



MINISTÉRIO DA EDUCAÇÃO
Universidade Federal de Ouro Preto
Escola de Minas – Departamento de Engenharia Civil
Programa de Pós-Graduação em Construção Metálica
Mestrado Profissional em Construção Metálica – MECOM



Renato Godoi da Cruz

***DESIGN GENERATIVO COMO FERRAMENTA E METODOLOGIA
PROJETUAL NA CONSTRUÇÃO METÁLICA***

Ouro Preto
2018

Renato Godoi da Cruz
renatogcruz@hotmail.com

***DESIGN GENERATIVO COMO FERRAMENTA E METODOLOGIA
PROJETUAL NA CONSTRUÇÃO METÁLICA***

Dissertação apresentada ao Programa de Pós-Graduação em Construção Metálica do Departamento de Engenharia Civil da Escola de Minas da Universidade Federal de Ouro Preto como parte integrante dos requisitos para a obtenção do título de Mestre em Construção Metálica.

Orientadora: Dra. Cláudia Maria Arcipreste

Coorientadores: Dra. Rovadávia Aline de Jesus Ribas
Me. Rafael Lemieszek Pinheiro

Ouro Preto
2018

cutte

Cruz, Renato Godoi da.

Design generativo como ferramenta e metodologia projetual na construção metálica [manuscrito] / Renato Godoi da Cruz. - 2018.

118f.:

Orientadora: Prof^a. Dr^a. Cláudia Maria Arcipreste.

Dissertação (Mestrado) - Universidade Federal de Ouro Preto. Escola de Minas. Departamento de Engenharia Civil. Mestrado Profissional em Construção Metálica.

Área de Concentração: Construção Metálica.

1. Design generativo. 2. Algoritmos genéticos. 3. Otimização estrutural. 4. Otimização ambiental. 5. Construção metálica. I. Arcipreste, Cláudia Maria . II. Universidade Federal de Ouro Preto. III. Titulo.

CDU: 624.014



Universidade Federal
de Ouro Preto



MINISTÉRIO DA EDUCAÇÃO
Universidade Federal de Ouro Preto
Escola de Minas – Departamento de Engenharia Civil
Programa de Pós-Graduação em Construção Metálica
Mestrado Profissional em Construção Metálica – MECOM



DESIGN GENERATIVO COMO FERRAMENTA E METODOLOGIA PROJETUAL NA CONSTRUÇÃO METÁLICA

AUTOR: RENATO GODOI DA CRUZ

Esta dissertação foi apresentada em sessão pública e aprovada em doze de dezembro de 2018, pela Comissão Examinadora composta pelos seguintes membros:

Profa. Cláudia Maria Arcipreste, D. Sc. – UFOP (Presidente)

Profa. Rovadávia Aline de Jesus Ribas, D. Sc. – UFOP

Prof. Rafael Lemieszek Pinheiro, M. Sc. – Pitágoras

Prof. Flávio Teixeira de Souza, D. Sc. – IFMG

Prof. Roberto Eustáquio dos Santos, D. Sc. – UFMG

AGRADECIMENTO

Para a concretização deste trabalho, contei com as orientações seguras e cuidadosas das professoras Cláudia Maria Arcipreste e Rovadávia Aline de Jesus Ribas e do professor Rafael Lemieszek Pinheiro.

Aos demais professores do MECOM, pela contribuição em suas disciplinas para o aprimoramento da minha formação.

Agradeço a oportunidade ao debate, às críticas e sugestões enriquecedoras feitas pelos professores Flávio Teixeira de Souza e Roberto Eustaáquio dos Santos na banca examinadora, durante a defesa desta dissertação.

Agradeço a minha mãe Ofélia, pelo apoio incondicional. Esse trabalho também é seu.

À Universidade Federal de Ouro Preto, pela oportunidade de me qualificar profissionalmente. Sou grato ao Departamento de Engenharia Civil da UFOP e a todos que passaram pela Secretaria do MECOM durante esses anos.

Agradeço também aos meus colegas de turma, pelo companheirismo, ensinamentos e momentos de descontração que vivemos juntos. Em especial Samantha, Rafael, Jarbas, Uáscar e Geórgia.

Credito à Oficina Paramétrica, ministrada pelo prof. Rafael Lemieszek, que lá em 2015 mostrou que, na arquitetura, o processo pode ser muito mais interessante do que o produto final.

À família Bastilha e aos irmãos republicanos pela eterna amizade e acolhimento no primeiro ano deste trabalho – e nas dezenas de outras vezes que voltei a Ouro Preto.

Aproveito para deixar registrado a recepção e a disponibilidade da comunidade Python Brasil, sou grato às pessoas pela generosidade com que receberam minhas dúvidas. Por fim, minha gratidão as pessoas que disponibilizam conteúdos gratuitamente na rede (github, fórum Grasshopper®, entre tantas outras plataformas), onde se pode aprender muito.

Considero muito o apoio e carinho de todos aqueles que em algum momento ou de alguma forma me auxiliaram nesta trajetória.

RESUMO

O projeto arquitetônico é composto por inúmeras variáveis, algumas delas contraditórias, que precisam ser negociadas constantemente. Os algoritmos genéticos representam uma técnica que contribuem para a solução desse tipo de problema, potencializando a busca de melhores resultados. Para isso, basta que estes sejam incorporados ao sistema gerativo de projeto. O presente trabalho descreve a construção de um sistema que combina estratégias de modelagem paramétrica e algoritmos genéticos para otimização do peso total de uma cobertura em estrutura metálica e de sua superfície como potencial área de geração de energia fotovoltaica. Por meio de reformulação do processo evolutivo darwiniano, procurou-se sistematizar um processo de projeto que permitisse ao arquiteto atuar na parametrização dos problemas, indo além da mera proposição formal de soluções, em favor da exploração de maior variedade de possibilidades projetivas do que seria possível usando métodos tradicionais de projeto. Assim, os objetivos desta pesquisa são avaliar o potencial da utilização dos algoritmos genéticos como ferramenta auxiliar integrada ao *design* gerativo como um método de projeto. A revisão do conteúdo teórico e prático da arquitetura abordou a prática projetual em que o arquiteto abre mão do controle de definição específica da solução para alcançar esta variedade maior de projetos. Como objeto de estudo, buscou-se aplicar o processo estudado em um projeto arquitetônico de construção metálica e de seu potencial energético. Como resultado destaca-se o projeto como uma prática em que arquiteto e projeto dialogam em um fluxo de informações que precedem o projeto em si, orientado por objetivos comuns. Os experimentos realizados mostram que a perda de controle na formulação final de uma solução se deu pela modelagem matemática do problema e na busca automática de respostas por meio de heurísticas. Destaca-se também uma abordagem transdisciplinar que articula teoria e prática dentro de um pensamento capaz de entrelaçar especialidades e superar fronteiras na produção do conhecimento.

Palavras-chave: *Design* gerativo, Algoritmos genéticos, Otimização estrutural, Otimização ambiental, Construção metálica.

ABSTRACT

The architectural design is composed of innumerable variables, some of them contradictory, that need to be negotiated constantly. The genetic algorithms represent a technique that contribute to the solution of this type of problem, potentializing the search for better results. For this, it is enough that these are incorporated into the generative design system. The present work describes the construction of a system that combines parametric modeling strategies and genetic algorithms to optimize the total weight of a roof in metallic structure and of its surface as a potential area of photovoltaic energy generation. By means of a reformulation of the evolutionary Darwinian process, a systematization of a project process that allowed the architect to act in the parametrization of the problems, going beyond the mere formal proposition of solutions, in favor of the exploration of a greater variety of projective possibilities of what would be possible using traditional design methods. Thus, the objectives of this research are to evaluate the potential of the use of genetic algorithms as an auxiliary tool integrated to generative design as a design method. The review of the theoretical and practical content of the architecture approached the design practice in which the architect gives up the control of the specific definition of the solution to reach this greater variety of projects. As object of study, we tried to apply the studied process in an architectural design of metallic construction and its energetic potential. As a result, the project stands out as a practice in which the architect and the project dialogue in a flow of information that precedes the project itself, guided by common objectives. The experiments show that the loss of control in the final formulation of a response occurred through the mathematical modeling of the problem and the automatic search of solutions through heuristics. It also highlights a transdisciplinary approach that articulates theory and practice within thinking capable of intertwining specialties and overcoming frontiers in the production of knowledge.

Key-words: *Generative design, Genetic algorithms, Structural optimization, Environmental optimization, Metallic construction.*

LISTA DE FIGURAS

Figura 1: Modelo de experimento de Gaudí	16
Figura 2: Modelo de experimento de Otto.....	17
Figura 3: Modelo de experimento de Spuybroek	18
Figura 5: Los Manantiales, obra de Felix Candela.	22
Figura 4: Inspiração e desenhos de Candela	22
Figura 6: Genótipo x Fenótipos	27
Figura 7: Diagrama de aplicação do processo generativo	32
Figura 8: Localização Ouro Preto.....	34
Figura 9: Diagrama de concepção	35
Figura 10: Construção dos pontos iniciais do modelo.....	36
Figura 11: Construção da primeira parábola do modelo	36
Figura 12: Diagrama visual da construção da primeira parábola do modelo	36
Figura 13: Construção dos demais elementos do modelo	38
Figura 14: Vista lateral da construção do modelo	38
Figura 15: Diagrama do modelo e controles paramétricos.....	39
Figura 16: Genótipo <i>vs</i> Fenótipo do modelo	41
Figura 17: Partes dos componentes de modelagem estrutural Karamba®.....	43
Figura 18: Configuração dos apoios – restrições e liberdades.....	43
Figura 19: Definição das cargas	44
Figura 20: Componentes de cálculo das combinações de ações.....	45
Figura 21: Reajuste na carga de Peso Próprio devido à combinação.....	45
Figura 22: Componentes de cálculo da área de cobertura	46
Figura 23: Componentes de construção e cálculo da área dos painéis fotovoltaicos (áreas cinza)	46

Figura 24: Ilustração das cargas nodais (permanentes)	47
Figura 25: Conversão, divisão e distribuição das cargas nos nós dos elementos D	47
Figura 26: Área utilizada para cálculo da força de arrasto.....	48
Figura 27: Diagrama para coeficiente de arrasto.....	49
Figura 28: Velocidade básica do vento	49
Figura 29: Algoritmo para definição do valor da carga de vento.	50
Figura 30: Algoritmo de busca das seções transversais	50
Figura 31: Blocos de algoritmos de extração de forças <i>Sd</i> (a) e cálculo de forças <i>Rd</i> (b).....	53
Figura 32: Bloco de algoritmo das condições de restrições, seus <i>impus</i> e seus <i>outputs</i>	55
Figura 33: Verificação da direção normal das áreas reservadas para os painéis fotovoltaicos.....	56
Figura 34: Direção normal das áreas reservadas para os painéis fotovoltaicos....	57
Figura 35: Localização Ouro Branco x Ouro Preto.....	57
Figura 36: Componente para leitura do clima específico	57
Figura 37: Componente para importar as configurações de módulo fotovoltaico específico.....	58
Figura 38: Conjunto de nós para calcular a quantidade de energia elétrica que pode ser produzida por uma superfície durante um ano.....	59
Figura 39: Construção da função composta do primeiro experimento	65
Figura 40: Gráfico de dispersão mono-objetivo.....	69
Figura 41: Percurso do projeto	69
Figura 42: Opções escolhidas no gráfico de dispersão	70
Figura 43: Progresso do projeto no gráfico de dispersão mono-objetivo	71
Figura 44: Opção número #5899 e seus parâmetros de entradas	73

Figura 45: Gráfico de dispersão multiobjetivo.....	74
Figura 46: Percurso dos projetos	75
Figura 47: Seleção de parte das opções ótimas.....	75
Figura 48: Opções ótimas selecionadas do segundo experimento.....	76
Figura 49: Alternativa número #5852 e seus parâmetros de entradas	79
Figura 50: Alternativa número #5808 e seus parâmetros de entradas	80

SUMÁRIO

1 INTRODUÇÃO.....	1
1.1 Justificativa	4
1.2 Objetivos	5
1.2.1 Objetivo Geral	5
1.2.2 Objetivos Específicos	6
1.3 Método e processo de projeto.....	6
1.4 Estrutura do Trabalho.....	8
2 REFERENCIAL TEÓRICO	11
2.1 Projeto como Pergunta	14
2.2 Projeto e Controle	16
2.3 Projeto e Natureza	22
2.4 Projeto do Projeto.....	27
3 FERRAMENTA E ESTRATÉGIA PARA O DESIGN GENERATIVO ...	30
4 IMPLEMENTAÇÃO E EXPERIMENTOS.....	33
4.1 A Geometria: construindo o espaço de projeto.....	34
4.1.1 Construindo a avaliação estrutural.....	42
4.1.2 Construindo a avaliação energética	56
4.2 Algoritmo Generativo: o solucionador evolutivo Discover [©]	61
4.2.1 Configurando o Algoritmo Generativo	64
4.3 Análise do resultado – Evolução do Projeto.....	68
4.3.1 Experimento 01: Otimização Mono-objetivo (função composta)	69
4.3.2 Experimento 02: Otimização Multiobjetivo.....	74
5 CONSIDERAÇÕES E SUGESTÕES	81
5.1 Considerações finais	82
5.2 Sugestões para trabalhos futuros	85
REFERÊNCIAS	87
GLOSSÁRIO.....	91
APÊNDICES.....	96

APÊNDICE A - ALGORITMO PARA A COMBINAÇÃO DE AÇÕES	96
APÊNDICE B - ALGORITMO PARA DETERMINAR A FORÇA DE ARRASTO	97
APÊNDICE C - ALGORITMO PARA A ESCOLHA DOS PERFIS	98
APENDICE D - ALGORITMO PARA ANÁLISE ESTRUTURAL	100
ANEXOS	104
ANEXO A - ALGORITMO PARA OTIMIZAÇÃO MONO-OBJETIVO	104
ANEXO B – ALGORITMO PARA OTIMIZAÇÃO MULTIOBJETIVO	106

1 INTRODUÇÃO

O presente trabalho trata de um dos caminhos que o processo de projeto arquitetônico pode percorrer na contemporaneidade.

Por meio de revisão bibliográfica e por meio da elaboração de projeto arquitetônico, buscou-se reformular o processo evolutivo darwiniano, sistematizando o processo de projeto para permitir ao arquiteto atuar na parametrização dos problemas, além da mera proposição formal de soluções, em favor da exploração de uma maior variedade de possibilidades projetivas do que seria possível usando métodos tradicionais de projeto.

Aplicando critérios de desempenho e de otimização estrutural, a intenção foi sistematizar campos de conhecimento comuns da arquitetura, da engenharia e de outras disciplinas – matemática e ciências da computação, numa articulação transdisciplinar dentro de um cenário próprio, mas configurável e disposto a incluir outros saberes. Para isso, combinamos estratégias de modelagem paramétrica e de algoritmo genético para otimização no desenvolvimento de uma estrutura em aço para a cobertura da arquibancada do campo de futebol de uma instituição de ensino.

O processo tradicional de projeto pode ser descrito como um processo consciente de experimentação: hipóteses de solução formal são formuladas e testadas (em desenhos e modelos) até que o projetista se considere satisfeito com o resultado. Neste trabalho, foi proposta a utilização de ferramentas paramétricas para explorar um

número muito grande de hipóteses de projeto dentro de um conjunto de variáveis estipuladas *a priori* com o objetivo de explorar complexidades que manualmente seriam, em termos práticos, impossíveis de serem testadas.

O processo de *design* generativo adotado utiliza o algoritmo genético como meio de geração de alternativas, utilizando diferentes *softwares* de modelagem paramétrica, de análises estruturais e ambientais. Neste caso, os *softwares* utilizados são, de fato, ferramentas, já o *design* generativo, em si, é processo apoiado por metodologias específicas, que vão além dos procedimentos demandados pelos *softwares* relacionados.

As ferramentas digitais, nesta dissertação, foram assumidas como um componente no processo de projeto, mas quando integradas ao método proposto, ao processo desenvolvido, percebeu-se que seus recursos não eram apenas ferramentas, mas dispositivos capazes de oferecer capacidades específicas a serem exploradas, potencializadas e redefinidas pelo método projetual e pela postura do arquiteto.

