trabalho.

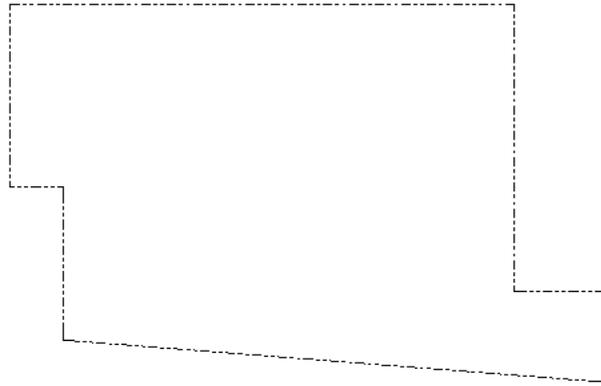
4.2 CONSTRUÇÃO DA ROTINA

A seguir, apresenta-se a estrutura lógica desenvolvida para implementação das rotinas nos softwares Dynamo e Grasshopper.

4.2.1 Terreno

O primeiro passo foi modelar o terreno que foi utilizado como referência para desenvolvimento do trabalho, o desenho foi realizado no Autodesk Revit, optou-se pela modelagem do terreno externa aos softwares Grasshopper e Dynamo pois o processo de importação do terreno para dentro da rotina era mais simplificado do que modelar o mesmo terreno em cada software. A Figura 22 ilustra os contornos do terreno.

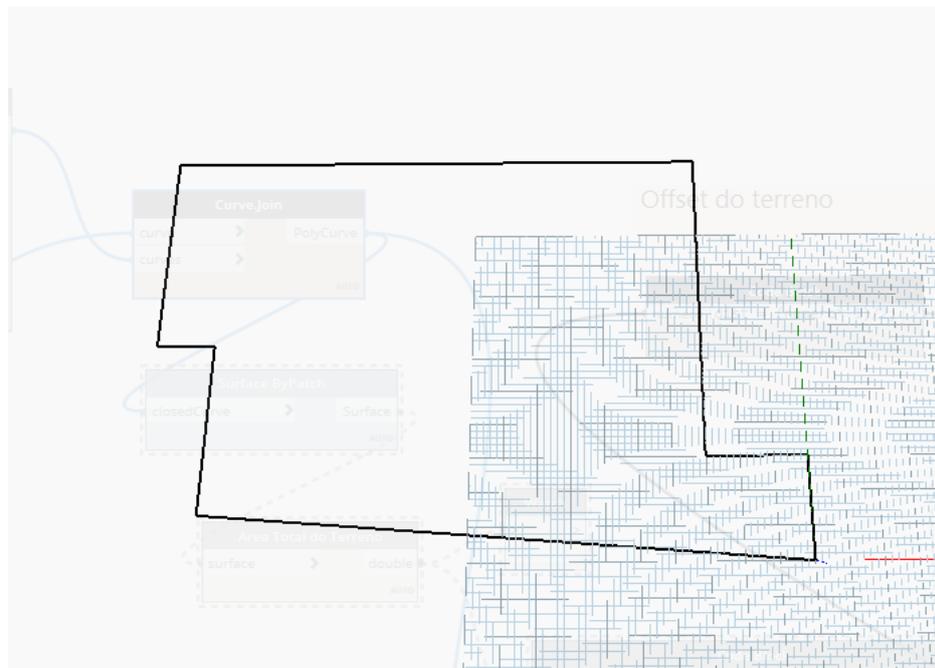
Figura 22 - Base do terreno



Fonte: Elaborado pelo autor.

Portanto, a primeira parte da rotina foi importar a geometria do terreno. A Figura 23 mostra o terreno importado.

Figura 23 - Terreno importado no software de programação visual



Fonte: Elaborado pelo autor.

De posse das informações geométricas do terreno, extraiu-se a área total do terreno para computar os valores limites do índice de aproveitamento e da taxa de ocupação. Os valores de IA e TO são variáveis na rotina, logo, podem ser alteradas pelo usuário conforme necessidade, para dar seguimento ao estudo, utilizou-se valores referência do plano diretor da cidade de Novo Hamburgo, assim, IA ficou definido como 2,4 e a TO de 75%, conforme Figura 24. Por fim, deslocou as linhas de contorno do terreno em direção ao seu interior, respeitando situações de recuos necessários por norma.

Figura 24 - TO e IA utilizados para construção da rotina

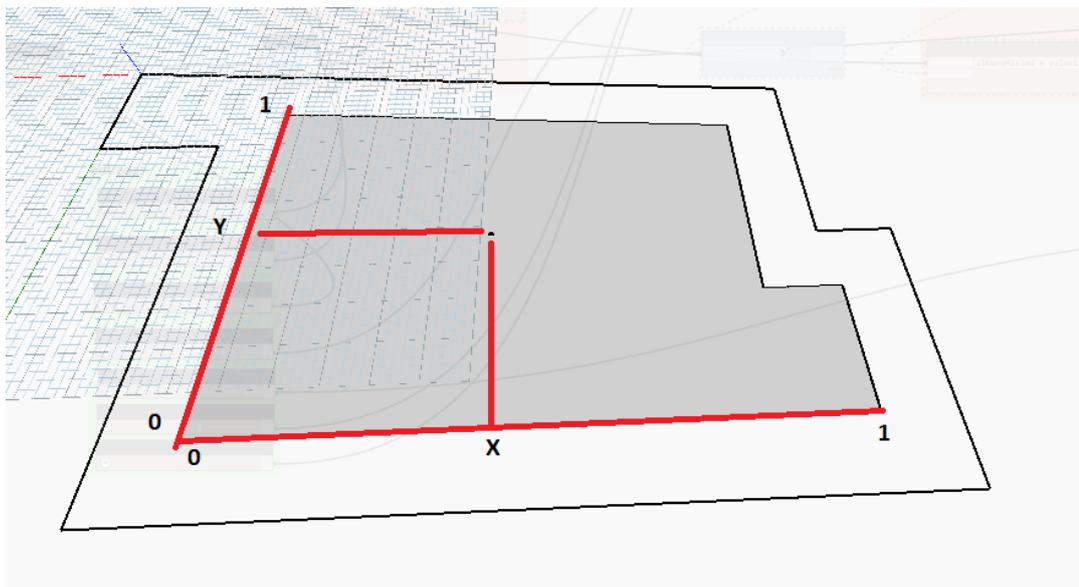
Macrozoneamento		APA			ZM														ZAP	ZI		
Regime Urbanístico	Setores	APA Norte	APA Sul	APA LG	SM1	SM2	SM3	SM4	SCC	CHHV	CC	CCS	CTT	CTR	CD	SCLG	Passo do Peão	Wallahai	Passo dos Corvos	Rotermund	ZAP	ZI
	TO	%(máx)	10	5	5	75	75	75	75	75	50	50	75	75	75	75	50	50	30	50	30	5
IA	(máx)	0.2	0.1	0.1	2	1	2,4	2	4	1	1	2,4	2,4	1	2,4	1	1	1	1	1	0.1	1

Fonte: Adaptado de Novo Hamburgo (RS) (2004).

4.2.2 Posição da torre

A lógica implementada para determinar a posição da torre no terreno foi utilizar duas variáveis de entrada, que podem ser controladas pelo algoritmo na iteração do estudo de otimização, coordenando as posições X e Y do ponto central da torre. As variáveis tinham uma faixa de valores entre 0 e 1 tornando possível acessar qualquer posição dentro das linhas deslocadas do terreno. A Figura 25 ilustra a ideia da lógica implementada.

Figura 25 - Lógica para posição central da torre



Fonte: Elaborado pelo autor.

Como é possível observar na Figura 25, devido à configuração da geometria das linhas, havia combinações de entradas em X e Y que resultavam com o ponto fora da área permitida, área com hachura cinza, portanto, necessitou-se implementar um sistema de punição nas rotinas. Adotou-se um sistema binário de pontuação, assim, caso o algoritmo determinasse que a posição da torre estivesse dentro da área cinza, recebia pontuação 1, caso as coordenadas do ponto estivessem fora da área permitida, recebia a pontuação 0, por fim, no processo de otimização do algoritmo, essa pontuação tinha como objetivo ser maximizada, devido a isso, a cada nova geração nos resultados, o algoritmo tinha cada vez menos tendência a deixar o ponto fora da área cinza.

4.2.3 Criação dos pavimentos

Com a posição central da torre estabelecida, a próxima etapa foi elaborar a lógica para criação dos pavimentos da torre. Definiu-se então que todos os pavimentos teriam a mesma área, simplificando a lógica criada, dando liberdade apenas para que cada pavimento seja rotacionado em comparação aos outros. O primeiro pavimento foi criado utilizando o nó para criação de um retângulo, com isso, foram criadas mais duas variáveis de entrada, uma variável para determinar a largura e outra para determinar o comprimento, então, com a interação dessas duas

variáveis era possível determinar o tamanho do primeiro pavimento, conseqüentemente, o tamanho que todos os pavimentos do prédio teriam. Adicionou-se ainda para esse retângulo base, uma variável de entrada para modificar sua rotação, ampliando a liberdade de criação na orientação da torre.

Para a criação dos demais pavimentos era necessário descobrir quantos pavimentos poderiam ser gerados sendo esse valor limitado pelo índice de aproveitamento. Esse cálculo seguiu na seguinte sequência, extraiu-se a área do retângulo do primeiro pavimento criado anteriormente, o valor limite do IA foi calculado pelo produto desse coeficiente pela área total do terreno. Por fim, esse valor limite foi dividido pela área do primeiro pavimento, resultando, na quantidade de pavimentos permitidos. Houve um ajuste realizado no resultado dado que foi o arredondamento para o mais próximo inteiro inferior ao resultado original, visto que, o resultado da divisão, usualmente, resultaria em um valor com casas decimais.

Com o primeiro pavimento criado e o número de pavimentos decididos, realizou-se a translação dos demais pavimentos, adotando uma distância entre pavimentos de 3 metros, como citado anteriormente, os pavimentos tinham liberdade para rotacionar, sendo assim, foi necessária a criação de mais uma variável de entrada que controlasse essas rotações.