Para a modelagem geométrica, utilizou-se um *software* de interface em *Visual Programming Language* (VPL) que opera a partir de um editor baseado em nós, que conduz a uma serie de operações geométricas para produzir uma geometria variável, baseada nos parâmetros de entrada. Podemos entender os parâmetros como uma espécie de interface, capaz de tornar o projeto arquitetônico

mutável. Esses parâmetros de entradas descrevem todos os *designs* possíveis dentro de um espaço de possibilidades, onde cada parâmetro adicionado aumenta significativamente o número de *designs*. Esse conjunto de *designs* possíveis é comumente chamado de espaço de projeto (NAGY, 2018). Neste trabalho, os parâmetros de entradas foram controlados por um algoritmo genético para otimizar o modelo a partir de objetivos. Juntos, esses objetivos criam uma abordagem que garante alto desempenho, ao mesmo tempo em que permite que o arquiteto atue orientado não pela determinação da forma em si, mas por um conjunto de condicionantes – físicas, ambientais, locais – que resultará, dialogicamente, na forma.

1.1 Justificativa

Atualmente, o desenvolvimento de novas tecnologias tem possibilitado alterações significativas nos processos projetuais em arquitetura.

Considerando a diversidade de ferramentas digitais na atualidade, o crescente emprego de algoritmos nos processos de projeto de arquitetura e o grande número de variáveis que deve ser considerado durante o desenvolvimento projetual, acredita-se que a integração de diferentes áreas de conhecimento pelos arquitetos faça com que esses profissionais fujam da postura de meros usuários de ferramentas para uma postura mais exploratória, capaz de integrar outros conhecimentos, de instrumentalizar essas ferramentas.

Frente à complexidade do projeto de arquitetura que se pretendeu abordar neste trabalho, acredita-se numa abordagem projetual que possibilitasse uma interação entre as diferentes áreas do conhecimento, que supere as fronteiras, a migração do conceito de um campo de saber para outro e a unificação do conhecimento a fim de superar a dicotomia decorrente da fragmentação da especialização.

Com o algoritmo evolutivo pretendeu-se desenvolver uma estratégia para a exploração sistemática das soluções ambientais e estruturais, adotadas como condicionantes para geração de mecanismos de simulações energéticas e desempenho estrutural, identificando regras fundamentais que permitam a criação de possíveis combinações, definindo um processo analítico. Assim, acredita-se que (a) a reversão metodológica aplicada ao projeto por meio do *design* generativo e da simulação como agenciadores da forma pode propiciar resultados inovadores e não previstos, já que, com as soluções geradas através da formalização e parametrização de problemas específicos em questão e (b) a partir dessa confirmação, é possível que a descrição do processo de instrumentalização deste método ofereça um material de suporte, de maneira eficiente, para os profissionais da área.

1.2 Objetivos

Esta dissertação tem os objetivos, geral e específicos, que se seguem.

1.2.1 Objetivo Geral

O objetivo geral dessa pesquisa foi avaliar o potencial da utilização dos algoritmos genéticos como ferramenta auxiliar integrada ao *design* gerativo como um método de projeto, identificando o conteúdo teórico de base, explorando e avaliando esta abordagem em um estudo de caso.

1.2.2 Objetivos Específicos

Para estruturar e conduzir o trabalho, foram definidos como objetivos específicos:

- (a) – a revisão do conteúdo teórico e prático da arquitetura baseada na prática projetual em que o arquiteto abre mão do controle da definição específica da solução em favor da exploração de uma variedade maior de projetos do que seria possível usando métodos de *design* tradicionais;
- (b) – a aplicação deste conteúdo em projeto arquitetônico de uma cobertura em construção metálica e do cálculo de seu potencial energético, bem como a descrição deste processo como forma de contribuir para o embasamento teórico e prático de suporte para os arquitetos, engenheiros, assim como estudantes de arquitetura e engenharia.

1.3 Método e processo de projeto

Os procedimentos metodológicos aplicados neste trabalho foram divididos em três etapas:

- (a) – a primeira etapa consistiu de uma pesquisa teórica referencial na qual se discutiu o projeto arquitetônico como processo;
- (b) – em uma segunda etapa foi aplicado o processo evolutivo em um estudo de caso. Este processo combinou estratégias de modelagem paramétrica e de algoritmo genético como método para se atingir os objetivos propostos;
- (c) – em uma terceira etapa as discussões e ponderações foram organizadas e descritas.

A segunda etapa se desdobrou em três partes:

- (a) – a construção do espaço de projeto, ou seja, o sistema que gera todas as soluções possíveis para um determinado problema de projeto;
- (b) – o desenvolvimento de medidas para julgar o desempenho de cada projeto, primeiramente abordado como uma função mono-objetiva composta, que deve minimizar o resultado entre a divisão do produto da multiplicação entre o peso total da estrutura (em kg) e o preço

médio do aço (em *R\$*) pelo produto da multiplicação entre o potencial energético da cobertura (em *kwh*) e o preço médio do *kwh* (em *R\$*). Em seguida, as funções foram divididas em objetivos distintos, que seriam: minimizar o peso total da estrutura e maximizar o potencial energético da cobertura, criando um problema multiobjetivo;

- (c) – a aplicação de algoritmos evolutivos para pesquisar o espaço de projeto e encontrar projetos de alto desempenho.

Para isso, foram utilizados os *softwares* Rhinoceros®, para modelagem tridimensional e seu *plug-in* de modelagem paramétrica Grasshopper®. Também foram aplicados os seguintes *add-ons* para Grasshopper®:

- (a) – Karamba® – aplicativo para análise estrutural;
- (b) – Ladybug® – aplicativo para análise ambiental;
- (c) – Discover© – framework para projeto generativo.

1.4 Estrutura do Trabalho

A dissertação foi organizada em cinco seções:

- (a) – contextualização inicial;
- (b) – revisão de literatura;

- (c) – ferramentas e estratégias para o *design* generativo;
- (d) – implementação e experimentos e
- (e) – considerações finais.

Em cada uma dessas seções procurou-se explorar conteúdos específicos e necessários para o entendimento do tema como um todo.

Na introdução são feitas considerações iniciais em que se apresentam o contexto geral da pesquisa, sua justificativa, objetivos e método.

No capítulo 2 apresenta-se a proposta do projeto como pergunta. Esta abordagem possui como característica modelos fluidos, que propõe maneiras convenientes para lidar na formalização do problema – a pergunta – e se contrapõe ao método convencional, baseado na resposta – a solução – que por ser definida previamente tende a diminuir as chances de soluções não previstas anteriormente se apresentem. O caminho explorado para isso está na reformulação do processo evolutivo como um processo matemático de otimização.

No capítulo 3 descrevem-se as ferramentas e estratégias para o *design* generativo.

No capítulo 4 apresenta-se a implementação e o experimento em si, as ferramentas e estratégias para o *design* generativo, assim como a descrição da construção do espaço de projeto, a construção do sistema de avaliações e a aplicação do algoritmo genético.

Na última seção estão as considerações finais e sugestões para trabalhos futuros. Nelas se

estabelece uma relação entre o conteúdo teórico e os experimentos, sua implementação, os resultados obtidos, assim como sugestões de futuros caminhos para o aprofundamento deste trabalho.

Para auxiliar no objetivo de tornar este trabalho uma contribuição teórica e prática efetiva para os estudantes e profissionais da arquitetura e engenharia civil, esta dissertação conta com glossário, apêndices e anexos. A produção pós-textual tem como objetivo assegurar maior clareza e compreensão do trabalho, primeiro porque apesar da preocupação em adotar termos claros, objetivos e que se entende ser conhecidos aos olhos dos diferentes públicos-alvo deste trabalho, observou-se que, em alguns casos, a terminologia empregada não pode deixar de ser um tanto hermética, empregando termos para os que já foram iniciados ao tema. Com o mesmo fim, os apêndices e anexos garantem acesso a todos os algoritmos escritos e utilizados nesta dissertação, que por ventura poderão ser copiados e testados.

O cuidado com os elementos pós-textuais é uma tentativa de desfazer a barreira contra os leigos, tradicionalmente construídos pela linguagem técnica e oportunamente mantidos como forma de poder dentro dos campos, neste caso, da engenharia e arquitetura.

2 REFERENCIAL TEÓRICO

O ponto inicial do trabalho é também uma questão central para a prática de projeto de arquitetura na atualidade: não há como prever completamente todas as demandas inerentes ao projeto de arquitetura.

Diversos autores, como Giaccardi e Fischer (2008) e Schumacher (2012), já nos alertavam de que não devemos assumir que o processo de projeto pode ser inicialmente planejado de forma integral.

Esta questão está diretamente relacionada com a da complexidade. Ambas se revelam em uma consciência emergente, consequência de um pensamento projetual mais flexível e experimental, que reconhece as limitações do pensamento clássico disciplinar (ROCHA, 2016).

Devemos inicialmente tentar compreender o significado da complexidade nesse contexto e definir qual a natureza de nosso interesse em complexidade. Primeiramente, procuramos entender a complexidade como algo crescente e inerente à própria sociedade humana.

Há algumas diferenças fundamentais entre o que frequentemente entendemos como complexo – aquilo que está além de nosso entendimento – para aquilo que realmente é complexo, ou seja, algo que pela própria característica e natureza não pode ser reduzido, disposto, apresentado, assimilado como algo simples (VASSÃO, 2010).

Considerando o processo de *design* em arquitetura como nossa tentativa de compreensão do mundo, Ribeiro e Pratschke (2005) acreditam

que a atividade projetual é por si só inherentemente complexa. Para os autores, a questão da complexidade em arquitetura é uma alternativa não reducionista, uma maneira de pensar o projeto como processo em um ambiente de interação.

Para Rutten (2013) e Nagy (2018), o *design* é complexo porque, embora sua materialização seja um objeto físico no espaço tridimensional, seu processo pertence a múltiplas dimensões.

Para entender melhor o processo de *design* desse ponto de vista, devemos entender a dimensionalidade do processo de *design* como algo atribuída à independência de suas escolhas, ou seja, às possibilidades de liberdades deste processo. Dessa forma, devemos considerar o projeto além da sua dimensão geométrica, devemos considerá-lo em todas suas decisões, onde cada escolha o definirá diferente, definindo-o como algo altamente complexo.

Nosso interesse na complexidade, então, está em considerar o processo de *design* arquitetônico como um sistema de múltiplas dimensões. A partir disso, segue-se uma revisão de literatura baseada na identificação teórica e prática de como abordar o projeto de arquitetura de maneira a não limitar, ou pelo menos minimizar a limitação das possibilidades do projeto.

2.1 Projeto como Pergunta

O processo de projeto em arquitetura tende a ser altamente dependente do caminho descrito pelo arquiteto. Isso significa que à medida que o processo de projeto avança no tempo, as decisões são tipicamente tomadas quando são confrontadas, limitando as possibilidades do projeto em seus estágios posteriores.

A consequência deste fato, é que o projeto acaba sendo o resultado de aproximações sucessivas, em pequenos ciclos de análise, síntese e avaliação (LAWSON, 1983). A competência do arquiteto para a tomada de decisão está na sua habilidade de retomar, reestruturar, adaptar e aplicar ideias relevantes advindas de projetos anteriores (OXMAN e OXMAN, 1992). Entretanto, isso pode se tornar um problema, pois quando os conceitos dos arquitetos tornam-se um padrão de pensamento, as soluções para novos problemas podem se restringir a mera reprodução de soluções anteriores, comprometendo a descoberta de novas possibilidades (BUCHANAN, 1992) ou, ainda, soluções melhores.

Além de que, em arquitetura (projeto), não há uma única solução (resposta) para um problema (pergunta). A partir desse reconhecimento, o arquiteto Caio Adorno Vassão termina seu livro *Metadesign: ferramentas, estratégias e ética para a*

complexidade (2010) propondo o projeto como pergunta. Essa ideia contradiz o entendimento que busca como produto de um problema, uma solução como resposta.

Essa proposta de Vassão tem várias dimensões e vai além do que se pretende abordar neste trabalho. Entretanto, podemos destacar duas delas.

A primeira, é que todo projeto é complexo, se assim ele for encarado. Isso porque todo objeto, todo processo está inserido em um sistema maior, sempre. Então, para mapear a complexidade, o arquiteto deve se questionar em que contexto aquele objeto está. Abordando o projeto como pergunta, o arquiteto estará mapeando esse contexto e não especulando uma resposta que ele julga, muitas vezes sozinho, ser a correta.

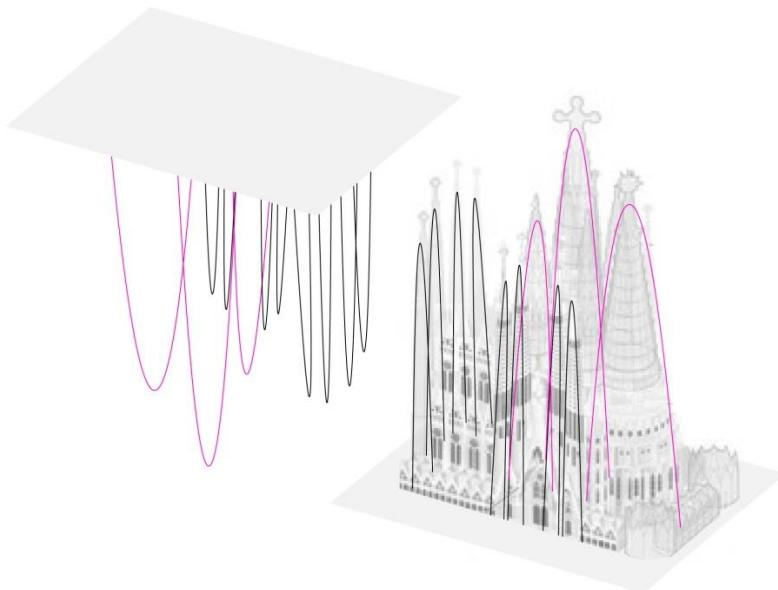
A segunda dimensão é que abordar o projeto como uma pergunta é mais ético, não no sentido moral, mas no sentido do arquiteto se colocar de maneira mais intencional no processo, mais presente. Esta interferência com intenção o coloca em uma posição de observador que permite ao arquiteto agir de maneira mais eficaz, permitindo obter melhores resultados com menos esforços.

2.2 Projeto e Controle

Abordagens de ordem não-teoremática do projeto não são algo novo e, há tempo, diferentes profissionais de projeto vêm adotando metodologias projetuais que possibilitam oportunidades para o surgimento de resultados dos quais são estruturalmente diferentes dos obtidos pelos procedimentos convencionais, se não em seus produtos finais, ao menos nos caminhos para se chegar até eles.

Um exemplo é Antoni Gaudí e seus experimentos com redes suspensas (Figura 1). Presas nas extremidades inferiores de uma superfície plana, as redes, quando estabilizadas, são representações inversas de catenárias.

Figura 1: Modelo de experimento de Gaudí



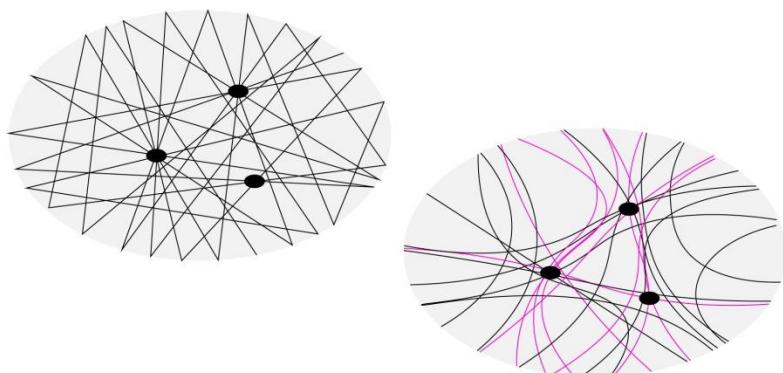
Fonte: O autor, 2018.

Esse experimento foi utilizado para a definição de colunas inclinadas e abóbadas baseadas em hiperboloides e paraboloides determinada pela

forma ótima da catenária durante o projeto da Sagrada Família.

Outro exemplo é o estudo *Optimized Path Systems*, de Frei Otto. Semelhante à técnica empregada por Gaudí na Sagrada Família, o método de Otto, conhecido como *máquina de fios de lã*, foi usado para gerar modelos de cidades, simulando fisicamente seus sistemas de circulação (Figura 2).

Figura 2: Modelo de experimento de Otto



Fonte: O autor, 2018.