Tendo a lógica de criação dos planos horizontais, para a criação dos fechamentos externos do empreendimento foi realizada apenas a extrusão dos planos horizontais na distância do pé-direito estabelecido, 3 metros, para finalizar a etapa responsável pela forma do edifício.

4.2.4 Custo

A última parte da estrutura da rotina é a orçamentação do projeto gerado. A lógica montada foi realizar a extração do perímetro e área de cada pavimento, com o perímetro foi multiplicado o valor do pé-direito, obtendo-se assim o valor da área para os planos verticais. Para a área de cada pavimento foi multiplicado o valor de uma variável fixa de 0,15 metros, resultando no volume necessário para os planos horizontais.

Após a coleta de dados, como explicado no tipo de orçamentação que foi utilizado no trabalho, esses valores foram multiplicados por preços de composições extraídas do SINAPI. Para os planos horizontais foi contabilizado o custo total do

item: concretagem de edificações (paredes e lajes) feitas com sistema de fôrmas manuseáveis, com concreto usinado bombeável FCK 20 MPa – lançamento, adensamento e acabamento, código 90861, custo na data referência do mês agosto de 2021 R\$ 497,78/m³. Já os planos verticais tiveram a combinação de 3 itens, um referente à alvenaria estrutural, outro para o revestimento externo e o último pelo revestimento interno, sendo eles: alvenaria estrutural de blocos cerâmicos 14x19x39, (espessura de 14 cm) - para paredes com área líquida maior ou igual que 6 m², sem vãos, utilizando palheta e argamassa de assentamento com preparo em betoneira, código 89284, custo de R\$ 61,03/m²; emboço ou massa única em argamassa traço 1:2:8, preparo mecânico com betoneira 400 L. aplicada manualmente em panos de fachada com presença de vãos, espessura de 25 mm, código 87775, custo R\$ 48,64/m²; massa única, para recebimento de pintura, em argamassa traço 1:2:8, preparo mecânico com betoneira 400 L, aplicada manualmente em faces internas de paredes, espessura de 20 mm, com execução de taliscas, código 87529, custo total R\$ 29,01/m², respectivamente.

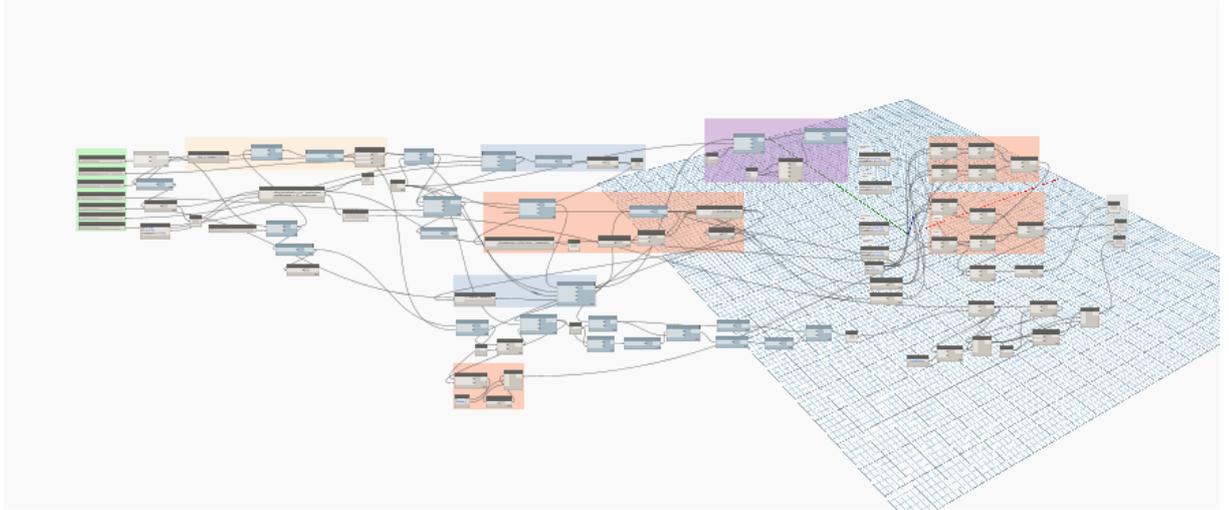
Por fim, todos os valores foram somados, obtendo assim, o custo total da edificação, esse é o valor quantificável necessário para realizar a otimização pelo algoritmo, sendo então, configurado para minimizar o custo.

Todas as variáveis fixas, como o valor do pé-direito, itens do SINAPI, valores de IA e TO e espessura do plano horizontal foram escolhidas pelo autor para dar seguimento à construção das rotinas, na tentativa de ser o mais lógico possível, contudo, são variáveis alteráveis podendo assumir qualquer valor que o usuário deseje.

4.2.5 Rotinas

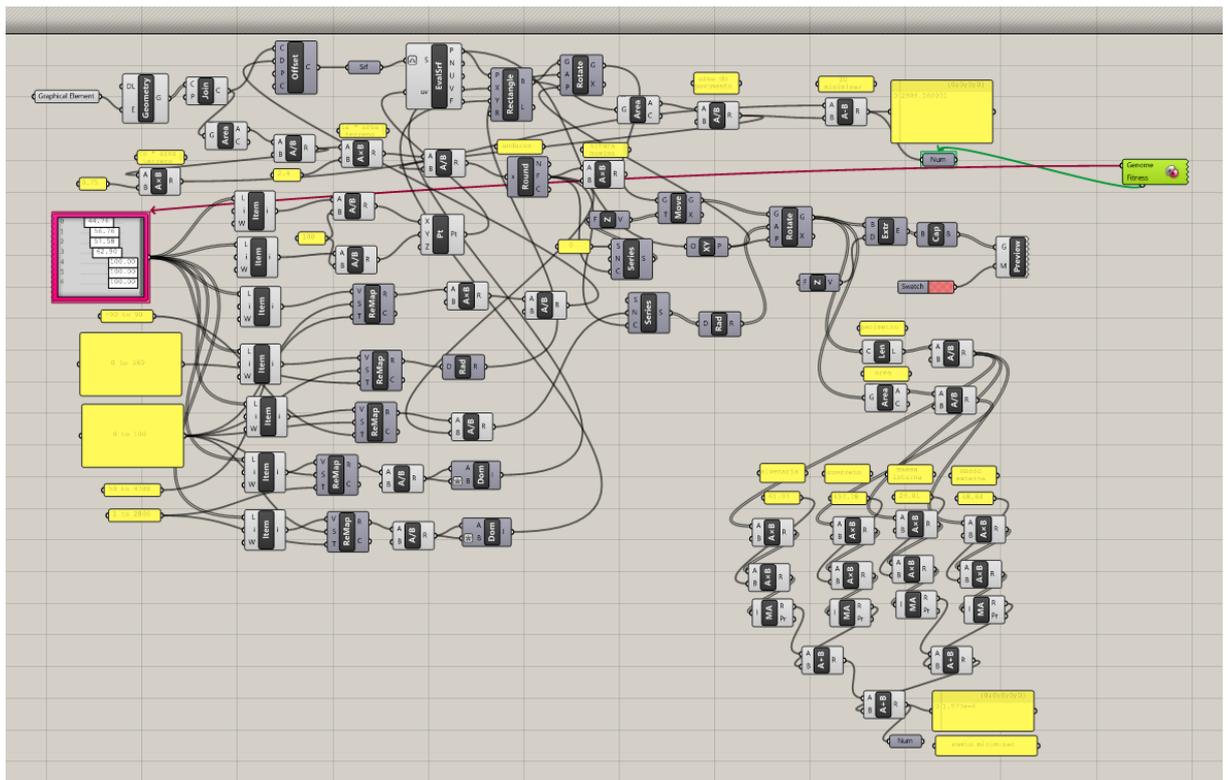
Como resultado das lógicas explicadas anteriormente, tem-se, então, as rotinas, em programação visual, desenvolvidas em ambos os softwares. A Figura 26 apresenta a rotina desenvolvida no software Dynamo e a Figura 27, a rotina construída no Grasshopper.

Figura 26 - Rotina Dynamo



Fonte: Elaborado pelo autor.

Figura 27 - Rotina Grasshopper



Fonte: Elaborado pelo autor.

4.3 COMPARAÇÃO ENTRE OS SOFTWARES

O primeiro critério de avaliação contabilizado foi o tempo de duração do processamento para obtenção dos resultados da otimização. Começando pelo Dynamo, a rotina teve sua execução iniciada às 19h45, do dia 11 de outubro de

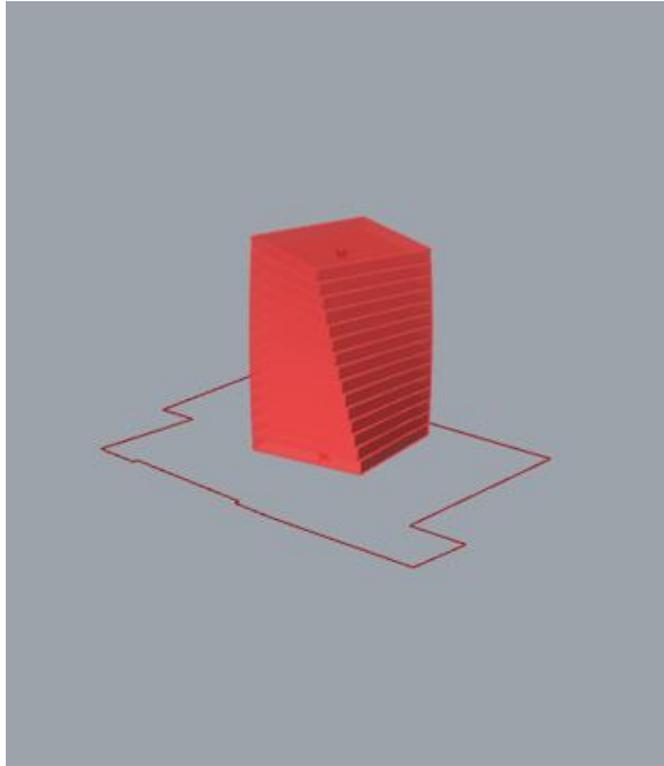
2021, e sua conclusão se deu às 21h17, do mesmo dia, totalizando assim, um tempo de duração de 1h32.