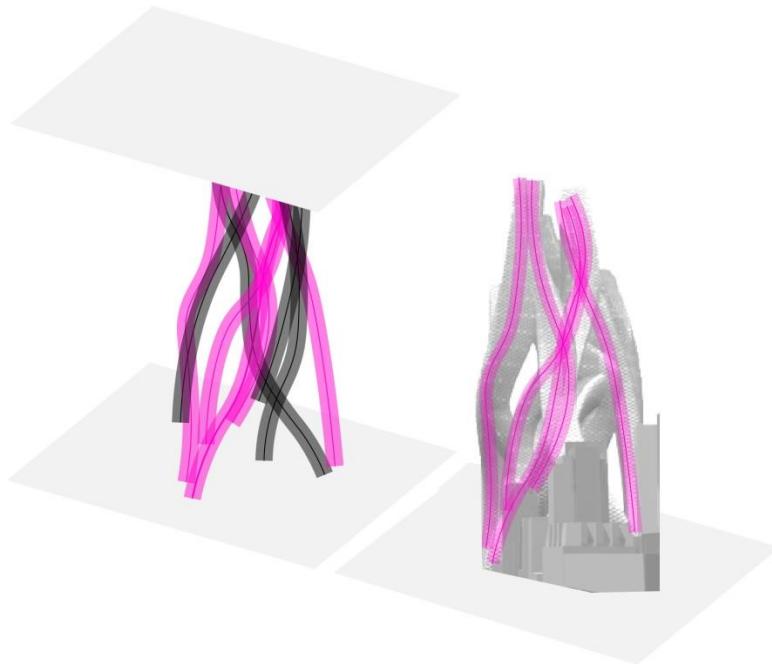
O processo de concepção do projeto *Oblique WTC*, de Lars Spuybroek, é mais um exemplo de método não-teoremático da prática arquitetônica.

Análogo ao modelo de fios de lã de Otto, a técnica de fios de lã de Spuybroek consistiu em duas estruturas quadradas, demarcadas por *grids* ortogonais e conectadas por fios de lã. Quando esse conjunto é mergulhado em água e balançado horizontalmente, os fios imediatamente começam a se auto-organizarem em um sistema complexo de ramificações (Figura 3).

Após este processo, o modelo foi digitalizado e cada fio ganhou mais espessura, constituindo os

núcleos das torres que compõem o complexo. (FERRANTE, 2013).

Figura 3: Modelo de experimento de Spuybroek



Fonte: O autor, 2018.

Em todos estes casos, os métodos adotados possuem como característica modelos que buscam por outros caminhos, maneiras convenientes para lidar com o problema (pergunta). Assim como a ciência nômade, o projeto deixa de ser teorematíco, da ordem das razões e passa a ser problemático, aquele que ultrapassa o obstáculo, que se projeta (DELEUZE e GUATTARI, 1995).

Nessa reinterpretação e ressignificação da prática projetual podem-se destacar duas características importantes:

- (a) – eles possuem como característica um modelo fluido;

- (b) – são exemplos que se opõem às metodologias baseadas na busca direta de uma solução (resposta).

A primeira característica se refere a uma aplicação customizada de procedimentos científicos (modelagem) para a síntese da forma (*design* final). Estas metodologias de projeto se contrapõem à tentativa de delimitar uma base científica mais geral (teorema), base da crítica de Vassão. A segunda característica diz respeito ao controle, ou, mais especificamente, a falta dele no processo de projeto. Nota-se que em todos os três casos, os arquitetos perderam controle das definições específicas das soluções para operarem especificamente na formalização dos problemas.

Para entendermos melhor essa abordagem, precisamos rever os exemplos sob duas perspectivas.

A primeira é reconhecer a arquitetura como um sistema altamente complexo, onde o número de variáveis do projeto (sistema controlado) é exponencialmente maior do que o número de variáveis que o arquiteto (sistema controlador) é capaz de antecipar. A segunda é que quando admitido à primeira ideia e mesmo assim o arquiteto tenta exercer o controle do projeto, muitas vezes, o controle é mal empregado e acabam assumindo a forma de restrição, ao invés de gerenciamento efetivo.

Isso acontece porque, segundo a Lei da Variedade Requerida de William R. Ashby

(1970), para que um sistema seja capaz de regular outro, exige-se daquele que se disponha de pelo menos tantos estados (variáveis) quanto o sistema que se deseja controlar possa apresentar.

No contexto do projeto em arquitetura, podemos ver a tentativa desse tipo de gerenciamento nos processos de projetos baseados na solução (resposta). Como resultado, pode-se dizer que o número de possíveis novas descobertas e, principalmente, aquelas não previstas anteriormente, tenham menores chances de aparecer.

Para Glanville (2002), a ausência de controle pode ser uma estratégia interessante para expandir o número de possibilidade, permitindo que opções não previstas anteriormente tenham a oportunidade de surgir. Para isso, nos sugere uma reformulação de como entendemos o valor do controle para nos permitir o benefício de estar fora do controle.

A ausência de controle, neste caso, pode ser usada para descobrir alternativas inovadoras que, de outra maneira, ficariam ocultas aos olhos do arquiteto.

Assumir estar fora do controle na prática de projeto não é aceitar um *design* menor ou incompleto, mas sim entendê-lo como uma oportunidade de alteridade e de mutabilidade que estão à disposição de uma operação de projeto.

Mas então, como abrir mão do controle na prática de projeto de arquitetura? Para

Nagy (2018) uma boa estratégia está no modo como os organismos são projetados na Natureza.

2.3 Projeto e Natureza

A inspiração na Natureza sempre esteve presente nas estratégias projetuais de arquitetos e engenheiros. No entanto, muito dessa inspiração tem como foco a reprodução formal de sistemas naturais (Figuras 4 e 5).

Figura 5: Los Manantiales, obra de Felix Candela.

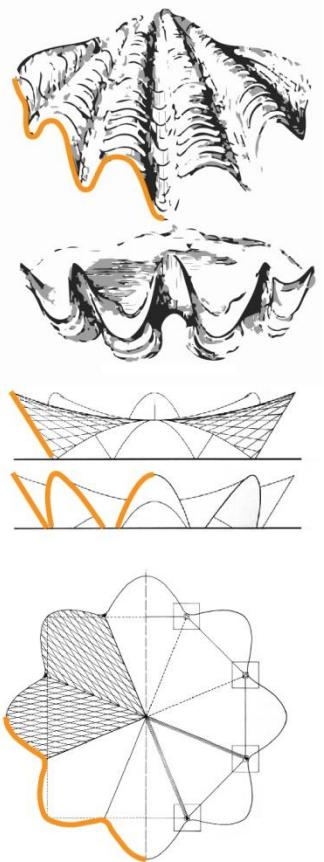


Fonte: Adaptado de Archdaily, 2018.

Campos de investigação dos comportamentos da natureza, como a biomimética é outro exemplo da busca por estratégias baseadas na Natureza nas práticas projetuais desses profissionais.

O conceito de biomimética consiste no desenvolvimento de *design* ecologicamente correto visando à sustentabilidade. Seu desenvolvimento iniciou-se durante os anos 1970, quando pesquisas realizadas no *New Alchemy Institute*, antigo centro de pesquisa fundado por John Todd, Nancy Jack Todd e William McLarney, em Massachusetts, começaram a explorar como a ecologia, a biologia e a abordagem dos sistemas biocibernéticos poderiam

Figura 4: Inspiração e desenhos de Candela



Fonte: Adaptado de Archdaily, 2018

fornecer soluções mais sustentáveis para atender às necessidades humanas.

Apesar desses esforços, Nagy (2018) acredita que devemos dar um passo adiante da simples reprodução de formas e comportamentos da Natureza e começar efetivamente a projetar como ela.

Não se deve entender ‘projetar como a Natureza’ no sentido de uma intencionalidade projetual, como aquela existente na arquitetura. O sentido de projeto empregado por Nagy, neste caso, está na forma como a Natureza responde ao processo que determina as características formais de uma espécie quando influenciada pelo ambiente no decorrer de um determinado tempo.

Descrito pela primeira vez por Charles Darwin em seu livro *A origem das espécies* (1994), este processo opera no nível de uma espécie que codifica as características, propriedades e habilidades que todos seus membros compartilham entre si.

Uma das características mais importante delas é sua capacidade de reprodução, de criarem novos indivíduos. Isso porque, com o tempo, esse processo melhora, continuamente, as espécies por meio da adaptação e interação, num processo conhecido como seleção natural. Este processo é composto de três etapas:

- (1) – seleção: etapa onde os indivíduos de uma espécie competem entre si,

e somente aqueles melhores adaptados sobrevivem;

- (2) – reprodução: período onde os sobreviventes se reproduzem transmitindo algumas de suas características aos seus descendentes;
- (3) – mutação: estágio onde algumas das características descendentes são alteradas aleatoriamente.

Para Nagy (2018), mesmo que ainda não possuamos as ferramentas e o conhecimento necessário para projetar sistemas totalmente naturais, conceitos matemáticos nos dão uma maneira concreta de reformular o processo evolutivo como fundamentalmente um processo de otimização.

A maioria dos problemas de otimização, de acordo com Yang (2010), pode ser descritos na forma genética:

$$\min_{x \in R^n} / \max f_i(x), \quad (i = 1, 2, \dots, M),$$

$$\text{sujeito a } h_j(x) = 0, \quad (j = 1, 2, \dots, J),$$

$$g_k(x) \leq 0, \quad (k = 1, 2, \dots, k)$$

Sendo $f_i(x)$; $h_j(x)$ e $g_k(x)$ funções do vetor de *design*:

$$x = (x_1, x_2, \dots, x_n)^T$$

Sendo os componentes x_i e x chamados de variáveis de projeto. As funções $f_i(x)$ onde $i = 1, 2, \dots, M$ são chamadas de funções objetivas.

Podemos classificar a otimização como monoobjetiva, quando $M = 1$, ou multiobjetivo, quando $M > 1$. O espaço atingido pelas variáveis de decisão é comumente chamado de espaço de projeto R^n , enquanto o espaço formado pelos valores da função objetivo é chamado de espaço de solução e as igualdades h_j e desigualdades g_k são chamadas de restrições (YANG, 2010).

Problemas de otimização podem ser operados por diferentes abordagens, sendo a mais comum a utilização de algoritmos. De maneira sintética, podemos classificar os algoritmos de otimização em duas categorias:

- (1) – métodos determinísticos:
aplicação direta de uma série de etapas definidas;
- (2) – métodos estocásticos: introdução de algum nível de aleatoriedade enquanto buscam uma solução.

O método escolhido depende da maneira como o problema de otimização é descrito e do tipo de informação que se tem sobre as funções objetivas e as restrição do problema. Para este trabalho, o método que nos interessa são os Algoritmos Genéticos (GA), método estocástico que particularmente, nos interessa porque suas regras e operações são inspiradas na evolução darwiniana e na seleção natural (SHIFFMAN, 2012).

Em Ciência da Computação, GA se refere à técnica de buscar um resultado específico para um dado problema. Muito do que pensamos hoje como

algoritmo genético foi desenvolvido por John Holland, cujo livro *Adaptation in Natural and Artificial Systems* (1975) foi pioneiro na pesquisa deste campo de estudo, apresentando os fundamentos teóricos e explorando aplicações desde agentes da teoria econômica até projeto de dispositivos complexos como turbinas de aeronaves e circuitos integrados. Hoje, os GAs fazem parte de um campo mais amplo de pesquisa, frequentemente chamado de Computação Evolucionária que compreende o ramo da inteligência computacional e da computação natural (SHIFFMAN, 2012).

Na arquitetura, um dos métodos que se utiliza desse tipo de algoritmo é o chamado *design* gerativo. O *design* gerativo pode ser mais amplamente definido como o emprego de um sistema gerador (um conjunto de regras, um programa de computador, um conjunto de transformações geométricas, um diagrama, etc.) no processo de *design* através do qual o projeto final emerge. O emprego do sistema gerador tem como finalidade explorar, de maneira ágil e eficaz, novas possibilidades do *design*, dentro de um universo de possibilidades imensuráveis.

Na seção seguinte será abordado um dos métodos para reformular a prática de arquitetura do projeto de um único objeto (o indivíduo), para o projeto de um sistema, aquele que codifica todo um conjunto de possibilidades finais (a espécie), capaz de dotá-lo da capacidade de se expandir em novas oportunidades de *design* e que permita que um sistema gerador o explore.

2.4 Projeto do Projeto

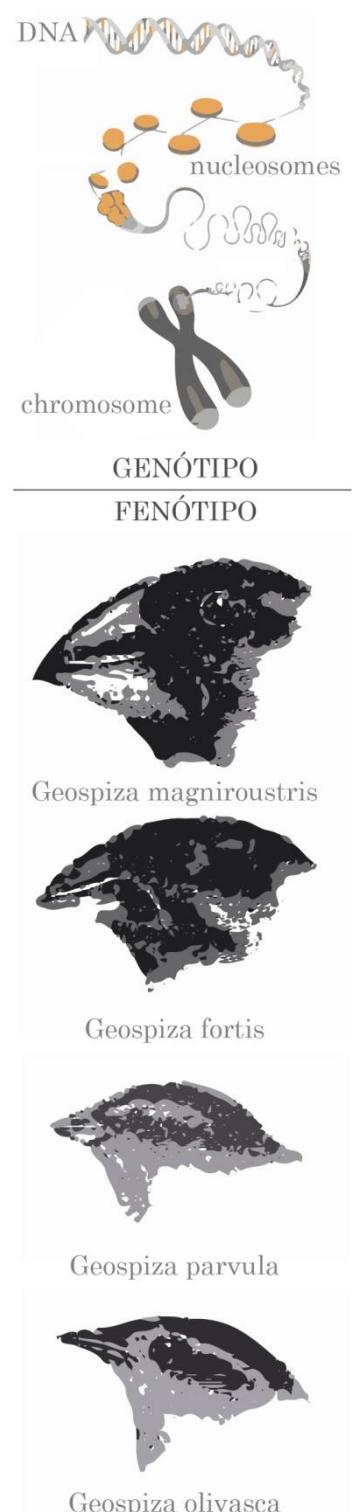
Para projetar como a natureza, precisamos considerar como podemos projetar uma espécie (NAGY, 2018) e uma das relações mais importantes no sistema de *design* da natureza para este trabalho é sua relação entre genótipo e fenótipo (Figura 6).

Parte do estudo da genética, os conceitos de genótipo e fenótipos são fundamentais para a compreensão de diversos fenômenos e fatores atribuídos aos seres vivos, aos mecanismos de evolução, às questões de hereditariedade e paternidade.

Criados pelo pesquisador dinamarquês Wilhelm L. Johannsen, em 1909, o termo genótipo se refere ao conjunto de genes que um indivíduo possui (JUSTINA *et al.*, 2010). Os genes de um organismo é seu *deoxyribonucleic acid* (DNA) e este se faz presente em todas suas células. O DNA codifica todas as informações para o desenvolvimento do organismo. Já o termo fenótipo atribui às características físicas do genótipo, o que inclui a morfologia do organismo e seu comportamento.

Dessa forma, há uma relação direta entre o fenótipo e o genótipo, sendo que o fenótipo de um organismo é o produto da expressão de seu genótipo e sua interação com o seu ambiente. O processo biológico que faz com que um organismo se desenvolva e resulte no fenótipo é chamado morfogênese.

Figura 6: Genótipo x Fenótipos



Fonte: O autor, 2018.

Na arquitetura, modelos de geometrias mutáveis podem ser especialmente construídos através de softwares de modelagem paramétrica.

Devido à ambiguidade e à própria imprecisão do termo, definir o que é modelagem paramétrica e o que a torna tão singular é, para Davis (2013) e Menges (2005) passo fundamental para identificar os desafios idiossincráticos que o acompanha.

Parte dessa ambiguidade está no fato de que os arquitetos sempre produziram modelos paramétricos, uma vez que todo o projeto, necessariamente, deriva de parâmetros, tais como aqueles adotados no auxílio a análises de viabilidade econômica e financeira; das normas referentes aos índices urbanísticos de uso e ocupação do solo; das propriedades dos materiais entre tantos outros.

Apesar disso ser verdade, a questão fundamental do desenho paramétrico não é a presença de parâmetros em si mas, sim, que esses parâmetros definem um conjunto de ações que permite ao profissional definir o modelo. Esta nova abordagem de projeto permite que os parâmetros sejam alterados em qualquer etapa do processo, sem a necessidade de se refazer os passos já realizados e sem perder a consistência do conjunto.

Um dos softwares paramétricos mais conhecidos, o Grasshopper®, interface de programação visual – *Visual Programming Language* (VPL) – que opera a partir de um editor baseado em nós – *node based editor* – usa um

conjunto de parâmetros para conduzir uma série de operações geométricas para produzir uma geometria final.

Na modelagem paramétrica, os projetos não são mais desenhados, mas sim programados a partir de relações geométricas e matemáticas. Isso permite a construção de espaços mutáveis, também possibilita projetar como a natureza, pois amplia a definição de projeto como indivíduo – projeto não mutável – para o projeto como sistema – aquele que codifica toda uma espécie – dentro da concepção de projeto baseado na natureza, colocada por Nagy (2018).

Essa nova compreensão não deve configurar uma transformação radical frente às produções cotidianas recorrentes, mas necessita incorporar às teorias desenvolvidas dentro do seu próprio campo específico e outras que viabilizem a ampliação de seus limites. Nesta perspectiva, o processo de evolução da natureza, por causa da sua ênfase no desempenho, nos fornece um ótimo modelo para o *design*.

Construindo o projeto como um problema de otimização, a aplicação de um algoritmo genético nos permite explorar o processo de projeto em arquitetura a partir do uso de algumas das potencialidades do *design* natural que discutimos anteriormente.

3 FERRAMENTA E ESTRATÉGIA PARA O *DESIGN GENERATIVO*

O método desta pesquisa é baseado no projeto, sendo, então, o procedimento metodológico nesta etapa a aplicação do processo evolutivo em um estudo de caso.

O objetivo deste experimento é aprofundar a discussão da temática proposta, levantando dados a partir do projeto, para posteriormente discuti-los.

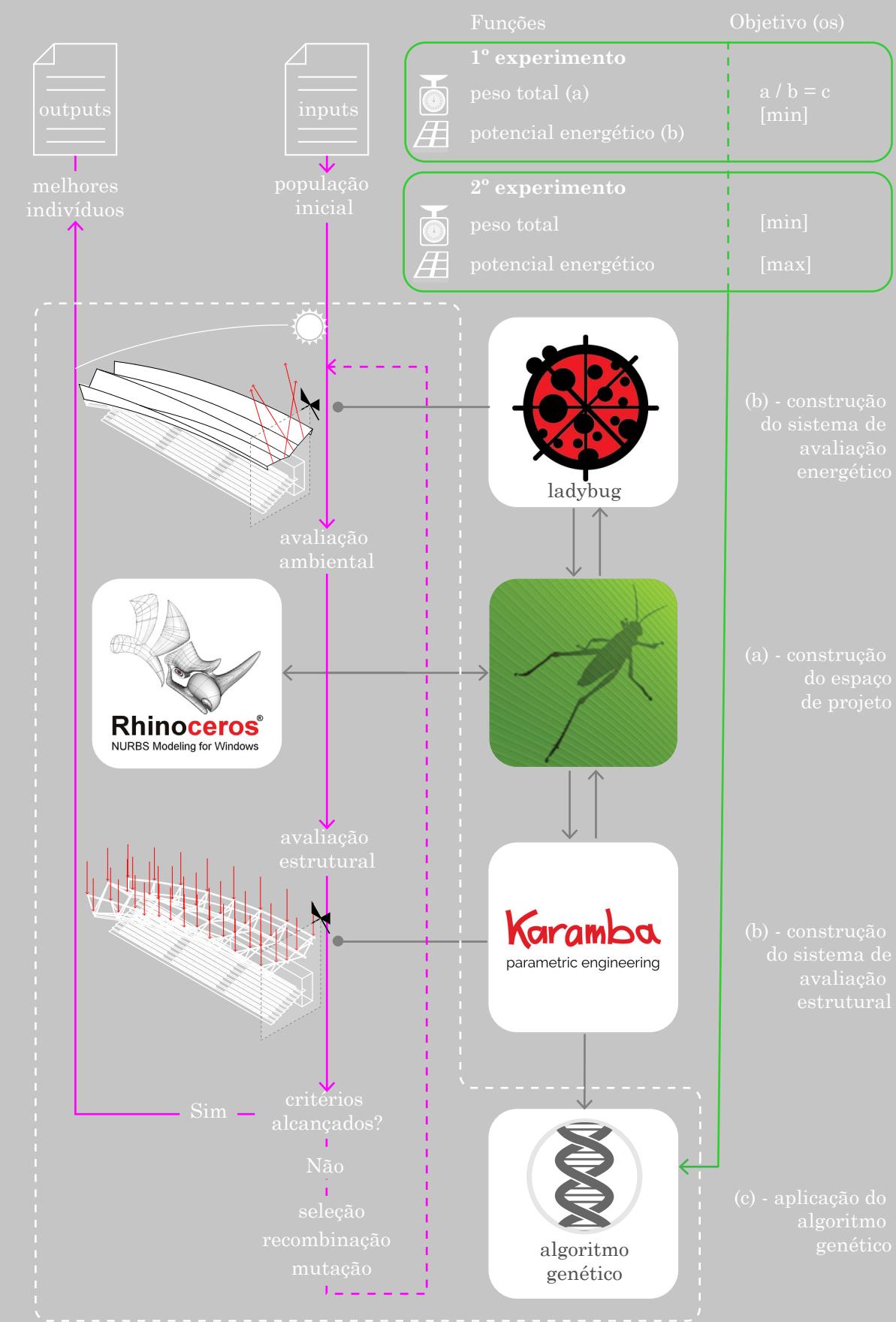
Esta fase se desdobrou em outras três partes (Figura 7): (a) – a construção do espaço de projeto; (b) – o desenvolvimento de medidas para avaliar o desempenho estrutural e ambiental de cada projeto e (c) – a aplicação de algoritmos evolutivos para pesquisar o espaço de projeto e encontrar projetos de alto desempenho.

Para isso, foi utilizado o *software* Rhinoceros®, para modelagem tridimensional e seu *plug-in* de modelagem paramétrica Grasshopper®.

Também foram aplicados os seguintes *addons* para Grasshopper®: (a) – Karamba® – para análise estrutural; (b) – Ladybug® – para análise ambiental e (c) – Discover® – um *framework* para projeto generativo.

Em seguida, realizou-se a análise quantitativa dos dados obtidos, através de gráficos de dispersão e da seleção dos melhores indivíduos (projetos) e, logo após, as discussões e ponderações foram organizadas e descritas.

Figura 7: Diagrama de aplicação do processo generativo



Fonte: O autor, 2018.

4 IMPLEMENTAÇÃO E EXPERIMENTOS

Para este trabalho, escolhemos como demanda arquitetônica hipotética a requalificação do campo de futebol da Universidade Federal de Ouro Preto, localizado no campus Morro do Cruzeiro (Figura 8), na cidade de Ouro Preto – MG.

Um campo de futebol tem como propósito óbvio receber partidas de futebol. Isso envolve, além dos equipamentos destinados a receber os atletas, outras demandas, como por exemplo, equipamento para acomodar os espectadores. Dessa forma, este exercício tem início com o desafio de implantar uma arquibancada coberta para 1.134 assentos. Esses assentos foram distribuídos em dez patamares de um metro (1,0) de profundidade por oitenta metros (80,0) de comprimento cada.

Apesar de ser uma necessidade real, tendo em vista que o campo de futebol da instituição não dispõe de tal equipamento, o foco deste exercício não é o projeto da arquibancada em si, mas sim a sua cobertura e seu potencial energético por meio da instalação de placas fotovoltaicas. Por isso, não foram priorizadas todas as exigências funcionais, recomendações técnicas e parâmetros de dimensionamentos, segurança e de conforto existentes para a construção de arquibancadas em estádios de futebol.

4.1 A Geometria: construindo o espaço de projeto

A proposta da cobertura partiu da ideia básica da superfície plana para uma forma tridimensional, composta de dobraduras planas no

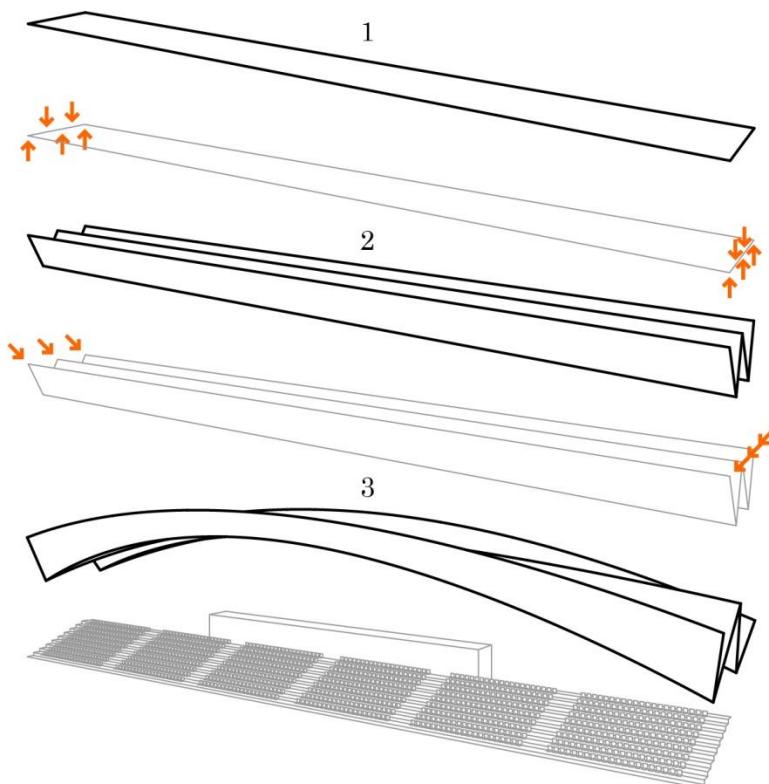
Figura 8: Localização
Ouro Preto



Fonte: O autor, 2018.

sentido transversal e curvatura simples no sentido longitudinal (Figura 9). O objetivo desse formato é de assegurar maior liberdade de forma, garantir maior área de superfície além de possibilitar uma melhor eficiência estrutural para a tarefa de cobrir o grande vão proposto.

Figura 9: Diagrama de concepção



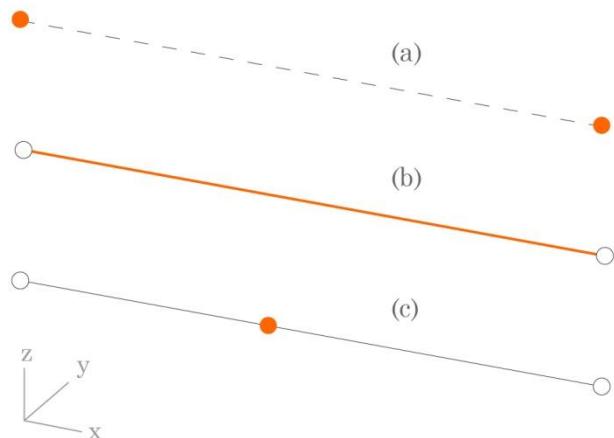
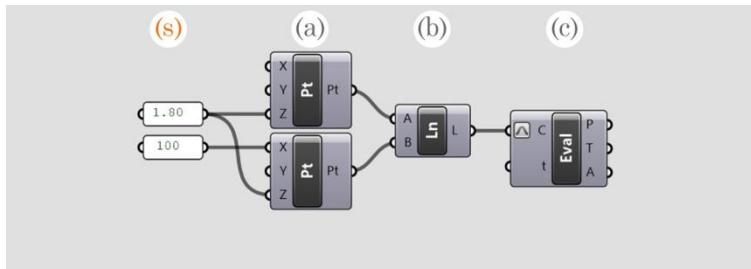
Fonte: O autor, 2018.

Seguindo os princípios da arquitetura paramétrica, que se refere à automação de parâmetros, iniciou-se a construção do modelo a partir da determinação espacial do primeiro par de apoios. Para isso, construíram-se os dois pontos (Figura 10b) atribuindo-lhes valores fixos às suas coordenadas (Figura 10s).

A determinação do primeiro par de apoios serviria como base para a construção dos demais

elementos do modelo (fluxo de informações), característicos na construção de sistema paramétrico.

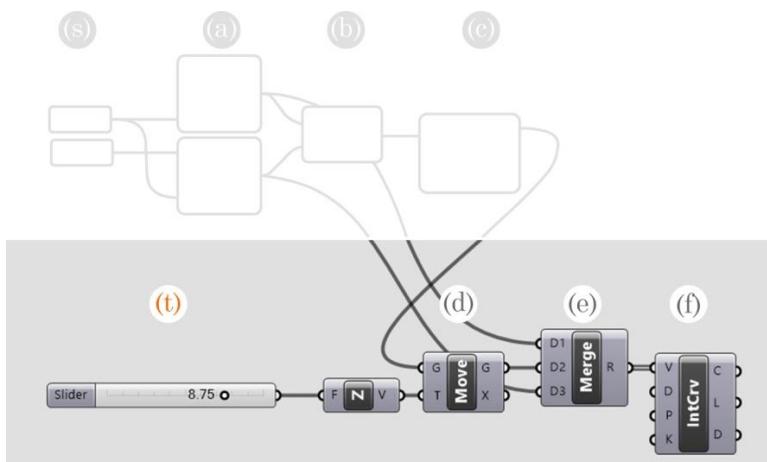
Figura 10: Construção dos pontos iniciais do modelo



Fonte: O autor, 2018.

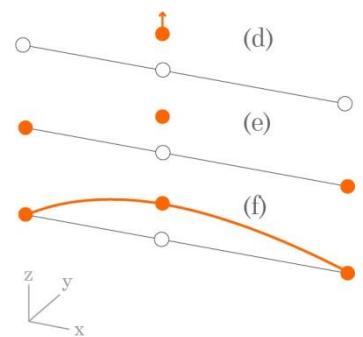
A partir destes pontos, definiu-se a primeira parábola (Figura 12f), determinada pelo ponto médio (Figura 10c), do modelo.

Figura 11: Construção da primeira parábola do modelo



Fonte: O autor, 2018.

Figura 12: Diagrama visual da construção da primeira parábola do modelo



Fonte: O autor, 2018.

Ressalta-se uma diferença importante no valor de atribuição ao terceiro ponto. Diferente dos anteriores, a ele, foi-lhe atribuído valor móvel como parâmetro de entrada (t) ao eixo (z).

Em seguida, outras duas parábolas foram construídas (Figura 13h). Para elas, foram atribuídos parâmetros de entradas nas suas extremidades – eixo y e z - e no ponto médio – eixo x e no eixo y (ângulo).

Uma quarta parábola (Figura 13j) foi construída atribuindo-lhe parâmetros de entradas para o controle (Figura 13i) de seu ponto médio – eixo z – e nas suas extremidades – eixo x e z.

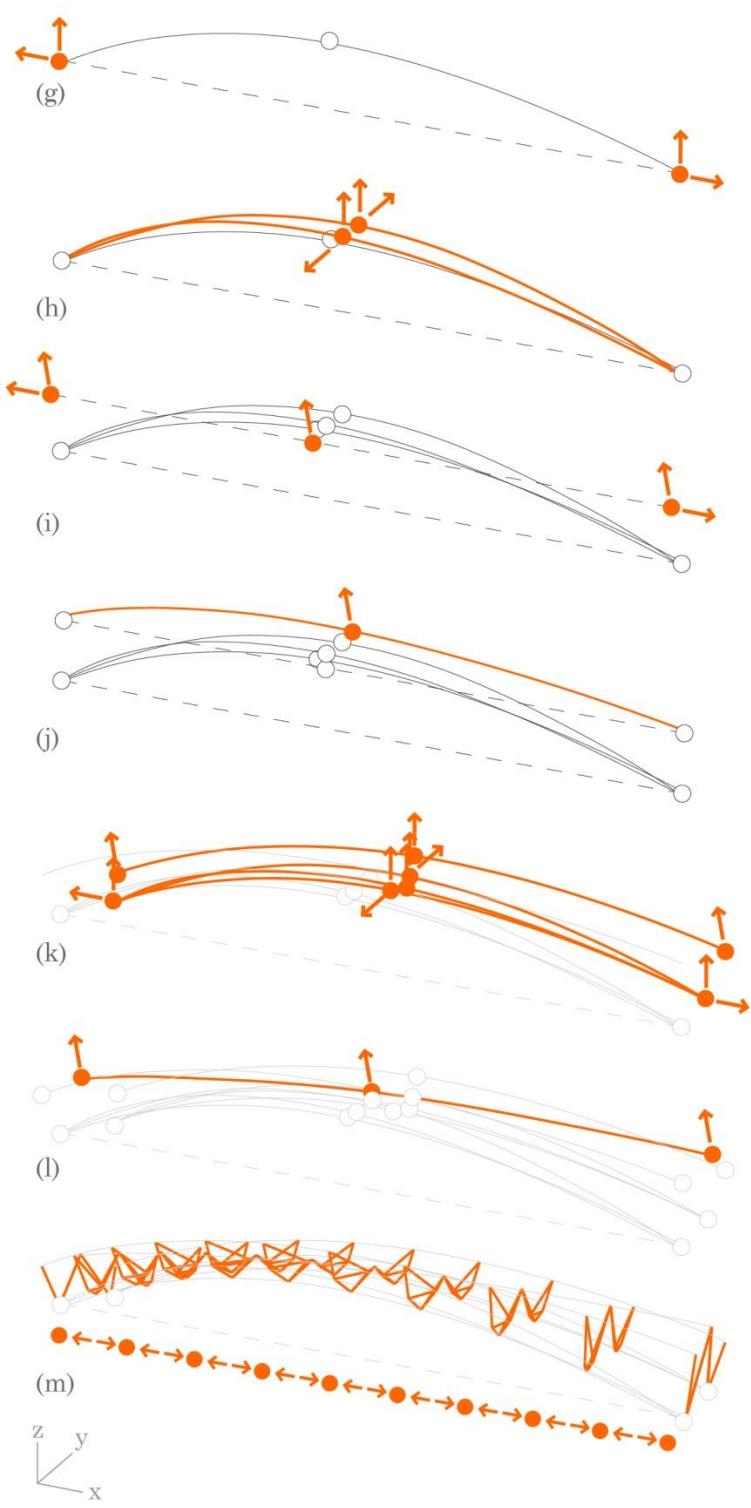
Ao final, todo esse processo é refeito para um segundo conjunto de parábolas (Figura 13k), constituídas a partir de um novo par de apoios a 7,0 metros de distância, no eixo y, do par anterior.