Em contrapartida, a rotina no Grasshopper teve sua execução iniciada às 20h43, do dia 16 de outubro de 2021, finalizando seu processamento às 21h06, do mesmo dia, totalizando 23 minutos de duração. Ao considerar os dois resultados obtidos, nesse primeiro critério, o Grasshopper performou melhor que o Dynamo por uma diferença de 1h09.

Ao avaliar a quantidade de variáveis otimizadas por cada software, o Dynamo levou vantagem, pois o nó de otimização Galapagos, pertencente ao Grasshopper, permite que uma variável seja otimizada por vez, por outro lado, o Dynamo otimizou, ao mesmo tempo, todas as variáveis testadas.

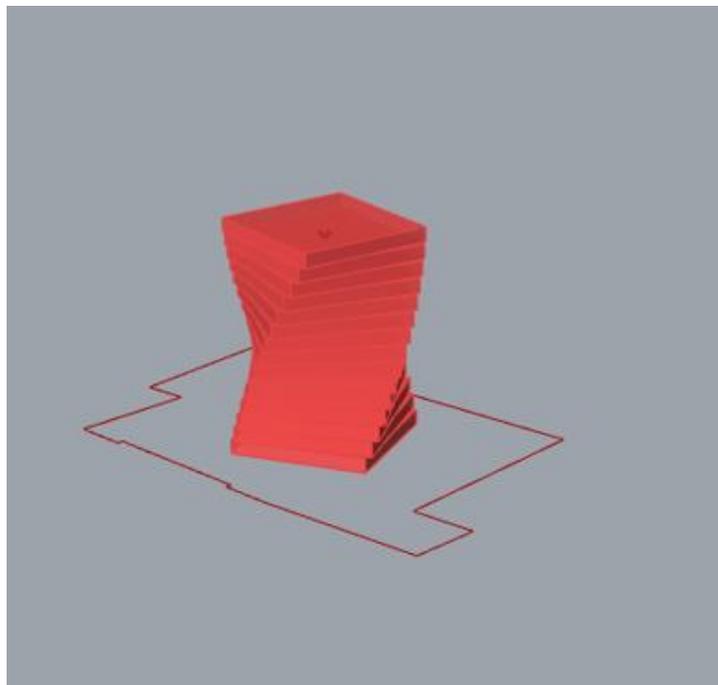
O último critério avaliado foi a qualidade dos resultados produzidos. Quanto ao Grasshopper, como apenas uma variável era otimizada por vez, escolheu-se dois resultados, um quando a variável custo foi otimizada, apresentando assim, o menor custo possível, exposto pela Figura 28, outro quando otimizou-se a taxa de ocupação, obtendo assim, o menor valor possível para a TO, mostrado na Figura 29, os outros melhores resultados produzidos foram apenas derivados dos que foram escolhidos, sendo as diferenças entre eles, apenas os valores de entrada que não modificavam o resultado das variáveis otimizadas, como: a rotação da torre, a posição da torre no terreno e a rotação dos andares.

Figura 28 - Resultado do custo otimizado



Fonte: Elaborado pelo autor.

Figura 29 - Resultado da TO otimizada

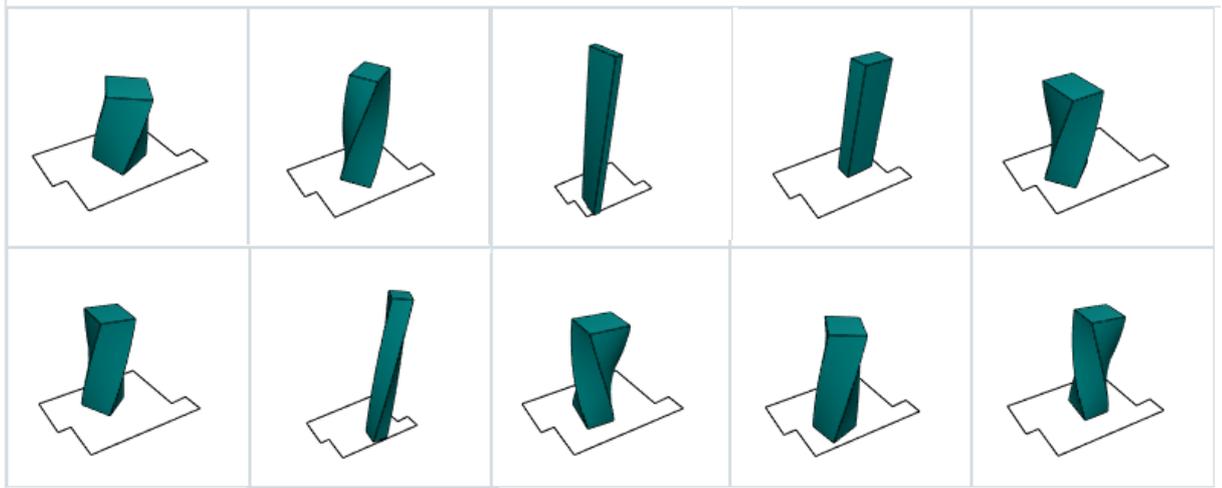


Fonte: Elaborado pelo autor.

Já para o caso do Dynamo, ao otimizar, ao mesmo tempo, todas as variáveis, produziu-se diversas soluções diferentes, devido a como cada variável era otimizada

para atingir a frente ótima de Pareto, como mostra na Figura 30, soluções divergentes entre si.

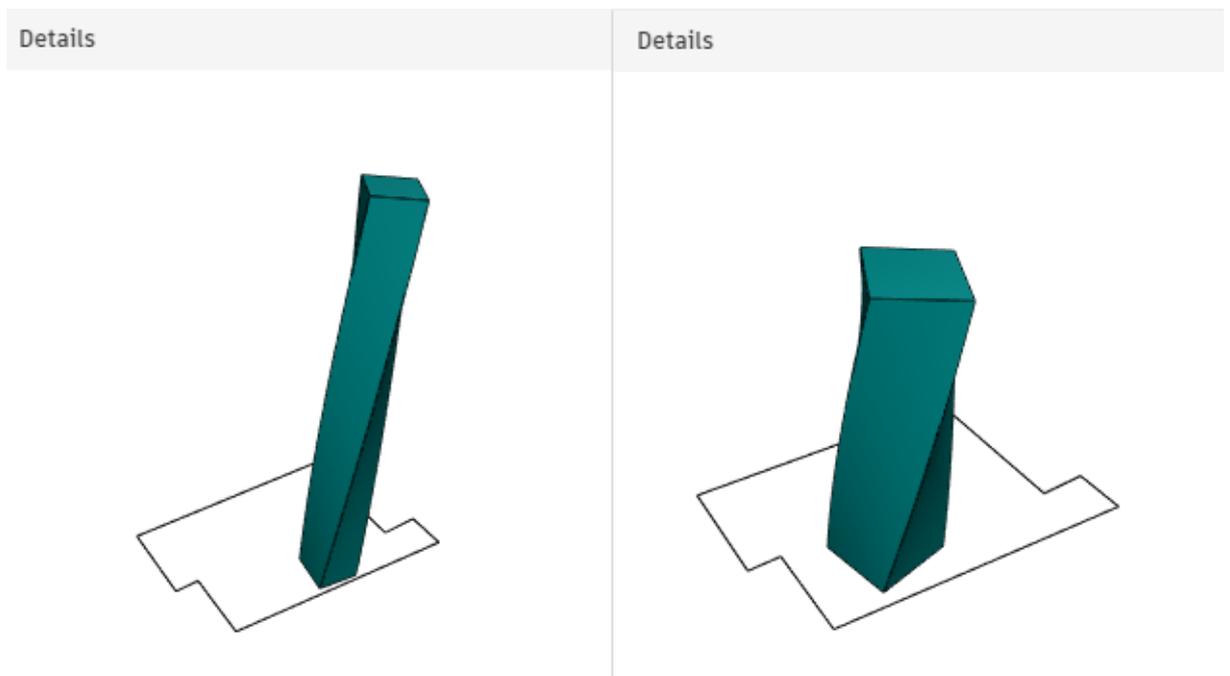
Figura 30 - Resultados gerados no Dynamo



Fonte: Elaborado pelo autor.

Assim, o critério de desempate para escolha dos resultados a serem apresentados, foi a preferência arquitetônica do autor do trabalho, a Figura 31 mostra as soluções escolhidas.

Figura 31 - Soluções escolhidas



Fonte: Elaborado pelo autor.

Por fim, para efeitos comparativos, foi elaborado o Quadro 3, a solução do Grasshopper que teve seu custo otimizado, recebeu a nomenclatura de G1, a solução que teve TO otimizada no Grasshopper, denominada G2, já para as soluções do Dynamo, a solução à esquerda da Figura 31, denominada de D1, por consequência, a da direita de D2.

Quadro 3 - Comparativo entre as soluções escolhidas

	G1	G2	D1	D2
TO	1954,84	1433,61	1893,19	1770,47
Custo (R\$)	1505800,42	2003718,57	2139130,33	1807277,70

Fonte: Elaborado pelo autor.

Com o Quadro 3, foi possível perceber que as soluções vindas do Grasshopper possuem os melhores valores nas variáveis que estavam sendo otimizadas, contudo, a outra variável por não estar sendo considerada na otimização, acaba obtendo um valor pior. Em contrapartida, os resultados do Dynamo possuem valores intermediários quando comparados aos obtidos pelo Grasshopper, indicando os efeitos dos *trade-offs* realizados pela otimização multiobjetivo para alcançar a frente ótima de Pareto.