Com a intenção de integrar os dois conjuntos, uma nova parábola foi construída ao meio, atribuindo-lhe parâmetros de entradas para o controle (Figura 13l) de seu ponto médio – eixo z – e nas suas extremidades – eixo x e z.

Em seguida, ambos os conjuntos de parábolas são subdivididos e ligados por diagonais (Figura 13m).

Para finalizar, cada um dos quatro tipos de barras são controladas a partir de uma tabela com as 30 seções tubulares circulares mais frequentemente produzidas pela empresa *Vallourec*.

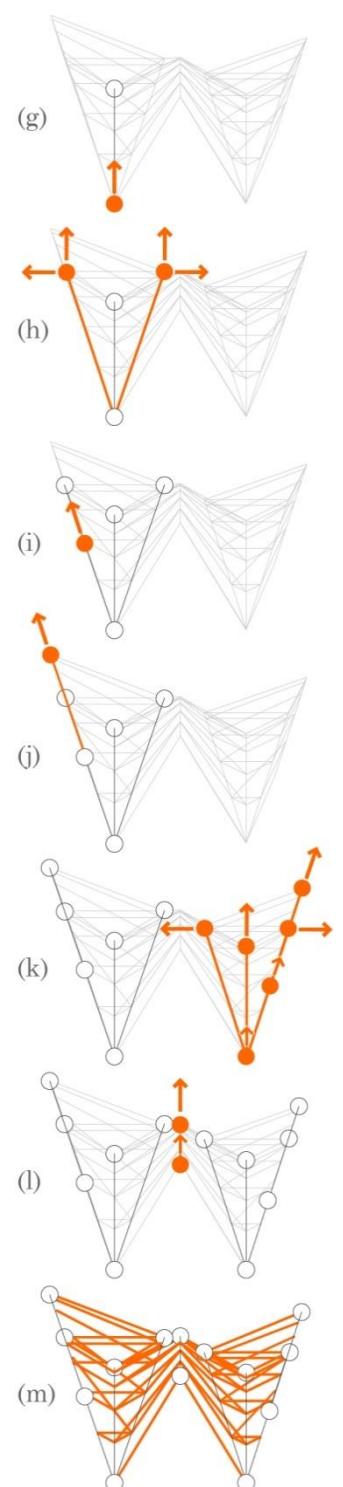
Figura 13: Construção dos demais elementos do modelo



Fonte: O autor, 2018.

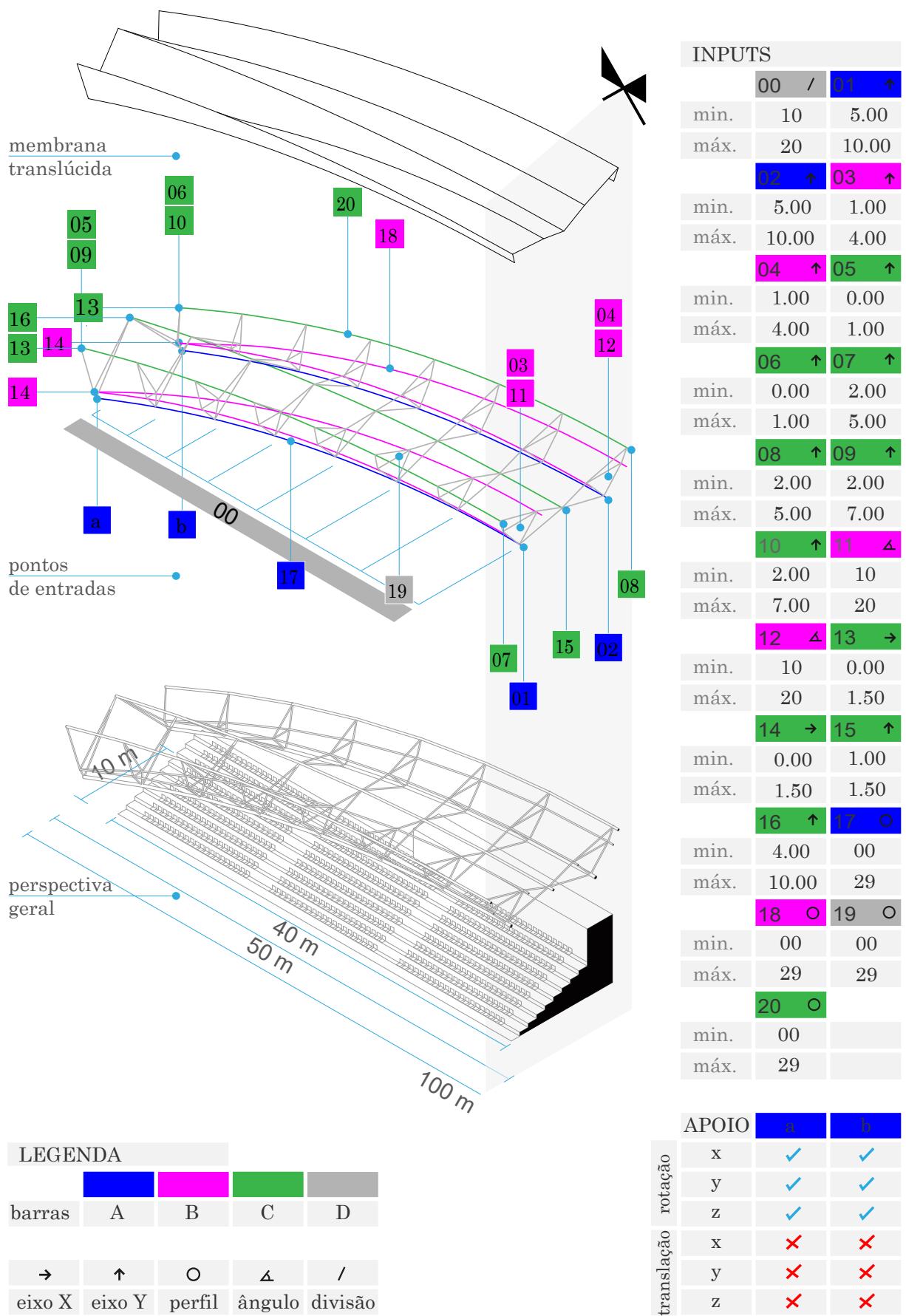
A Figura 15 seguinte descreve todos os elementos e seus respectivos parâmetros de entradas.

Figura 14: Vista lateral da construção do modelo



Fonte: O autor, 2018.

Figura 15: Diagrama do modelo e controles paramétricos



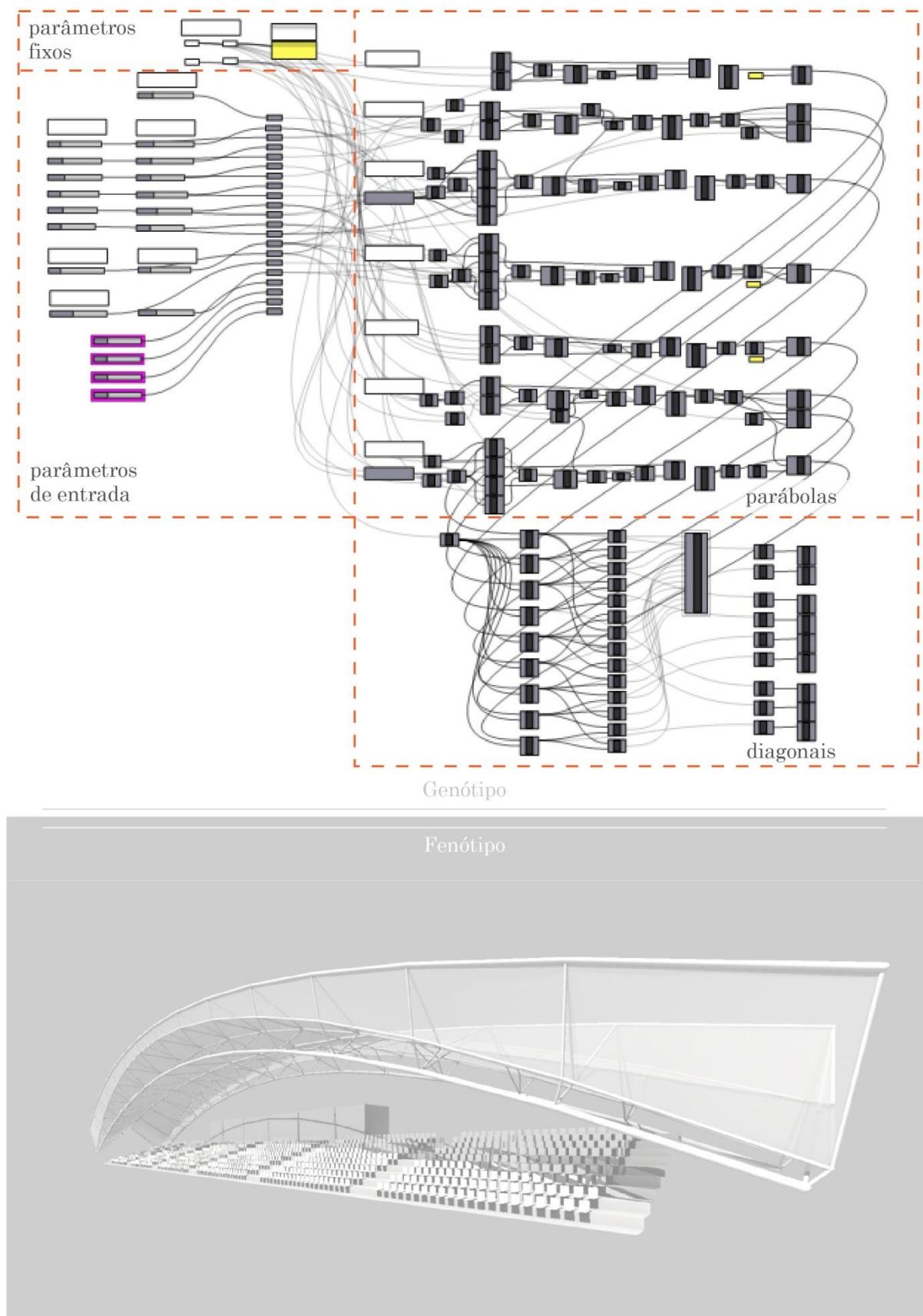
O espaço de projeto deste modelo possui 21 entradas com valores que variam de 10 a 500 caracteres.

Cada combinação dessas entradas especifica um ponto dentro do espaço de projeto que representa um projeto único formado por essas entradas. Ainda que a complexidade possível não seja imediatamente aparente, o total de soluções possíveis deste espaço de projeto é de $2,21 \times 10^{42}$.

O teste manual de cada uma das hipóteses é virtualmente impossível, uma vez que, considerando o gasto hipotético médio de 10 minutos por hipótese, seria necessário dedicar $1,76 \times 10^{25}$ vezes a idade do planeta Terra, considerando sua idade estimada em $4,54 \times 10^9$ anos (SHIFFMAN, 2012).

Para se viabilizar, o processo tradicional de projeto descarta virtualmente todas as soluções possíveis em favor das poucas que se alinhem com o conhecimento prévio e os gostos pessoais do projetista.

Em analogia com o processo evolutivo, os resultados do modelo mutável desenvolvido seriam os fenótipos da espécie *Estrutura da cobertura*, enquanto o conjunto dos parâmetros de entrada que define os modelos representaria seu genótipo, ou seja, o conjunto de cromossomos que descreve todas as possibilidades formais deste organismo (Figura 16).

Figura 16: Genótipo *vs* Fenótipo do modelo

Fonte: O autor, 2018.

4.1.1 Construindo a avaliação estrutural

Para aplicar o algoritmo genético na busca por projetos de alto desempenho, o modelo deve também conter um conjunto de medidas que informem ao algoritmo quais projetos têm melhores desempenhos. Devido à natureza do experimento desenvolvido, a análise estrutural foi considerada como etapa fundamental na elaboração deste sistema.

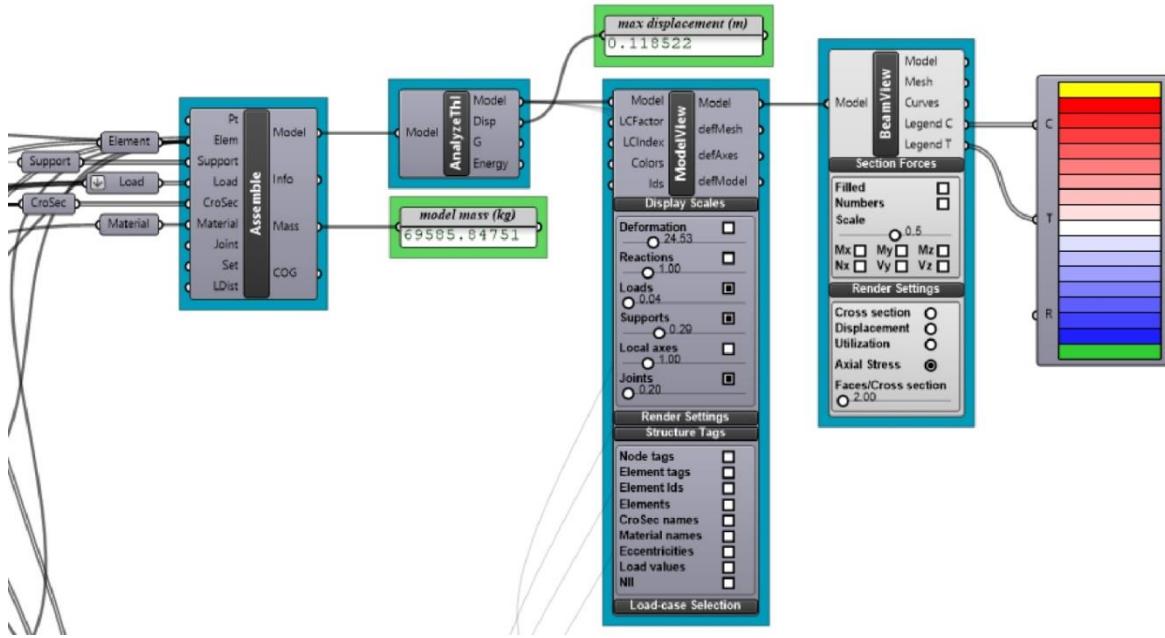
Para se obter a resposta da estrutura perante as ações que são aplicadas ao modelo, foi utilizado o Karamba®, *add-on* que calcula os deslocamentos e os esforços solicitantes nos elementos de modelo construídos no ambiente Grasshopper®.

Para o modelo estrutural do Karamba®, baseado no Método dos Elementos Finitos, foi utilizado o elemento de seções tubulares circulares para todos os elementos. Após a montagem do modelo estrutural, o Karamba® oferece, em tempo real, o *feedback* dos cálculos de deslocamentos e esforços durante as alterações de dimensões e resistência do material no modelo estrutura (PREISINGER, 2016). Essa simultaneidade entre a mudança da geometria e resposta do modelo estrutural é fundamental para a definição da proposta de restrição, que será abordada mais adiante.

Na Figura 17 são ilustra partes do bloco da modelagem estrutural composto por componentes do Karamba®, responsável pela conversão das

entidades geométricas em elementos de modelo estrutural.