Como, no geral, o comparativo não indicou nenhuma ampla vantagem de um software para o outro, ao analisar que ambos produziram soluções que satisfazem ao objetivo de produzir um resultado que atenda os requisitos de aplicação de um projeto arquitetônico em fase de concepção. Todavia, ao considerar que o Dynamo gerou resultados otimizando duas variáveis simultaneamente, como visto na literatura anteriormente, isso acaba influenciando no tempo de processamento, a vantagem de tempo de processamento que o Grasshopper obteve sobre o Dynamo, não pode ser considerado como um fator decisivo, pois não é uma comparação em critérios de igualdade. Sendo assim, por possuir vantagem sobre o Grasshopper na análise multiobjetivo, o Dynamo foi considerado como a melhor opção para aplicação do método de design generativo no cenário considerado neste trabalho.

Ainda, vale ressaltar que tiveram questões que não foram consideradas formalmente, neste trabalho, para escolha do Dynamo como melhor software para uso do design generativo aplicado à fase de concepção do projeto arquitetônico, mas que foram observadas e podem ser relevantes para escolhas em outras

situações, são elas: o Dynamo possui a interface com a melhor usabilidade, é mais fácil identificar qual a função que determinado nó realizará e na criação de uma rotina, ao escolher um nó, estão descritos claramente quais são os tipos de entrada que aquele nó permite e o tipo de saída que ele produzirá, devido a isso, iniciantes ou não programadores terão mais facilidade para utilização, em contrapartida, no Grasshopper, a interface de apresentação dos resultados produzidos pelo algoritmo genético mostra informações mais significativas que as apresentadas pelo Dynamo, como a curva de otimização ao longo das gerações e as soluções categorizadas entre os melhores e piores resultados.

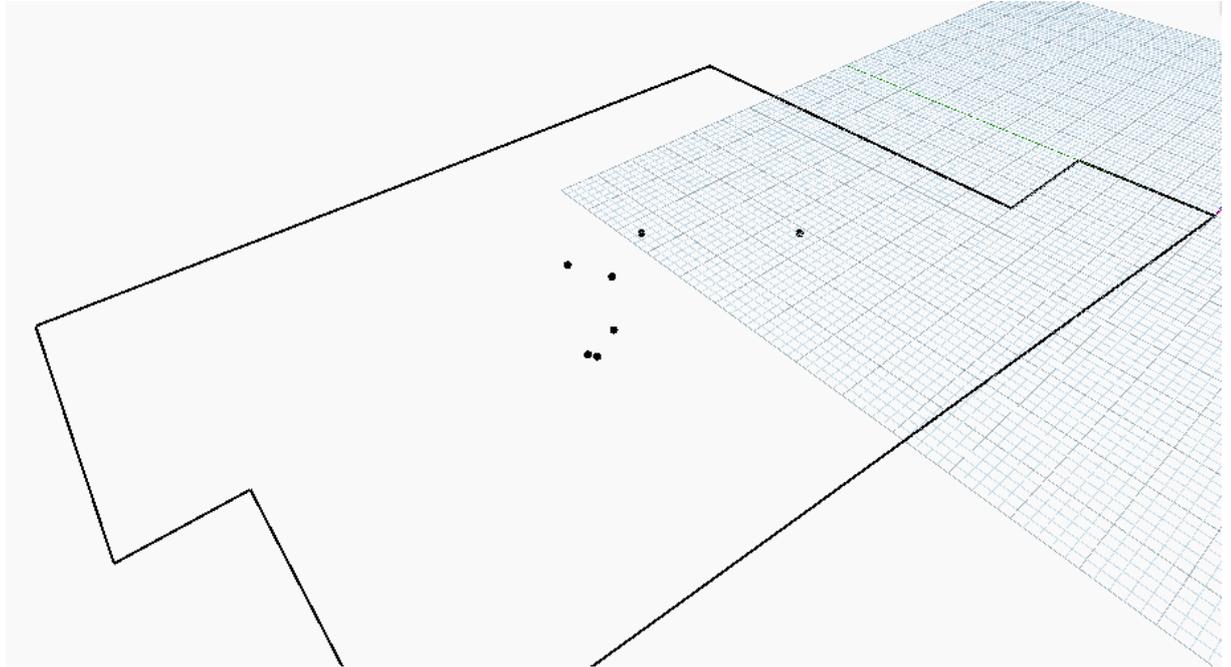
4.4 INCREMENTAÇÃO DA ROTINA NO DYNAMO

A seguir, apresenta-se as alterações realizadas na rotina do Dynamo visando testar os limites da aplicação.

4.4.1 Criação do pavimento base

A principal alteração realizada foi na construção do pavimento que serve como base para extrusão dos pavimentos seguintes na criação do prédio. A lógica de duas variáveis de entrada controlando a posição central da torre, seguindo, para criação do pavimento base com as variáveis controlando as dimensões do retângulo, foi substituída por uma lógica que utiliza até sete pontos para formar o contorno do prédio. O controle da posição dos pontos é idêntico ao utilizado na etapa anterior de construção da rotina, ilustrado pela Figura 25. Assim, por se tratar da utilização dessa vez de sete pontos, foi necessário adicionar 14 variáveis de entrada, cada dupla controlando as posições X e Y de cada ponto dentro dos limites do terreno. A Figura 32 mostra os pontos posicionados no terreno.

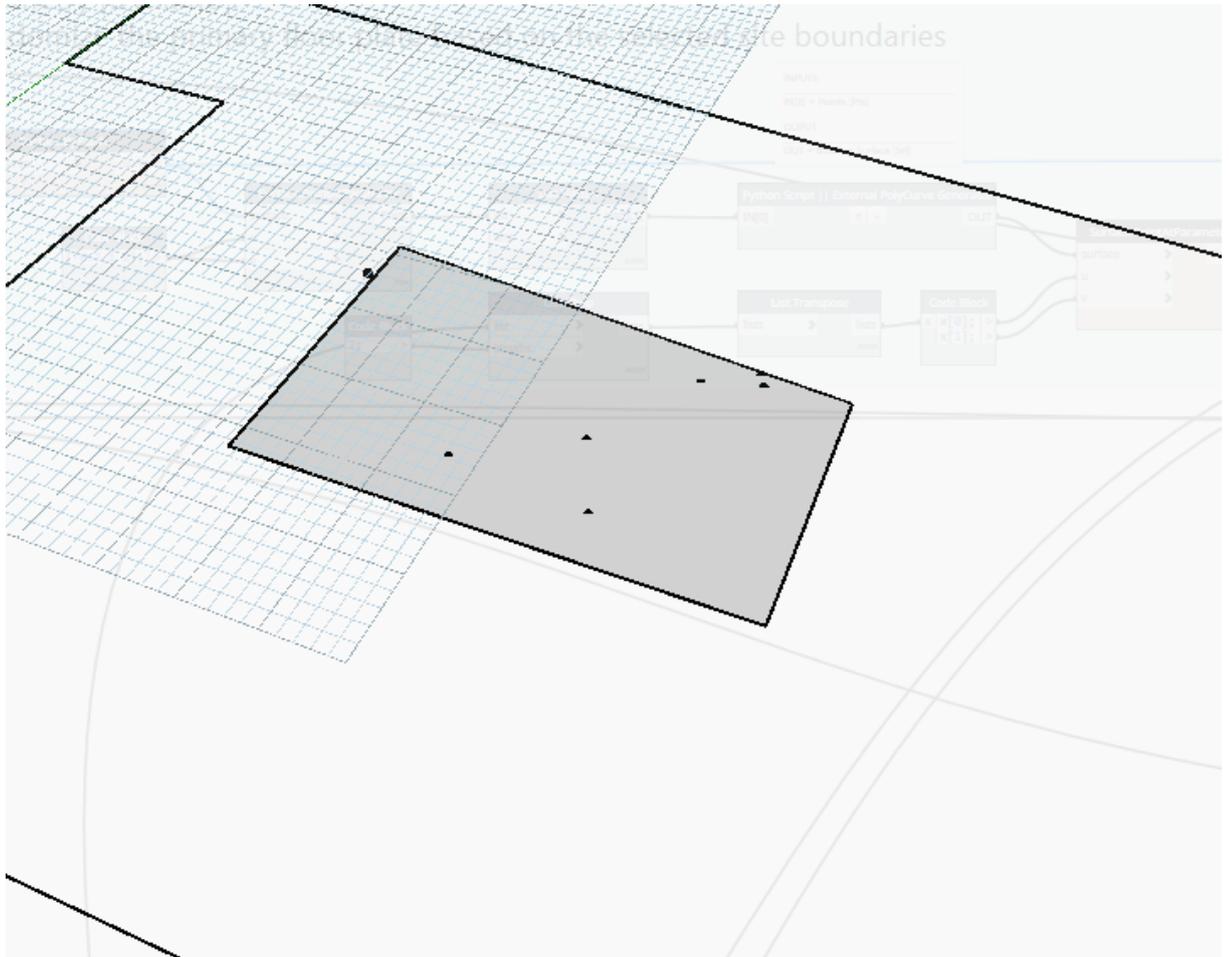
Figura 32 - Pontos para criação da geometria do pavimento base



Fonte: Elaborado pelo autor.

A segunda parte de alterações foi a retirada do método de punição do algoritmo, caso a torre fique posicionada fora dos limites do terreno. Em contrapartida, a lógica adicionada, foi utilizar as linhas que tinham sido deslocadas do contorno do terreno para dentro como posição máxima do contorno do pavimento. A estrutura da rotina que se seguiu foi de extrair a superfície total formada por essas linhas deslocadas, como mostra a Figura 33.