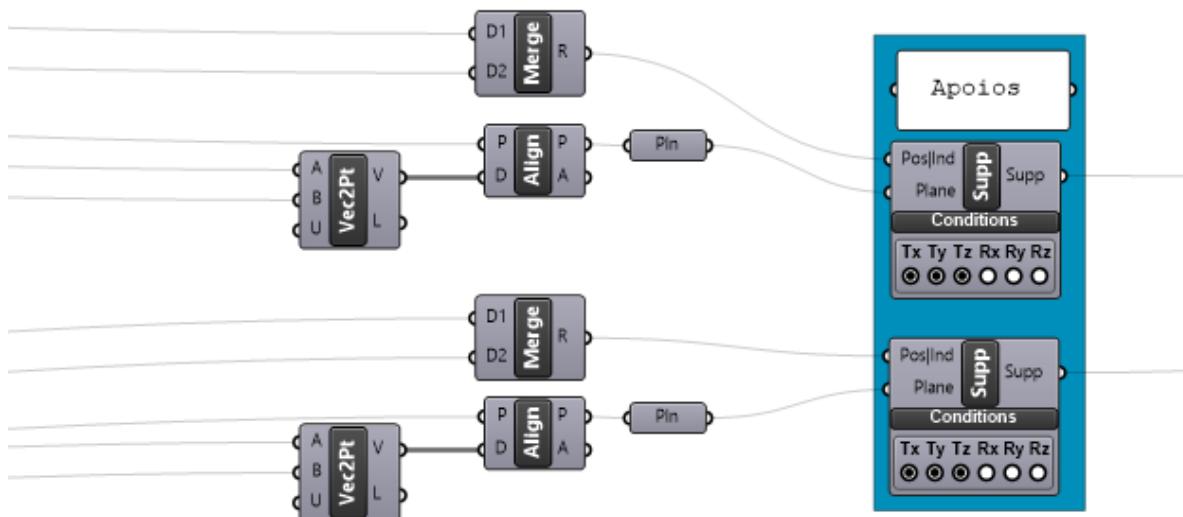
Figura 17: Partes dos componentes de modelagem estrutural Karamba®.



Fonte: O autor, 2018.

Na Figura 18 ilustra-se a configuração dos apoios (bases dos elementos A), admitindo liberdade para todos os graus de movimento nos eixos de rotação (R_x, R_y, R_z) e restrição para todos os graus de liberdades nos eixos de translação (T_x, T_y, T_z).

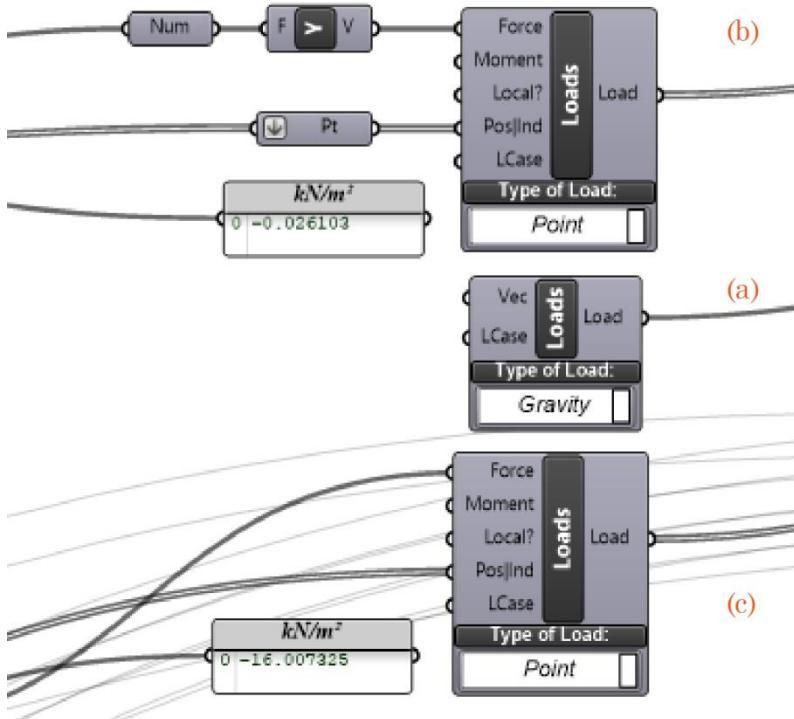
Figura 18: Configuração dos apoios – restrições e liberdades.



Fonte: O autor, 2018.

Na Figura 19 ilustra-se a definição das cargas, divididas em: peso próprio da estrutura (a), carga de vento (b) e cargas permanentes (c).

Figura 19: Definição das cargas



Fonte: O autor, 2018.

Para definição destas cargas foram consideradas as seguintes combinações (Figura 20) (ABNT, 2003):

$$comb_{01} = 1.4xCP + 1.4xCA$$

$$comb_{01} = 1.4xCP + 1.4xCA + 1.4x0.6xCV$$

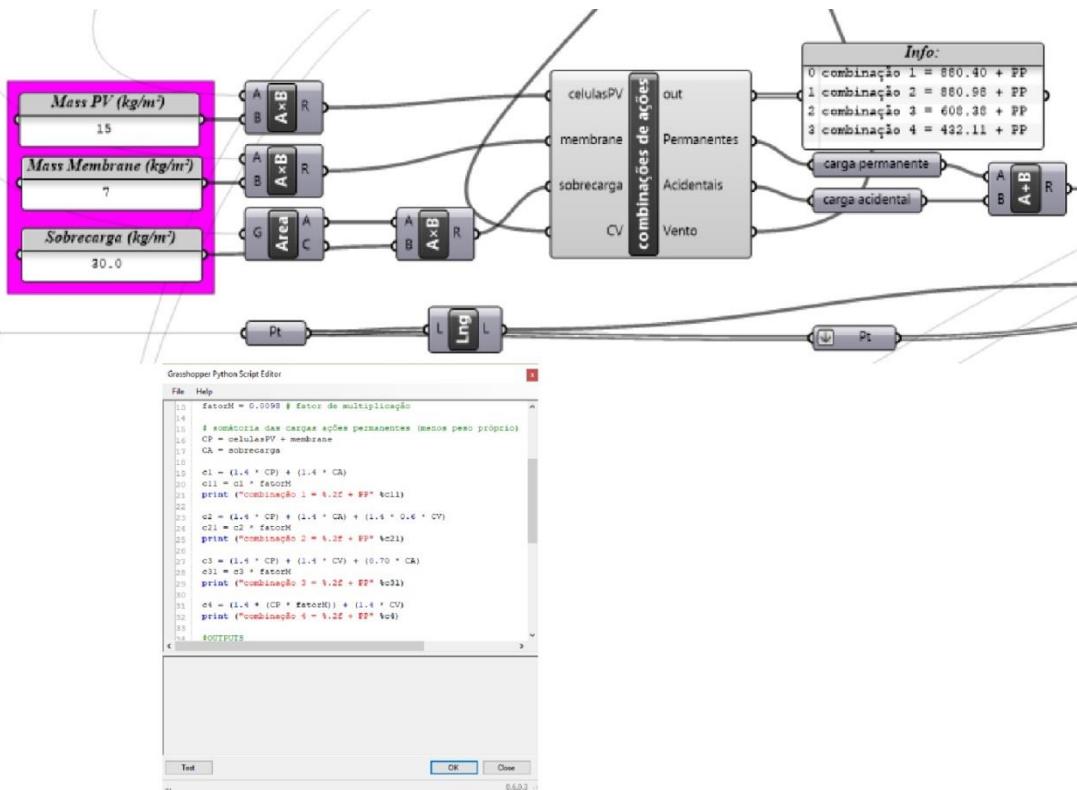
$$comb_{01} = 1.4xCP + 1.4xCV + 1.4x0.7xCA$$

$$comb_{01} = 1.4xCP + 1.4xCV$$

Sendo, CA a carga accidental, CV a carga de vento e CP a somatória das cargas permanentes das células fotovoltaicas e da cobertura.

As combinações foram calculadas sem o peso próprio, pois como este é constante para todas as combinações, não há prejuízo para o cálculo do Karamba®, além do que o próprio aplicativo se encarrega deste cálculo, levando em considerações outras entradas¹.

Figura 20: Componentes de cálculo das combinações de ações

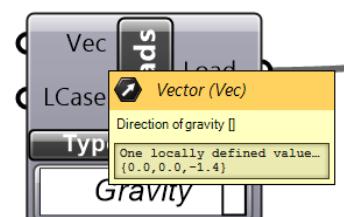


Fonte: O autor, 2018.

A partir da determinação da combinação mais exigente, os valores são inseridos nas entradas do Karamba®, corrigidos.

Para equilibrar o sistema, o valor do peso próprio da estrutura é reajustado com o valor de -1.4, no elemento *Load/Gravity*, responsável pelo cálculo deste item (Figura 21).

Figura 21: Reajuste na carga de Peso Próprio devido à combinação.

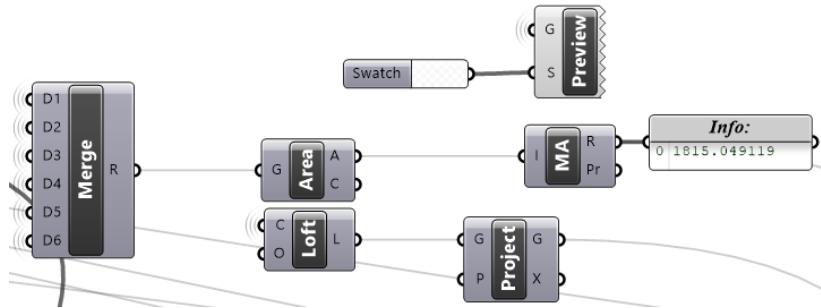


Fonte: O autor, 2018.

¹ Ver apêndice A

Para determinar parte das cargas permanentes, foi calculada antes à área da cobertura da membrana (Figura 22).

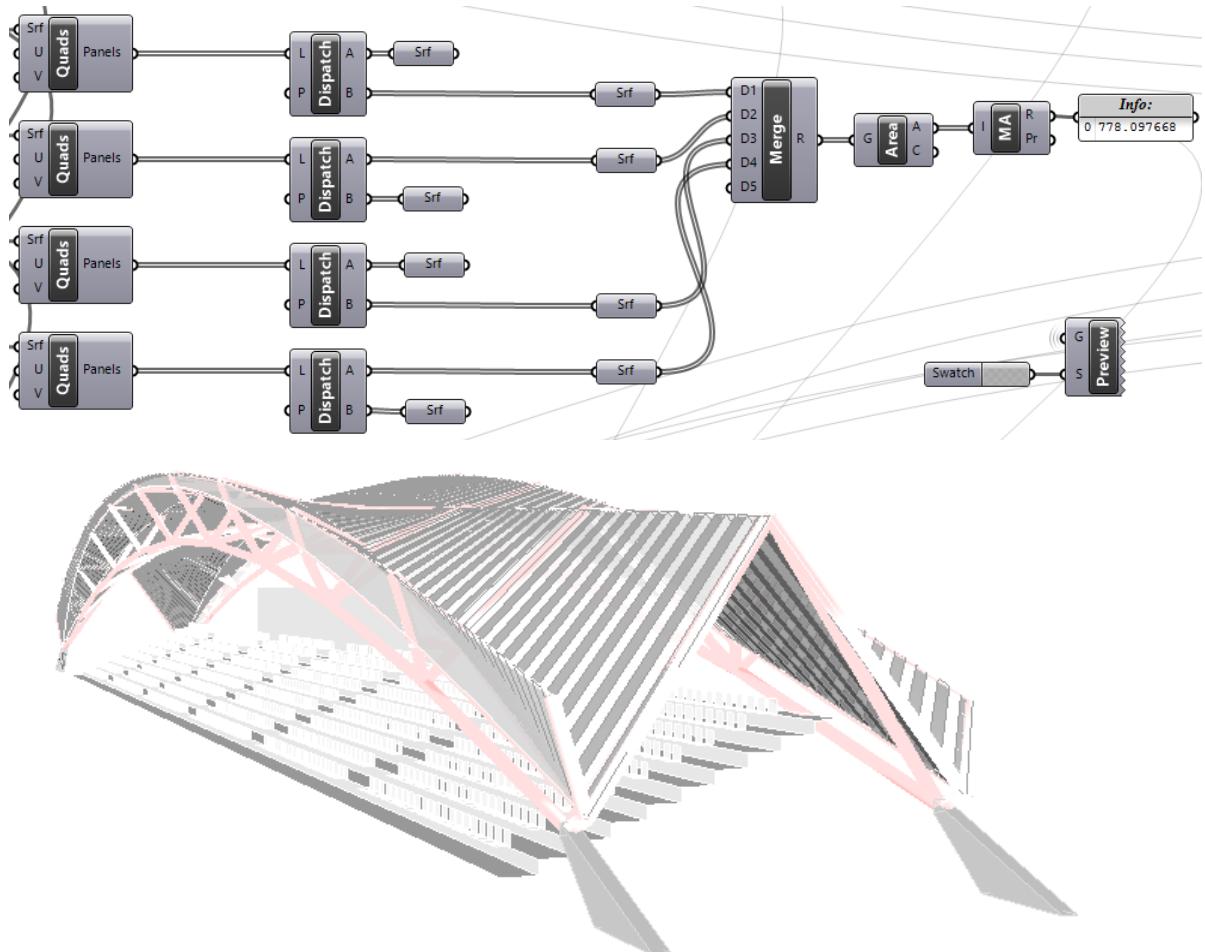
Figura 22: Componentes de cálculo da área de cobertura



Fonte: O autor, 2018.

Também foi calculada a área reservada para os painéis fotovoltaicos (Figura 23), construção apresentada na próxima seção.

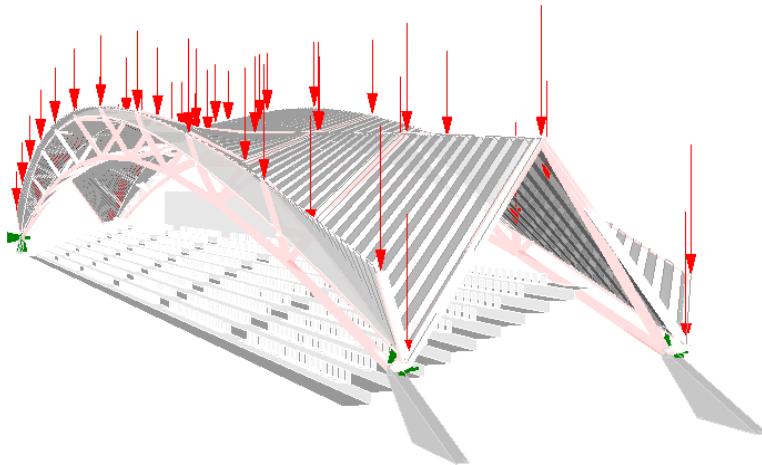
Figura 23: Componentes de construção e cálculo da área dos painéis fotovoltaicos (áreas cinza)



Fonte: O autor, 2018.

A figura 24 ilustrada as cargas permanentes e de sobrecarga distribuídas sobre os nós dos elementos D.

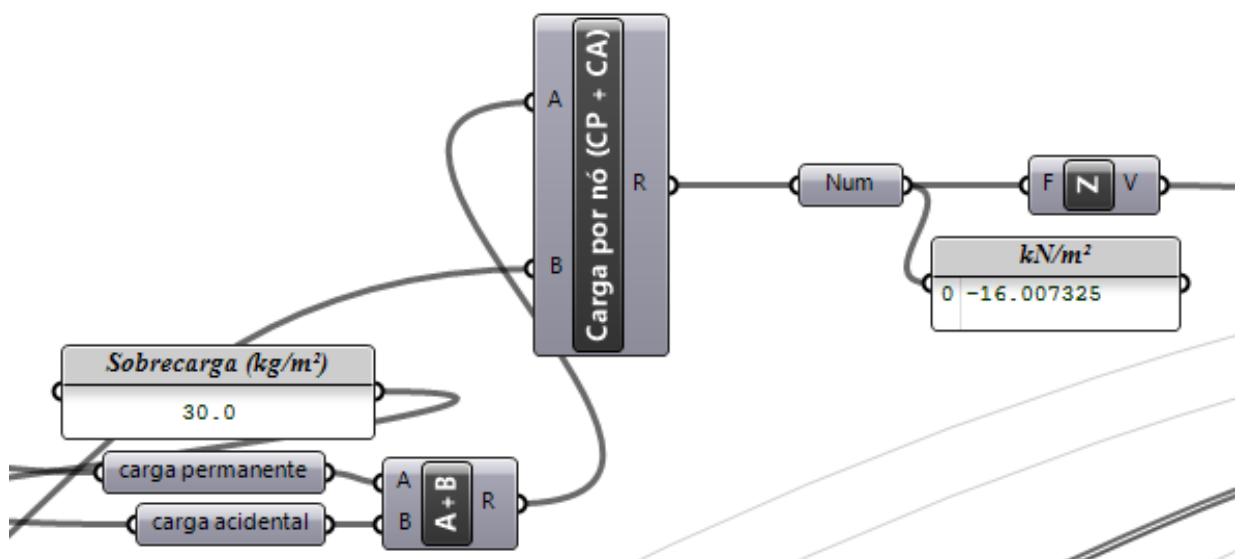
Figura 24: Ilustração das cargas nodais (permanentes)



Fonte: O autor, 2018.

Essas cargas foram convertidas para quilonewton (kN), divididas e distribuídas pelo número de nós dos elementos D (Figura 25).

Figura 25: Conversão, divisão e distribuição das cargas nos nós dos elementos D



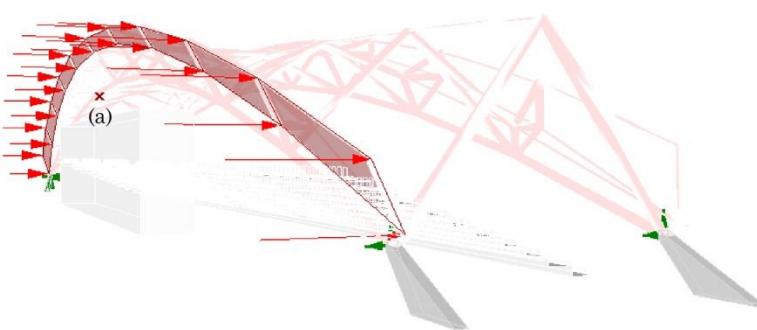
Fonte: O autor, 2018.

Para definir a carga de vento (CV), foram considerados dois pontos:

- (a) – determinar carga de vento em edifícios que não sejam retangulares e de duas aguas não é algo simples;
- (b) – este experimento é um processo de otimização e por isso precisa incorporar cálculos que sejam dinâmicos.

A partir disso, fez-se uma série de simplificações para determinar a carga de vento aplicando a NBR 6123 (ABNT, 1988). A primeira dela foi calcular a força de arrasto (F_a) apenas a 90° , o que corresponde ao maior comprimento da estrutura e lado mais ortogonal. A outra simplificação foi a determinação de uma altura média (a) para o modelo todo (Figura 26).

Figura 26: Área utilizada para calculo da força de arrasto



Fonte: O autor, 2018.

A ideia de atribuir um único ponto para altura tornou à construção do algoritmo mais fácil e o trabalho de otimização mais ágil, pois:

(a) – cada vez que a geometria mudar durante o processo de otimização, a força de arrasto (F_a) pode ser recalculada instantaneamente;

(b) – o ponto médio é referente à área (A_e) onde o vento atua, que neste caso, não é uma geometria regular e constante (se altera durante o todo o processo de otimização).

Para definir a força de arrasto (F_a) para carga devida ao vento (CV), foi considerado:

$$F_a = C_a q A_e$$

Sendo, C_a o coeficiente de arrasto (Figura 27), q é a pressão dinâmica do vento, calculado a partir da seguinte equação:

$$q = 0.613 V_k^2$$

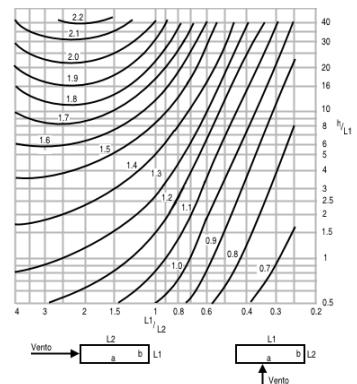
Sendo, V_k a velocidade característica, calculado a partir da seguinte equação:

$$V_k = V_0 S_1 S_2 S_3$$

Sendo, V_0 a velocidade básica do vento (Figura 28), S_1 é determinado pelo tipo de terreno, S_2 é determinado pela rugosidade do terreno e pela dimensão da superfície da edificação e S_3 é um fator estatístico que considera o grau de segurança e a vida útil da edificação.

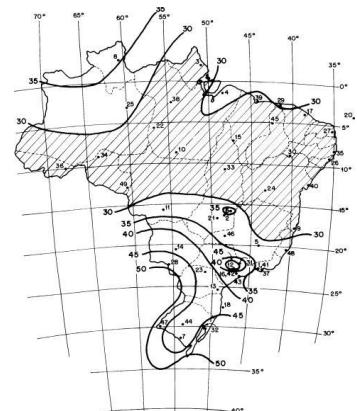
Na Figura 29 está ilustrado o bloco do algoritmo² de definição dos valores da carga de vento, convertendo a carga uniforme distribuída na fachada em cargas nodais.

Figura 27: Diagrama para coeficiente de arrasto



Fonte: NBR 6123 (1988)

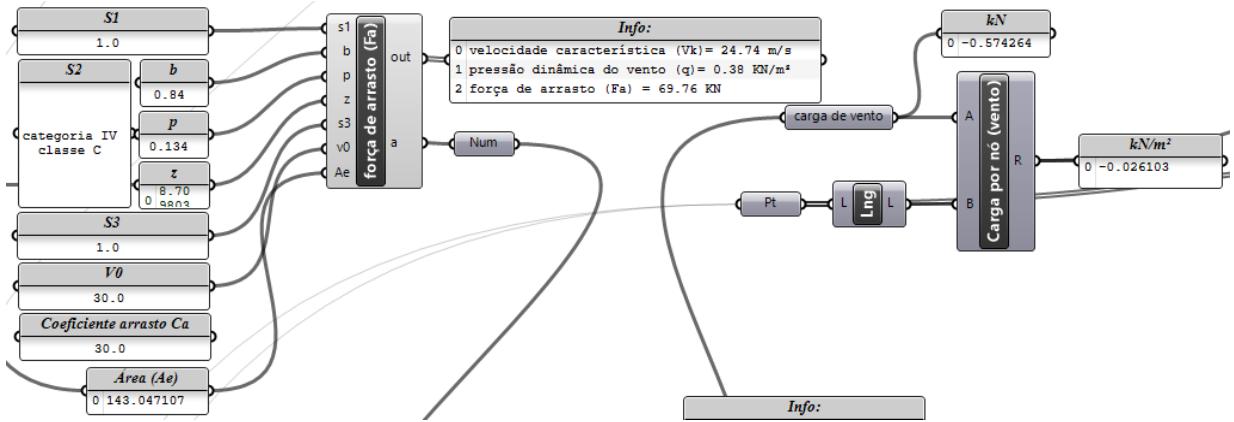
Figura 28: Velocidade básica do vento



Fonte: NBR 6123 (1988)

² Ver apêndice B

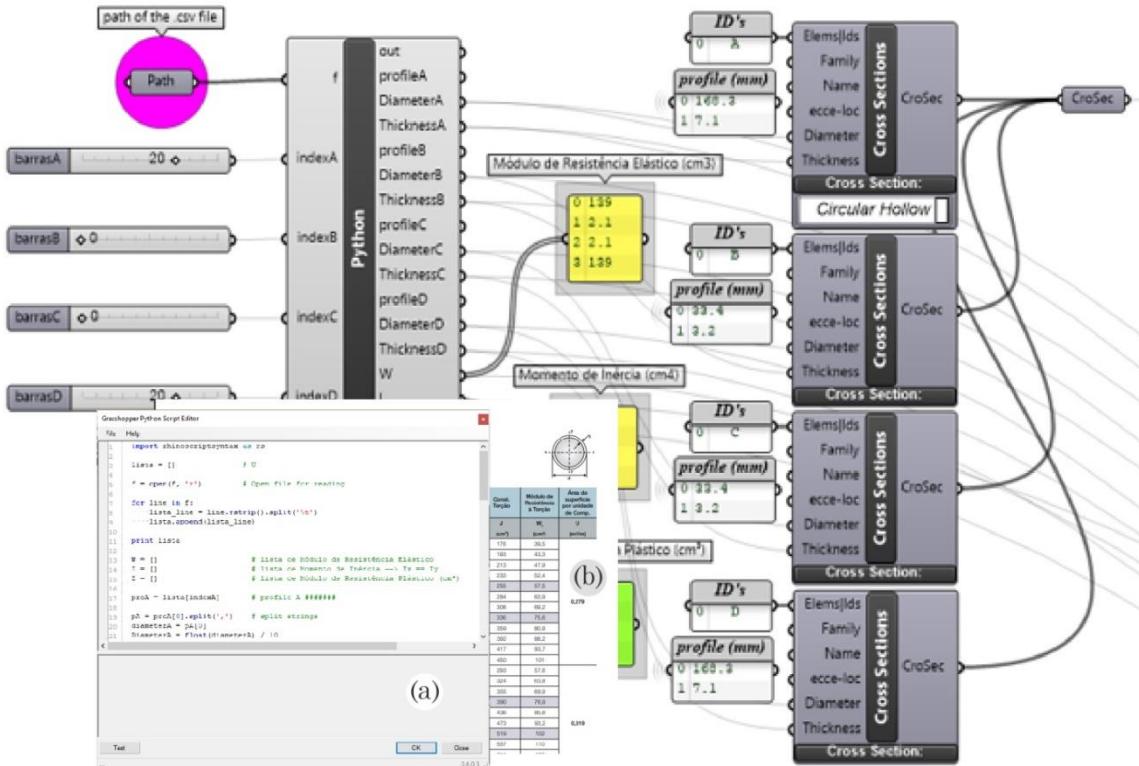
Figura 29: Algoritmo para definição do valor da carga de vento.



Fonte: O autor, 2018.

Por último, as seções transversais dos elementos estruturais (Figura 30), são escolhidas por um algoritmo³ (a) capaz de buscar 30 opções de seções tubulares circulares, seus diâmetros e demais informações geométricas e de cálculos, disponibilizados na tabela *Vallourec* (b).

Figura 30: Algoritmo de busca das seções transversais



Fonte: O autor, 2018.

³ Ver Apêndice C

Este algoritmo foi escrito de modo que as dimensões das seções transversais variem a cada iteração realizada pelo algoritmo de busca, alterando cada um desses parâmetros e consequentemente toda variação de projeto.

Como os resultados obtidos na análise estrutural influenciam diretamente na viabilidade do projeto, definimos como critérios para o projeto as seguintes condições:

- (1) – dimensionamento dos estados limites das barras submetidas à força axial de compressão, onde a força axial de compressão solicitante ($N_{c,sd}$) deve ser igual ou menor do que a força axial de compressão resistente de cálculo ($N_{c,rd}$);
- (2) – dimensionamento dos estados limites das barras submetidas à força axial de tração, onde a força axial de tração solicitante ($N_{t,sd}$) deve ser igual ou menor do que a força axial de compressão resistente de cálculo ($N_{t,rd}$);
- (3) – dimensionamento dos momentos fletores, onde o momento fletor solicitante (M_{sd}) deve ser igual ou menor ao momento fletor resistente (M_{rd});
- (4) – dimensionamento do deslocamento máximo, onde a flecha máxima (*Max. Displacement*) deve ser inferior

ao maior vão dividido por trezentos ($\frac{\ell}{300}$).

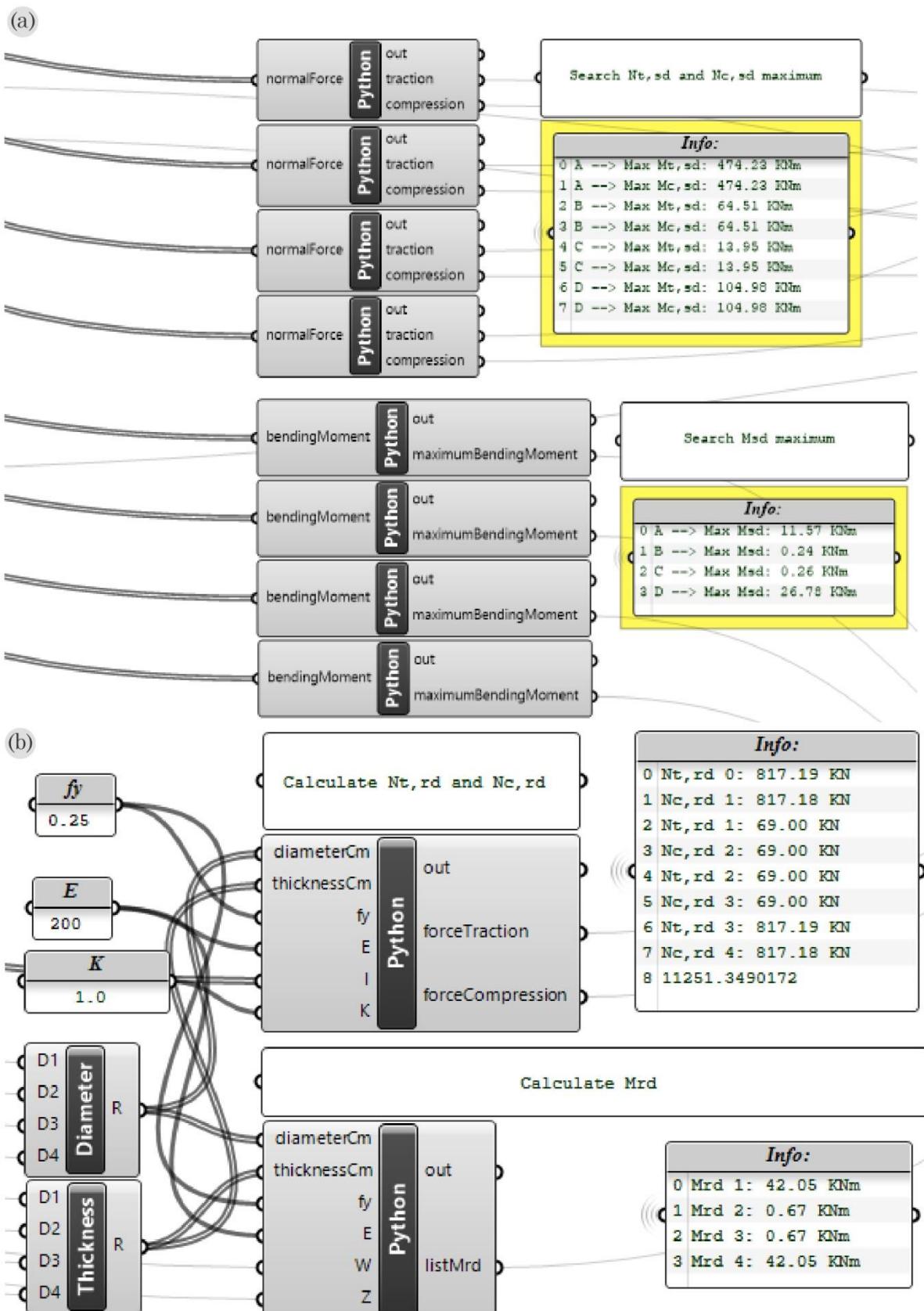
Para o dimensionamento no estado limite último, as ações de solicitações atuantes (S_d) foram todas majoradas (*multiplicadas por 1.10*) e as resistências do aço (R_d), por sua vez, foram minoradas (*divididas por 1.10*), conforme recomendação da norma NBR 8800 (ABNT, 2008):

$$S_d \cdot 1.10 \leq R_d / 1.10$$

Antes disso, foram criados blocos de algoritmos (Figura 31) para tratar as informações coletadas na escolha dos perfis e fornecidas pelo Karamba®, que funciona da seguinte maneira:

- (a) – as forças solicitantes (S_d) de cada conjunto de elementos, calculadas diretamente pelo *add-on* Karamba®, são extraídas – e separadas, no caso da tração e da compressão – e organizadas em ordem crescente (a);
- (b) – um segundo conjunto de blocos de algoritmos busca as informações extraídas do algoritmo para as escolhas das seções tubulares, explicado anteriormente, e calcula as forças resistentes (R_d) para cada conjunto de elementos (b).

Figura 31: Blocos de algoritmos de extração de forças S_d (a) e cálculo de forças R_d (b)



Fonte: O autor, 2018.

O tratamento adotado para lidar com as condições de restrições neste trabalho é de caráter punitivo, ou seja, aquelas opções que quebrarem as regras das restrições foram imediatamente punidos, desqualificados, o que os impediu de participar dos próximos cruzamentos (YENIAY, 2005).

O método de penalidade foi construído de duas maneiras. Primeiro na forma multiplicativa:

$$\text{eval}(x) = \begin{cases} f(x), & \text{if } x \in F \\ f(x)p(x), & \end{cases}$$

Sob essa convenção, o objetivo foi o resultado de $\text{eval}(x)$, onde $p(x)$ representa um termo de penalidade na lógica de verificação estrutural. Se nenhuma violação ocorrer, $p(x)$ será igual a zero. Caso contrário, desqualifica-se a opção atribuindo $p(x)$ igual a cem mil (100.000). Essa lógica foi aplicada na avaliação das análises de compressão ($Nc, rd \geq Nc, sd$), tração ($Nt, rd \geq Nt, sd$) e do momento fletor ($Mrd \geq Msd$).