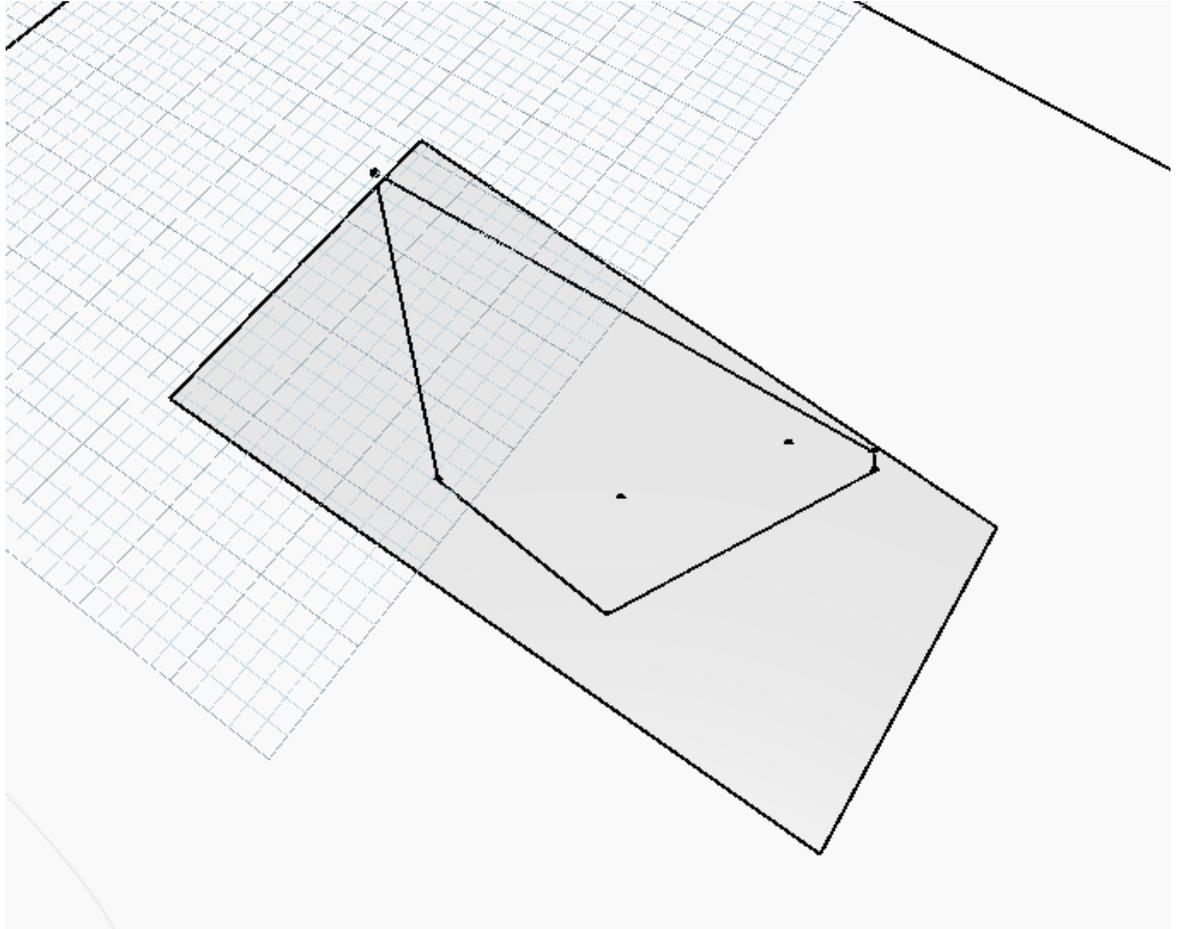
Figura 33 - Superfície total



Fonte: Elaborado pelo autor.

A partir das coordenadas de cada ponto, criou-se uma lista daqueles que possuíam posição mais externas entre eles. Em seguida, os pontos mais externos foram conectados, caso um dos pontos estivesse posicionado fora dos limites das linhas deslocadas, o segmento de reta que iria até o ponto é interrompido ao tocar o limite demarcado, como ilustrado pela Figura 34.

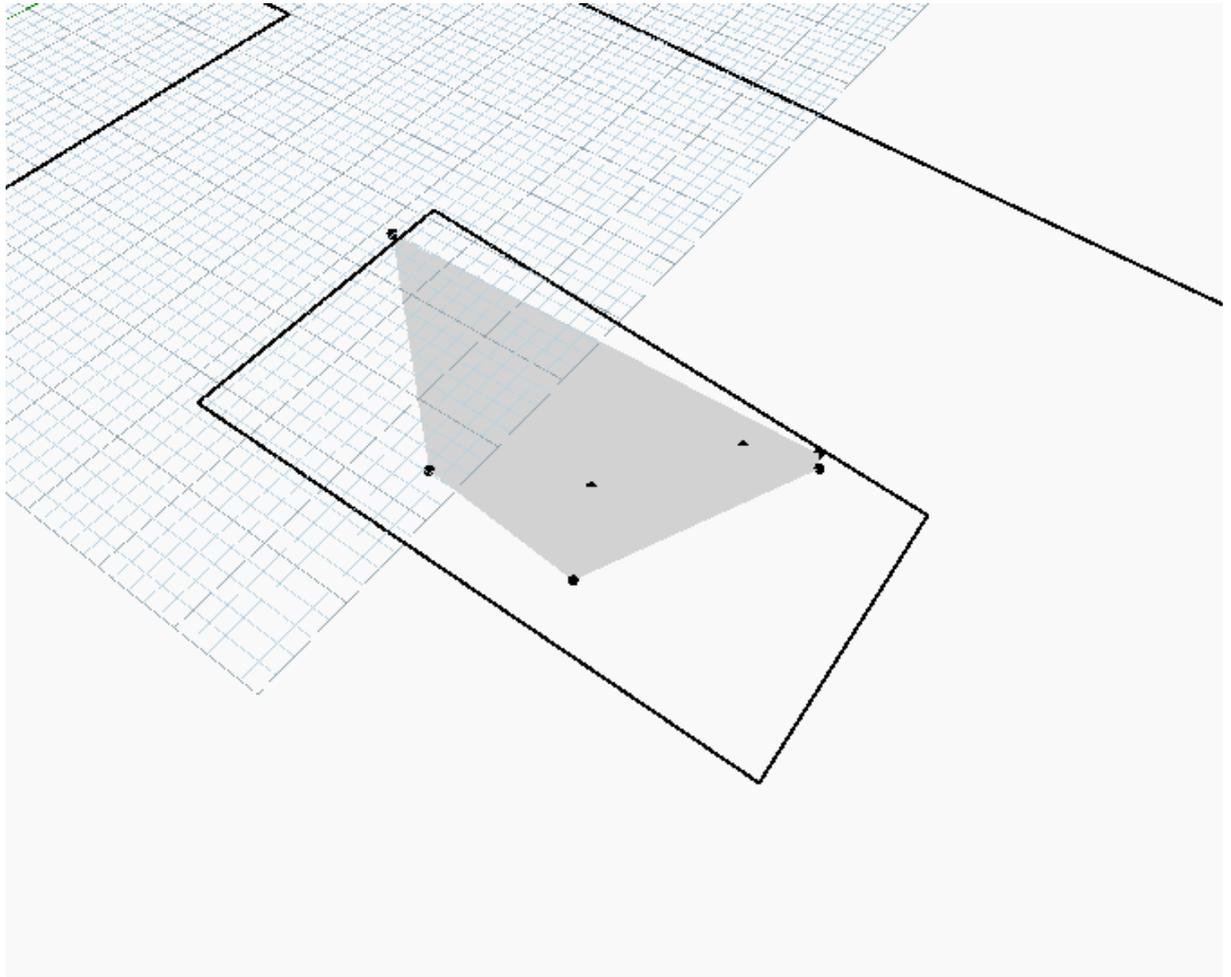
Figura 34 - Criação do contorno do pavimento base



Fonte: Elaborado pelo autor.

Por fim, foi realizada uma subtração entre a superfície total pelo contorno formado com o conjunto de linhas que ligam os pontos, resultando, na superfície que representa o pavimento base do projeto, como mostrado na Figura 35.

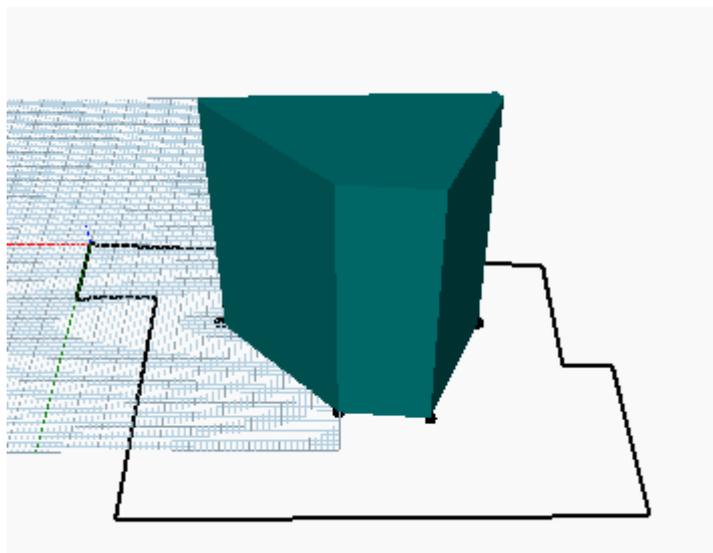
Figura 35 - Pavimento base



Fonte: Elaborado pelo autor.

Além das modificações relatadas, a estrutura da rotina, no que está relacionado à criação da estrutura do prédio, se manteve igual ao que já tinha sido criado, portanto, o que se seguiu após a criação do pavimento base, foi a criação dos outros planos horizontais através da translação desse pavimento criado. Por último, o fechamento do prédio com a criação dos planos verticais, gerando assim, o sólido total do prédio, como ilustrado na Figura 36.

Figura 36 - Resultado do sólido do prédio



Fonte: Elaborado pelo autor.

4.4.2 Parâmetros de projeto

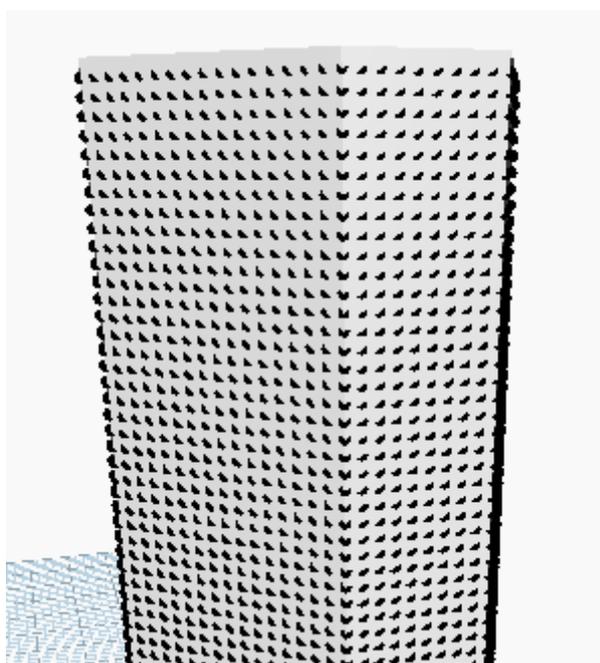
As últimas alterações realizadas na rotina foram as adições do cálculo do índice econômico de compactidade e o de incidência solar.

No caso do índice econômico de compactidade, a estrutura criada na rotina para obtenção do resultado do leC foi, primeiramente, extrair a quantidade de arestas que o prédio possuía e o valor do perímetro, podendo então, calcular o perímetro econômico do projeto. Na variável de perímetro das paredes exteriores curvas, foi adotado o valor 0, visto que, na rotina o contorno do prédio é um conjunto de segmentos de retas, utilizando-se do valor calculado referente ao perímetro econômico do projeto, juntamente, com a área do projeto, informação que já fora extraída previamente, calculou-se o índice econômico de compactidade. Por fim, com o resultado gerado sendo um percentual, fazendo assim um resultado quantificável, foi possível configurar essa variável para otimização do algoritmo, buscando maximizar esse valor.

Já para a variável de incidência solar, foram utilizados recursos nativos dispostos no Dynamo para estudo solar, necessitando apenas, informar as datas de início e fim que seriam considerados para o estudo. Nesse trabalho, configurou-se pelo período de um ano, 01 de janeiro até 31 de dezembro, a fim de, englobar todas as posições solares possíveis, informar a posição, em coordenadas de latitude e longitude, do local que seria realizado o estudo, para essa variável foi adicionada

uma posição aleatória da cidade de Novo Hamburgo – RS, sendo os valores adquiridos via Google Maps. Assim, os valores utilizados, em graus decimais, foram -29,68, para latitude; e -51,12, para longitude. Por fim, as superfícies consideradas para a exposição solar foram os planos verticais externos para o estudo. Para o cálculo da incidência solar cada superfície estudada recebe pontos distribuídos uniformemente, esses pontos são avaliados frente à exposição solar, durante o período de tempo configurado, que sofrem, resultando assim, no valor acumulado de incidência solar. A Figura 37 mostra os pontos distribuídos ao longo das superfícies.

Figura 37 - Pontos de análise solar



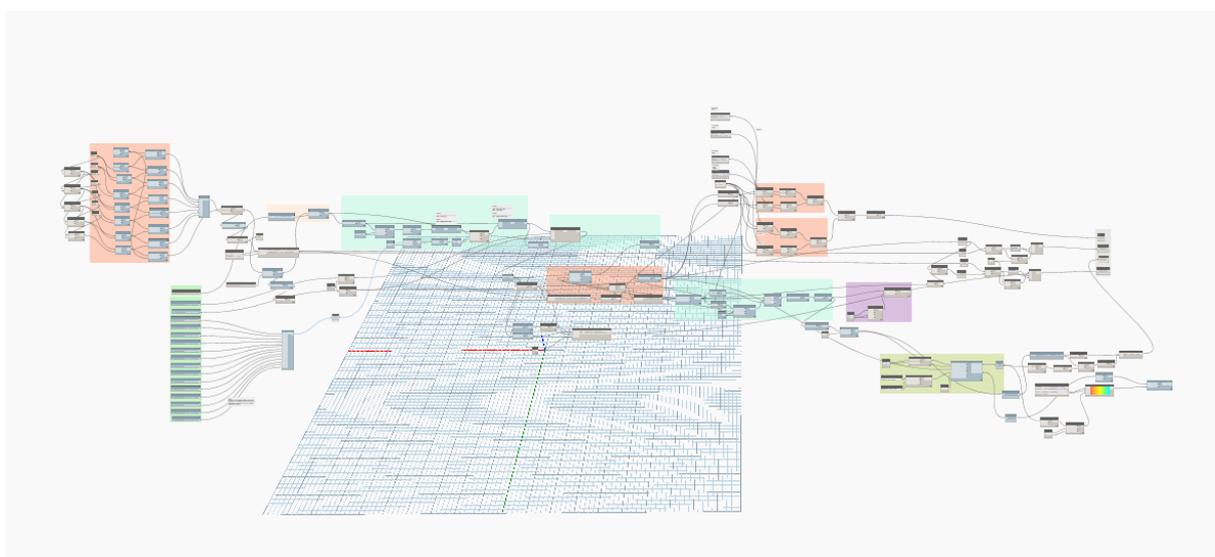
Fonte: Elaborado pelo autor.

Tendo o valor cumulativo de cada ponto, o processo que se seguiu para obter apenas um valor quantificável para otimização foi realizar uma somatória de todos os valores dos pontos que estavam na mesma superfície, obtendo, o valor de exposição geral daquela superfície. Por fim, foi realizada uma média ponderada dos valores de incidência solar em relação à área das superfícies analisadas, resultando, no valor de incidência solar média que o prédio está exposto. Como esse valor atende aos critérios para poder ser otimizado, foi configurado tendo como meta maximizar esse valor de exposição.

4.4.3 Rotina final

Após as implementações realizadas, a visão geral da rotina pode ser visualizada na Figura 38.

Figura 38 - Rotina final desenvolvida no Dynamo

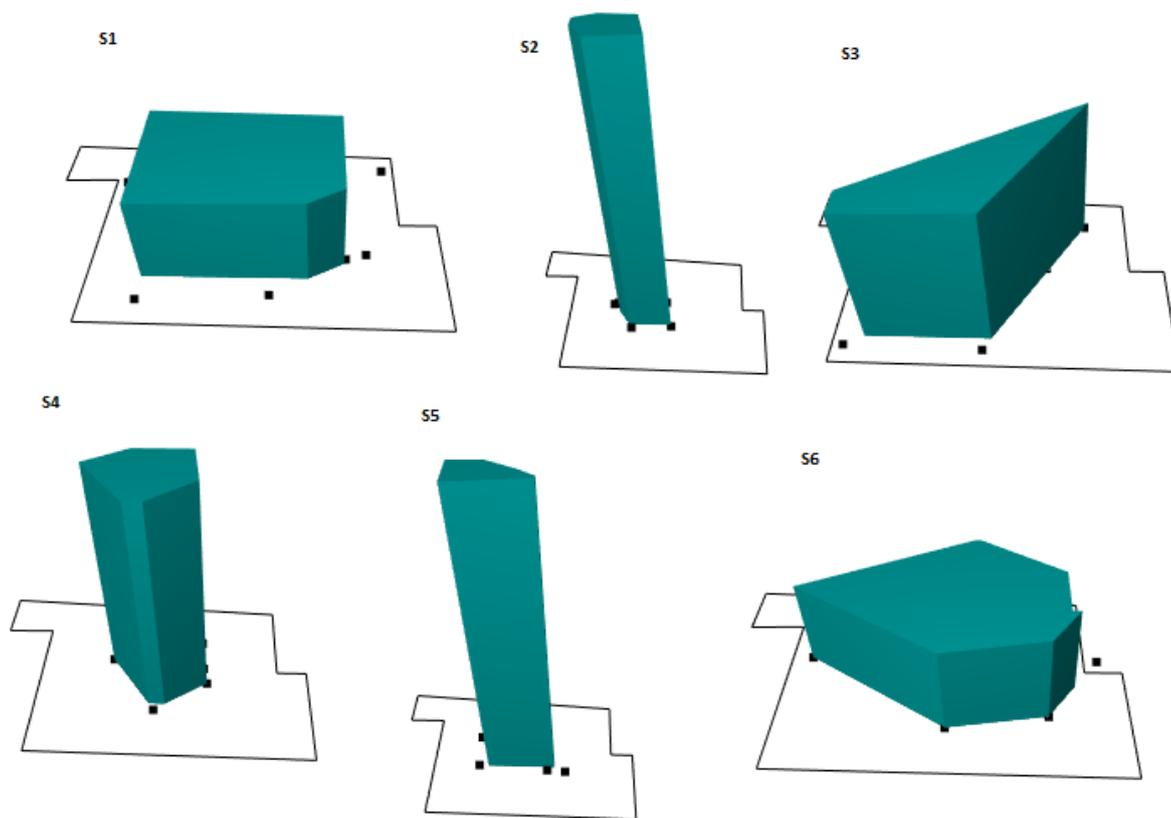


Fonte: Elaborado pelo autor.

4.5 ANÁLISE DOS RESULTADOS

A rotina final apresentou uma gama de resultados abrangentes. As soluções apresentaram características bastante divergentes entre si, mostrando desde edifícios baixos com geometria quadrada até edifícios altos com geometria complexa. A Figura 39 apresenta um compilado dos resultados produzidos.

Figura 39 - Resultados da rotina final



Fonte: Elaborado pelo autor.

Utilizando os mesmos critérios quando os softwares foram comparados tem-se, no quesito de velocidade de processamento, um aumento quase dobrado de tempo necessário para produzir todos os resultados, considerando que a rotina foi iniciada no dia 23 de outubro de 2021 às 21h18, e completou seu processo apenas às 0h13 do dia seguinte, totalizando 2h55. Obtem-se então, um acréscimo de 1h23 quando comparado ao tempo levado pelo Dynamo para otimizar a primeira rotina. Esse aumento pode ser justicado, visto que, a rotina final precisou otimizar duas variáveis a mais que a anterior e possuía uma lógica de construção de geometria diferente.

Ao avaliar o critério de otimização multiobjetivo, o Dynamo conseguiu produzir soluções para projeto arquitetônico na fase de concepção, otimizando, simultaneamente, quatro variáveis (TO, custo, incidência solar e IEC).

Por fim, o Quadro 4 apresenta os resultados das variáveis utilizadas referentes às soluções apresentadas na Figura 39 anteriormente, nomeadas de S1 até S6, seguindo a ordem da esquerda para direita, de cima para baixo, apresentado

na Figura 39, como forma de destacar os resultados, os melhores resultados, em cada variável, estão com a cor azul e os piores em amarelo.

Quadro 4 - Resumo dos resultados

Variáveis	S1	S2	S3	S4	S5	S6
Custo (R\$ em milhões)	1,4921	2,2969	1,8785	1,9247	2,4704	1,6026
TO	2146,446	3381,692	2804,813	3158,77	3376,839	2030,426
Incidência Solar (kW/m ² .ano)	441,54	943,03	968,11	669,93	1355,80	403,86
leC (%)	86,735	85,094	67,862	87,981	74,679	83,61

Fonte: Elaborado pelo autor.

Explorando os resultados apresentado no Quadro 4, verificou-se que das seis soluções escolhidas, quatro soluções obtiveram valores de leC na região de achatamento da curva, próximos ao valor ótimo, 88,6%, apontado por Mascaró (2010).

Considerando as métricas de cálculo para os demais parâmetros de projeto otimizados, constatou-se que há coerência nos valores apresentados. Na variável TO, as soluções que apresentaram melhor desempenho são aquelas que possuem o pavimento tipo de maior área, ocupando maior parte do terreno. Por consequência, se tornam mais baixos que as demais soluções. No custo, as soluções de melhor desempenho são aquelas que apresentam uma menor altura total da edificação. Em contrapartida, nos valores obtidos de incidência solar, as melhores soluções são aquelas que apresentam uma altura maior e, na sequência, uma maior área de fachada.

Ainda, é possível observar que nenhuma das soluções possui todos os valores melhores otimizados que as outras soluções, há sempre uma compensação entre as variáveis, situação esperada visto o funcionamento das otimizações buscando atingir a frente ótima de Pareto. Foi identificado que essa situação gerada, quando se tem uma otimização multiobjetivo, causa um leque de soluções que podem ser escolhidos para uso, todos tendo suas vantagens e contrapartidas quando comparados aos outros. Devido a isso, em aplicações reais, haverá a

necessidade de um projetista responsável por decidir qual será a solução escolhida, dado que, provavelmente, o critério que será utilizado como desempate são valores que ainda não podem ser quantificáveis, como as questões estéticas e as preferências arquitetônicas do projetista, por exemplo.

Isso é reforçado na literatura por Nagy et al. (2017) e Rorhmann (2019) que concluem em seus trabalhos que os aspectos mais voltados para subjetividade, como a forma do edifício e a beleza, são conceitos difíceis de serem traduzidos para regras quantificáveis. Assim, há a necessidade dos projetistas de complementar o trabalho desenvolvido pela máquina, colaborando na interação homem-máquina para desenvolver projetos que terão alto desempenho e serão esteticamente agradáveis.

4.5.1 Limites de aplicação da metodologia

O uso do design generativo se mostrou uma ferramenta satisfatória, contudo, o processo possui algumas limitações. A principal barreira para uma maior utilização, atualmente, é a necessidade de saber lógica de programação, ainda que a programação visual ajude na construção de toda a rotina. Problemas que são triviais para uma pessoa, como perceber se uma geometria está dentro da outra, exemplo aplicado para identificar se o edifício estava dentro do terreno, necessitam de uma estrutura lógica complexa, pois o computador não tem essa forma de percepção. Outro ponto, é na escolha das variáveis a serem otimizadas, dependendo da complexidade em se traduzir para um valor quantificável, haverá necessidade de utilizar linhas de código, ou seja, ter conhecimento de uma linguagem de programação para construir o cálculo da variável escolhida.

Outro ponto a se ressaltar, é da necessidade de total compreensão do problema que se deseja resolver. Aplicar design generativo envolve utilizar principalmente matemática e geometria. Caso os conceitos sobre o que se deseja otimizar não estejam totalmente claros e corretamente traduzidos na rotina para entendimento do algoritmo, a máquina, ao realizar a otimização, não tem poder de compreensão se aqueles resultados que estão sendo gerados são significantes para o problema pensado. Devido a isso, caso haja algum problema na lógica elaborada, os resultados produzidos não terão valor algum, como mostra a Figura 40, em um dos primeiros testes realizados neste trabalho.

Figura 40 - Problemas no resultado gerado



Fonte: Elaborado pelo autor.

Sendo assim, um erro nos resultados gerados levará a todo um processo de retrabalho, tornando o processo ainda mais demorado e custoso, transformando assim, em uma questão limitante para algumas aplicações.

4.5.2 Avaliar a capacidade da aplicação desenvolvida

Ao ponderar todo o processo realizado, de criação de rotinas e produção de soluções aplicando o método de design generativo, percebe-se que ter a ideia clara de quais parâmetros de projetos serão otimizados é essencial, dado que, há uma quantidade ilimitada de parâmetros pelos quais um projeto pode ser avaliado, e toda a aplicação será construída com foco em responder esses parâmetros.

Devido a essa necessidade de escolha em um leque ilimitado de possibilidades, a criação de uma aplicação de design generativo, de modo geral, pode ser considerada singular para cada conjunto de parâmetros adotados. Ao olhar para o trabalho desenvolvido neste projeto, mesmo visando atender questões de criação de projetos arquitetônicos voltados para fase de concepção de uma forma genérica, quando ocorre a decisão dos parâmetros de projeto que seriam utilizados, a aplicação torna-se em uma aplicação específica para aquele conjunto de parâmetros escolhidos.

Assim, ao avaliar os resultados produzidos, a aplicação desenvolvida conseguiu fornecer soluções suficientemente adequadas para atender as necessidades da fase de concepção, criando volumetrias que ao mesmo tempo, por passarem por um processo de otimização, possuíam os melhores valores para os critérios estabelecidos. Entende-se que as volumetrias geradas possam auxiliar o projetista no desenvolvimento do projeto para as próximas fases, a partir da escolha de uma solução, dentre várias possibilidades que atendem aos requisitos estudados.

5 CONCLUSÃO

Como conclusão para a pesquisa apresentada, é possível afirmar que na comparação dos softwares personalizáveis analisados, ambos possuem condições de atender ao método de design generativo. Contudo, devido a suas particularidades, os resultados apontam que o Grasshopper apresenta ser mais indicado para resolver otimizações pontuais, quando há apenas uma variável para ser otimizada, já o Dynamo, por conseguir otimizar múltiplas variáveis simultaneamente, torna-se mais indicado para problemas mais complexos, quando há a necessidade de avaliar concomitantemente um conjunto de parâmetros de projeto.

Em relação à aplicação proposta em responder determinados requisitos adotados para produzir resultado de projeto arquitetônicos voltado à fase de estudo de concepção, o trabalho, na rotina final incrementada, testou quatro variáveis a serem otimizadas e lógica com mais liberdade de criação de diferentes formas para edificação, e conseguiu produzir soluções volumétricas críveis, que poderiam ser escolhidas por um projetista, em uma aplicação real, para refino nas fases seguintes de projeto.

Portanto, considera-se que a aplicação do método de design generativo possui potencial como ferramenta auxiliar na criação de soluções para projeto arquitetônico na fase de estudo de concepção. O desenvolvimento deste tipo de aplicação permite com que o projetista foque nos conceitos e ideias que serão aplicados, dado que o trabalho mecânico de produzir os resultados ficará a cargo da máquina. Por fim, o projetista ainda terá a responsabilidade de escolher a solução final de projeto, contudo, devido ao processo de otimização realizado pelo algoritmo, essa escolha dependerá apenas dos ideais criativos do projetista, visto que, as opções ali disponíveis serão as mais otimizadas frente aos parâmetros adotados.

Para futuros trabalhos sugere-se aplicar o método de design generativo em um cenário real, comparando as soluções produzidas com projetos realizados por profissionais. Porém, ainda há necessidade de conhecimentos específicos, principalmente de programação, para construção das rotinas necessárias. Ao se considerar as muitas variáveis que podem ser adotadas em um projeto envolvendo design generativo, cada aplicação criada se torna muito específica. Devido a isso, a utilização em massa de uma mesma aplicação se torna inviável, visto que, qualquer

mudança nos conceitos do problema inicial, acarretará em mudanças na lógica montada. Considerando um cenário de aplicação real, isso acaba gerando retrabalho e tornando o processo ainda mais custoso, criando assim, uma barreira ainda para o método.

REFERÊNCIAS

- ABUSSAMRA, J. M. **O edifício residencial paulistano nos anos 70, 80 e 90: A arquitetura e o mercado imobiliário**. 2007. 254 f. Dissertação (Mestrado em Arquitetura e Urbanismo) – Universidade São Judas Tadeu, São Paulo, 2007.
- AISH, R.; WOODBURY, R. Multi-level interaction in parametric design. **Lecture Notes in Computer Science**, v. 3638, n. August, p. 151–162, 2005.
- AKELLA, R. **What Generative Design Is and Why It's the Future of Manufacturing**. Disponível em: <<https://www.newequipment.com/research-and-development/article/22059780/what-generative-design-is-and-why-its-the-future-of-manufacturing>>.
- AMERICAN INSTITUTE OF ARCHITECT CALIFORNIA COUNCIL. Integrated Project Delivery: A Guide. **American Institute of Architects**, p. 1–62, 2007.
- ANDRADE, V. A. **Modelagem dos custos para casas de classe média**. 1996. 198 f. Dissertação (Mestrado em Engenharia Civil), Universidade Federal de Santa Catarina, Florianópolis, 1996.
- ANDRADE, A. C. de; SOUZA, U. E. L. de. Críticas ao processo orçamentário tradicional e recomendações para a confecção de um orçamento integrado ao processo de produção de um empreendimento. In: SIMPÓSIO BRASILEIRO DE GESTÃO E ECONOMIA DA CONSTRUÇÃO, 3., 2003, São Carlos. **Anais...** São Paulo, 2003.
- ANTUNES, N. **Luz solar: sua importância e impacto no projeto arquitetônico**. Disponível em: <<https://blog.instacasa.com.br/luz-solar-sua-importancia-e-impacto-no-projeto-arquitetonico/>>. Acesso em: 05 out. 2021.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **NBR 12721**: avaliação de custos unitários de construção para incorporação imobiliária e outras disposições para condomínios edifícios - procedimento. Rio de Janeiro: ABNT, 2006.
- BAYER, J. et al. Migrating the classical pen-and-paper based conceptual sketching of architecture plans towards computer tools - prototype design and evaluation. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 9657 LNCS, n. January 2017, p. 47–59, 2017.
- BEASLEY, D.; BULL, D. R.; MARTIN, R. R. An overview of genetic algorithms : Part 1, fundamentals. **University Computing**, v. 2, n. 15, p. 1–16, 1993.
- CABRAL, E. C. C. **Proposta de Metodologia de Orçamento Operacional para Obras de Edificação**. 151 f. Dissertação (Mestrado em Engenharia de Produção) - Programa de Pós-graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 1988.
- CELANI, Maria Gabriela Caffarena; PUPO, Regiane Trevisan. Prototipagem rápida e

fabricação digital para arquitetura e construção: definições e estado da arte no Brasil. **Cadernos de pós-graduação em arquitetura e urbanismo**, v. 8, n. 1, 2008.

DAVIES, M. **Conceptual architectural design**. Disponível em: <https://www.designingbuildings.co.uk/wiki/Concept_architectural_design#:~:text=Concept design implies an idea,concept and a design intent.&text=The conceptual approach places the,concept design process in advance.>. Acesso em: 13 nov. 2020.

DAVIS, D. **Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture**. [s.l.] RMIT University, 2013.

DEB, K. et al. **Comparison of multiobjective evolutionary algorithms: Empirical results**. Technical Report 70, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, 1999.

DEB, K. et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. **IEEE Transactions on Evolutionary Computation**, v. 6, n. 2, p. 182–197, 2002.

DEB, K.; KALYANMOY, D. **Multi-Objective Optimization Using Evolutionary Algorithms**. USA: John Wiley & Sons, Inc., 2001.

DRÉO, J. et al. **Metaheuristics for Hard Optimization: Methods and Case Studies**. [s.l.] Springer Berlin Heidelberg, 2006.

DYNAMOBIM. **The Dynamo Primer**. Disponível em: <<https://primer.dynamobim.org/>>. Acesso em: 10 out. 2021.

EASTMAN, C. et al. **BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors**. Second ed. [s.l.] John Wiley & Sons, Inc, 2011.

ELTAWHEEL, A.; SU, Y. Parametric design and daylighting: A literature review. **Renewable and Sustainable Energy Reviews**, v. 73, p. 1086–1103, 2017.

ESTADÃO. **Entenda a importância da orientação solar para os imóveis**. Disponível em: <<https://imoveis.estadao.com.br/aluguel/entenda-a-importancia-da-orientacao-solar-para-os-imoveis/>>. Acesso em: 05 out. 2021.

FISCHER, T.; HERR, C. Teaching generative design. ... of the **4th Conference on Generative Art**, n. January, p. 1–14, 2001.

GONZÁLEZ, Marco A. S.. **Noções de orçamento e planejamento de obras**. São Leopoldo: Universidade do Vale do Rio dos Sinos – Ciências Exatas e Tecnológicas, 2008. Notas de Aula.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence**. Cambridge, MA, USA: MIT Press, 1992.

HUDSON, R. Strategies for parametric design in architecture. **Civil Engineering**, p. 274, 2010.

IORDANOVA, I. et al. Parametric methods of exploration and creativity during architectural design: Case study in the design studio. **Joining Languages, Cultures and Visions - CAAD Futures 2009, Proceedings of the 13th International CAAD Futures Conference**, p. 423–439, 2009.

KEOUGH, I.; BENJAMIN, D. **Multi-Objective Optimization in Architectural Design**. Multi-Objective Optimization in Architectural Design. **Anais...2010**

KERN, A. P. **Proposta de um Modelo de Planejamento e Controle de Custos de Empreendimentos de Construção**. 2005. 122 f. Tese (Doutorado em Engenharia Civil) - Escola de Engenharia. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2005.

LAWSON, B. **How Designers Think The design Process Demystified**. 4 ed. [s.l.] Elsevier, 2005.

MASCARÓ, Juan L. **O custo das decisões arquitetônicas**. 5. ed. Porto Alegre: Masquatro, 2010.

MATOS, W. S. **BIM: entendendo a curva de MacLeamy e como funciona basicamente o fluxo de trabalho em BIM**. Disponível em: <<https://engenhariaeetc.wordpress.com/2015/09/21/bim-entendendo-a-curva-de-macleamy-e-como-funciona-basicamente-o-fluxo-de-trabalho-em-bim/>>. Acesso em: 13 set. 2021.

MAYER, R. **A gramática da habitação mínima: análise do projeto arquitetônico da habitação de interesse social em Porto Alegre e Região Metropolitana**. 2012. 205 f. Tese (Doutorado em Arquitetura) - Programa de Pós-graduação em Arquitetura, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

MCCALL, I. **The importance of utilising conceptual design**. Disponível em: <<https://4dproducts.co.uk/the-importance-of-utilising-conceptual-design/>>. Acesso em: 13 nov. 2020.

MEOTTI, B. D. et al. A CONCEPÇÃO ARQUITETÔNICA COMO ELEMENTO DE HUMANIZAÇÃO DO MEIO URBANO. 2015.

MEREDITH, M. **From Control to Design: Parametric/Algorithmic Architecture**. [s.l.] Actar-D, 2008.

NAGY, D. et al. Project discover: An application of generative design for architectural space planning. **Simulation Series**, v. 49, n. 11, p. 49–56, 2017a.

NAGY, D. et al. The Buzz Metric: A Graph-based Method for Quantifying Productive Congestion in Generative Space Planning for Architecture. **Technology Architecture and Design**, v. 1, n. 2, p. 186–195, 2017b.

NAGY, D. **The problem of learning**. Disponível em: <<https://medium.com/generative-design/generative-design-introduction-64fb2db38e1>>. Acesso em: 13 nov. 2020.

NAGY, D.; VILLAGGI, L.; BENJAMIN, D. Generative urban design: Integrating

financial and energy goals for automated neighborhood layout. **Simulation Series**, v. 50, n. 7, p. 190–197, 2018.

OLIVEIRA, L. A.; MITIDIERI FILHO, C. V. O Projeto De Edifícios Habitacionais Considerando a Norma Brasileira De Desempenho: Análise Aplicada Para As Vedações Verticais. **Gestão & Tecnologia de Projetos**, v. 7, n. 1, 2012.

OXMAN, R. Theory and design in the first digital age. **Design Studies**, v. 27, n. 3, p. 229–265, 2006.

OXMAN, R. Performance-Based Design: Current Practices and Research Issues. **International Journal of Architectural Computing**, v. 6, n. 1, p. 1–17, 2008.

POTTMAN, H. et al. ARCHITECTURAL GEOMETRY First Edition. **Bentley Institute Press**, 2007.

QUEIROZ, Mario N. de. **Programação e Controle de Obras**. Juiz de Fora: Universidade Federal de Juiz de Fora, 2001. Notas de Aula.

ROHRMANN, J. Design Optimization in Early Project Stages A Generative Design Approach to Project Development. n. September, p. 121, 2019.

ROSSO, T. **Aspectos geométricos do custo das edificações. Simpósio sobre barateamento da construção habitacional**. Trabalho nº 83. Salvador, mar., 1978.

ROUDAVSKI, S. Towards Morphogenesis in Architecture. **International Journal of Architectural Computing**, v. 7, n. 3, p. 345–374, 2009.

RUTTEN, D. **Evolutionary Principles applied to Problem Solving**. Disponível em: < <https://www.grasshopper3d.com/profiles/blogs/evolutionary-principles>>. Acesso em: 10 out. 2021.

RUTTEN, David. Galapagos: On the logic and limitations of generic solvers. **Architectural Design**, v.83, n.2, p.132-135, 2013.

RYLANDER, S. G. B.; GOTSHALL, B. Optimal population size and the genetic algorithm. **Population**, v. 100, n. 400, p. 900, 2002.

SAMPAIO, F. M. **Orçamento e custo da Construção**. São Paulo: Hemus, 2004.

SÃO PAULO, lei nº 16.050, de 31 de julho de 2014. Disponível em: < <https://leismunicipais.com.br/plano-diretor-sao-paulo-sp>>. Acesso em: 09 set. 2021.

SCHMITZ, B. **The Importance of Concept Generation in the Early Phase of Design**. Disponível em: <<https://www.ptc.com/en/blogs/cad/what-a-concept-the-importance-of-the-early-phase-of-design>>. Acesso em: 13 nov. 2020.

SHEA, K.; AISH, R.; GOURTOVAIA, M. Towards integrated performance-driven generative design tools. **Automation in Construction**, v. 14, p. 253–264, 2005.

SIVANANDAM, S; DEEPA, S. N. **Introduction to Genetic Algorithms**. Springer, 2008.

TERZIDIS, K. The nature of computation Algorithmic Design: A Paradigm Shift in Architecture? p. 201–207, 2002.

TERZIDIS, K. **Algorithmic Architecture**. [s.l.] Architectural Press, 2006.

TURRIN, M.; VON BUELOW, P.; STOUFFS, R. Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. **Advanced Engineering Informatics**, v. 25, n. 4, p. 656–675, 2011.

UMAKOSHI, E. M. Avaliação de Desempenho Ambiental e Arquitetura Paramétrica Generativa para o Projeto do Edifício Alto. p. 253, 2014.

YU, R.; GU, N.; OSTWALD, M. A method for comparing designers' behavior in two environments: Parametric and geometric modeling. **Open Systems - Proceedings of the 18th International Conference on Computer-Aided Architectural Design Research in Asia, CAADRIA 2013**, n. Caadria, p. 479–488, 2013.