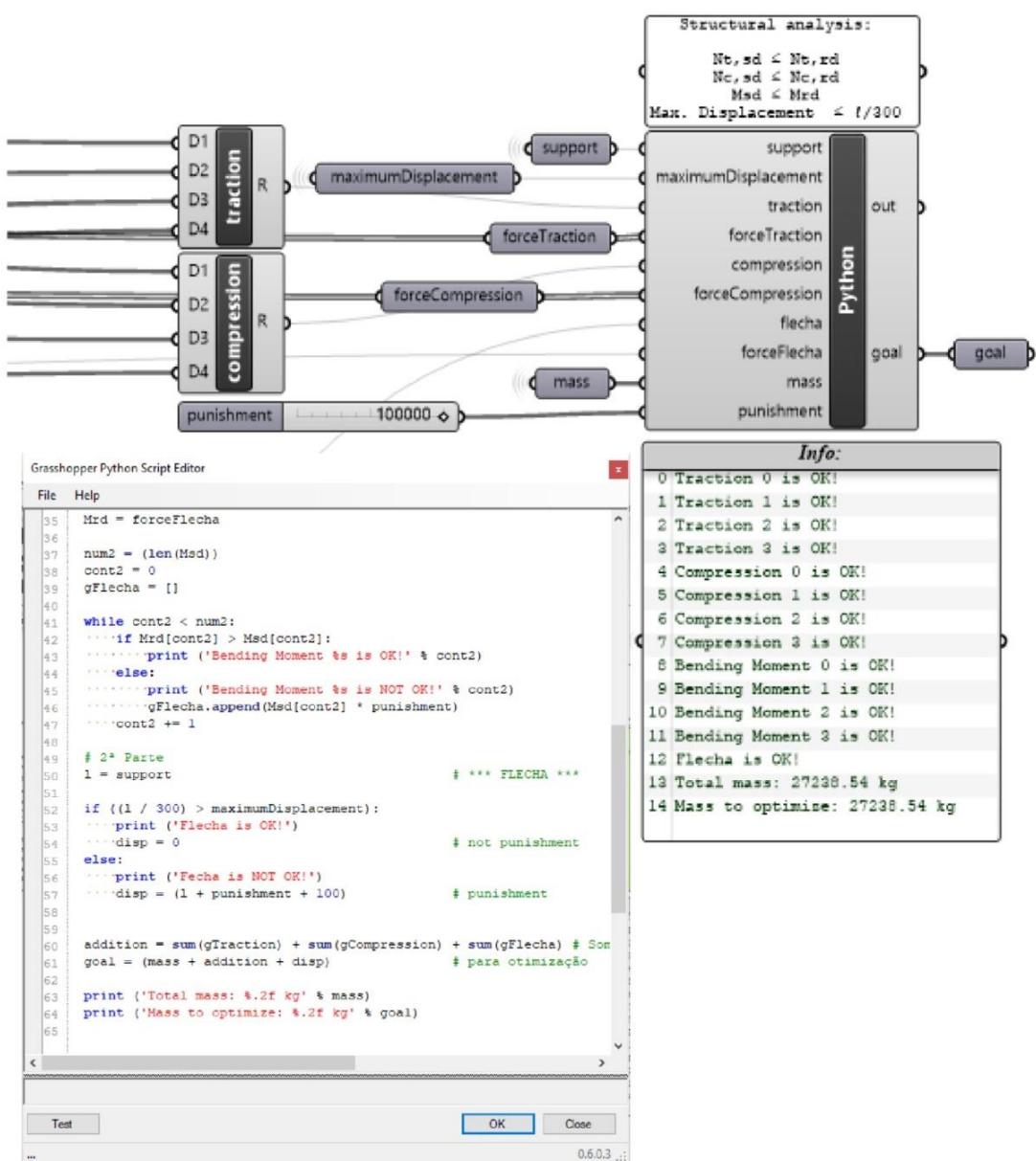
A segunda maneira ocorre na forma de adição. Esta convenção foi aplicada exclusivamente no critério de deslocamento ($\text{Max. Displacement} \leq \frac{\ell}{300}$):

$$\text{eval}(x) = \begin{cases} f(x), & \text{if } x \in F \\ f(x) + p(x), & \end{cases}$$

Apesar de não descreverem o desempenho relativo entre as opções de projeto, as restrições são importantes porque determinam se uma opção é viável ou não. Em nossa analogia aos processos evolutivos encontrados na natureza, as restrições funcionaram como um teste de sobrevivência para

os indivíduos (opções de projetos). Aqueles que foram reprovados nas regras foram eliminados do próximo cruzamento. As alternativas aprovadas, demonstraram aptidão ao contexto criado, o que significa que estes apresentaram características importantes para a sobrevivência da sua espécie. A Figura 32 mostra o bloco de algoritmo⁴ construído para as condições de restrições.

Figura 32: Bloco de algoritmo das condições de restrições, seus *impus* e seus *outputs*



Fonte: O autor, 2018.

⁴ Ver Apêndice D

4.1.2 Construindo a avaliação energética

Como vimos na seção anterior, este modelo usa análise de elementos finitos (FEA) para simular o desempenho de cada projeto sob as condições de dimensões e de cargas.

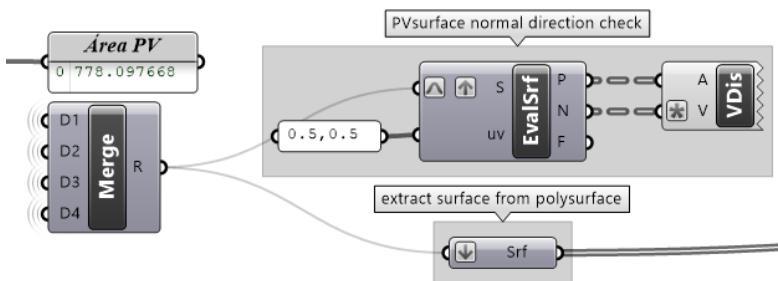
Aplicada esta restrição, temos segurança estrutural para acrescentar outros objetivos (problemas) para o sistema responder.

Como demonstração, acrescentamos um conjunto de nós para calcular o potencial energético da cobertura baseado no exemplo de Spasic (2015).

Esse objetivo foi construído com componentes do *plug-in* Ladybug®, de análise ambiental que permite importar e analisar dados meteorológicos no ambiente Grasshopper®.

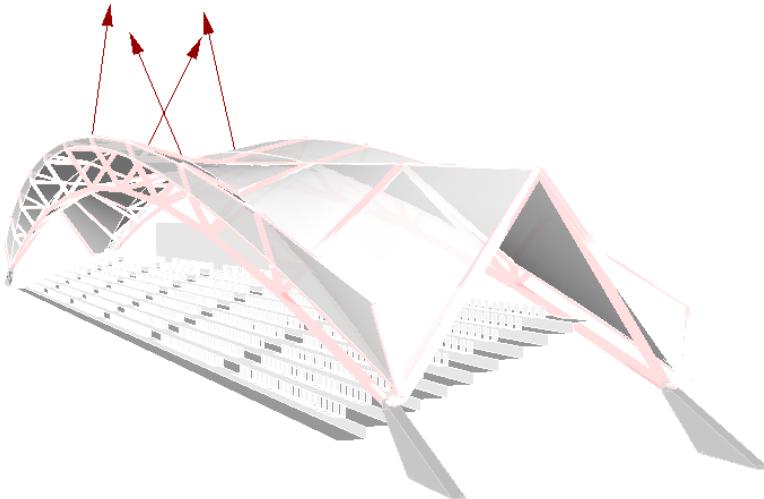
Para isso, o primeiro passo foi verificar a direção normal (Figura 33 e 34) das áreas dos painéis fotovoltaicos, cuja construção foi detalhada anteriormente (Figura 23, pág. 45).

Figura 33: Verificação da direção normal das áreas reservadas para os painéis fotovoltaicos



Fonte: O autor, 2018.

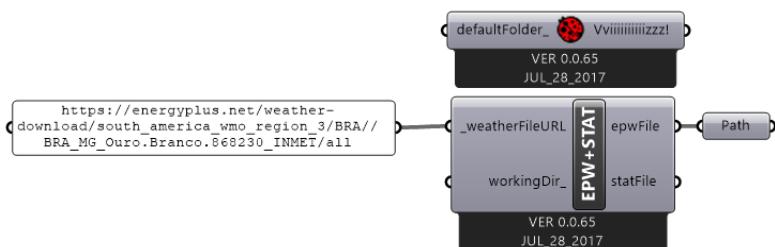
Figura 34: Direção normal das áreas reservadas para os painéis fotovoltaicos



Fonte: O autor, 2018.

Em seguida, foi usado o componente *EPW+START* para baixar, descompactar e ler o arquivo para o clima específico desejado do site do *United States Department of Energy* (DOE). Para isso foi copiado o *URL* do arquivo DOE e plugado ao componente, conforme a Figura 35.

Figura 36: Componente para leitura do clima específico



Fonte: O autor, 2018.

Como não existem arquivos para todas as cidades, escolhemos a cidade de Ouro Branco, Minas Gerais, levando em consideração as características climáticas parecidas (Figura 36).

Na Figura 37 ilustra-se o conjunto de nós aplicados para importar as configurações de módulo

Figura 35: Localização Ouro Branco x Ouro Preto



Ouro Preto

altitude
1179 m

clima
tropical de altitude



Ouro Branco

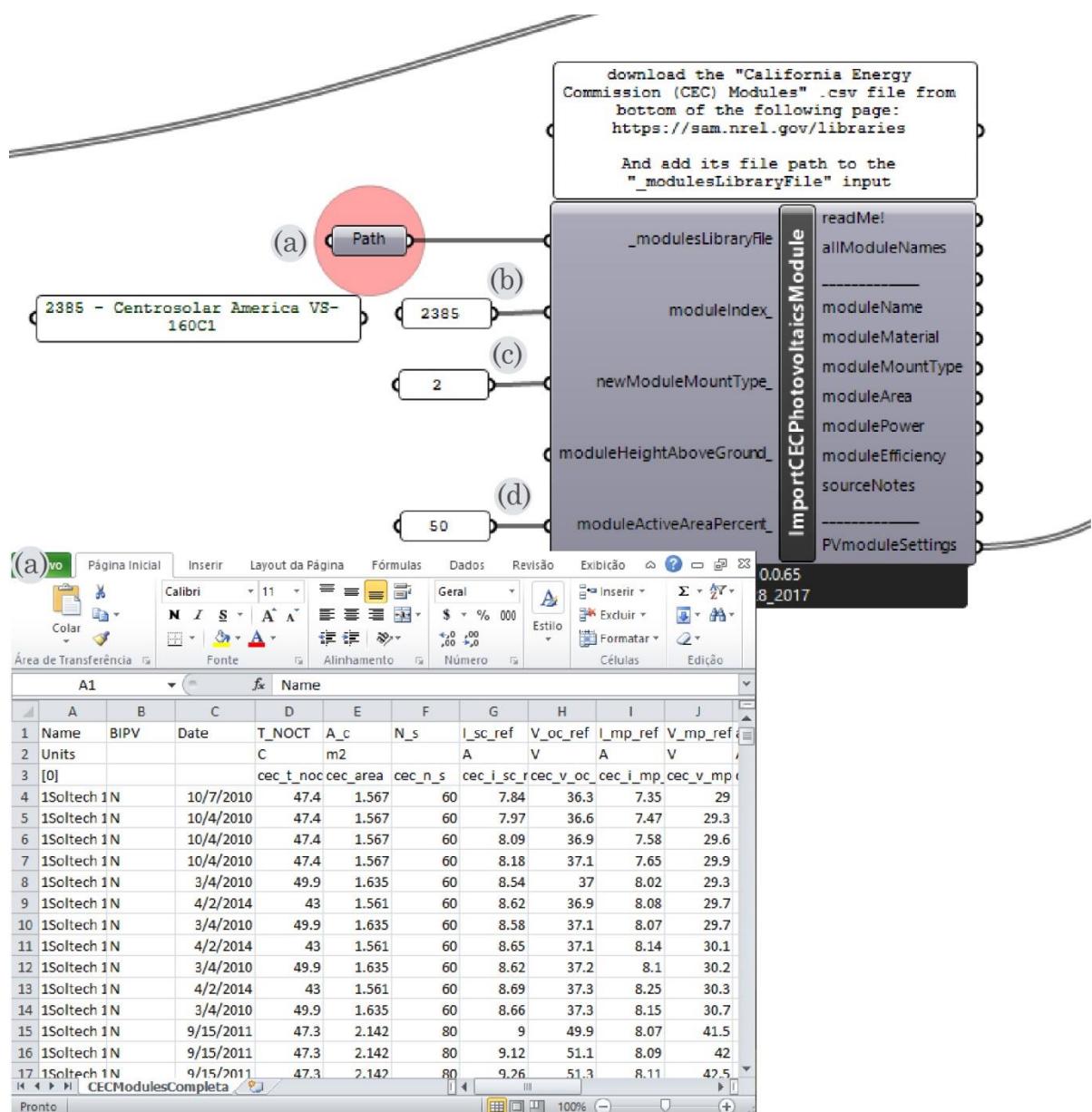
altitude
1100 m

clima
tropical de altitude

Fonte: O autor, 2018.

fotovoltaico específico da biblioteca *California Energy Commission (CEC) Modules* (a), disponível no *National Renewable Energy Laboratory*, laboratório do DOE, para calcular a quantidade de energia elétrica que pode ser produzida pela superfície. O módulo escolhido para esta fase foi a célula *Centrosolar America VS-160C1*, versão *NRELV1* (b).

Figura 37: Componente para importar as configurações de módulo fotovoltaico específico



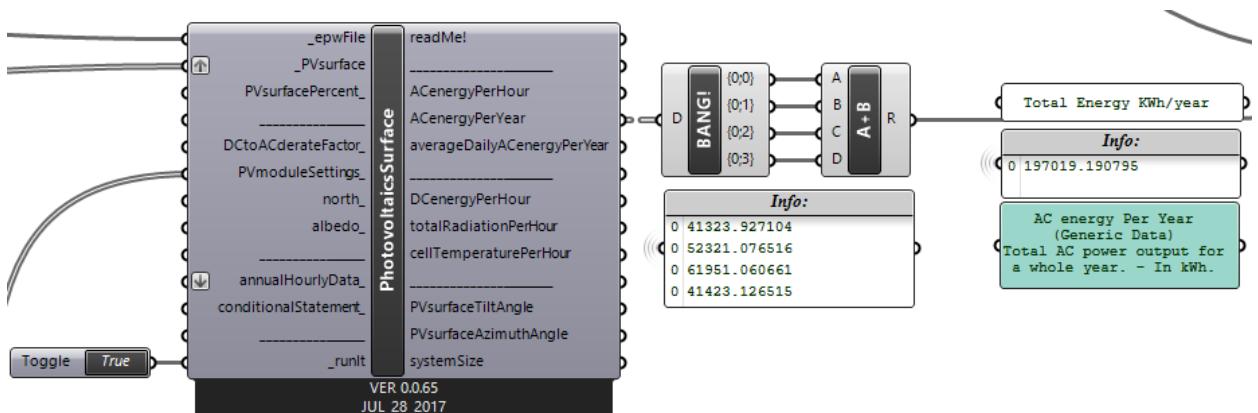
Fonte: O autor, 2018.

Para o experimento, a escolha do modelo de célula fotovoltaica levou em conta apenas algumas das características dos modelos, como por exemplo, a flexibilidade e o baixo peso.

Por fim, os dados climáticos e do módulo fotovoltaico específico importado alimentam o componente *Photovoltaics Surface* (Figura 38).

Este componente calcula a quantidade de energia elétrica que pode ser produzida por uma superfície se certa porcentagem dela for coberta com energia fotovoltaica e se baseia na calculadora *NREL PVWatts v1* que comporta, entre outros tipos de células, as fotovoltaicos de película fina, escolhida para o experimento.

Figura 38: Conjunto de nós para calcular a quantidade de energia elétrica que pode ser produzida por uma superfície durante um ano.



Fonte: O autor, 2018.

Para aplicar o algoritmo genético na busca por alternativas de alto desempenho, o modelo deve conter um conjunto de medidas definidas que

informam ao algoritmo quais opções têm os melhores desempenhos.

As análises construídas nos fornecem um conjunto de métricas que podemos usar para estabelecer os objetivos do nosso problema de otimização, que foram:

- (a) – peso total da estrutura (objetivo a minimizar);
- (b) – potencial energético da cobertura (objetivo a maximizar).

4.2 Algoritmo Generativo: o solucionador evolutivo Discover[©]

Para uma busca efetiva das opções de alto desempenho, precisamos de um sistema externo que possa trabalhar juntamente com o modelo. Uma das opções seria um algoritmo de busca. A vantagem desse tipo de algoritmo é que ele não precisa receber informações prévias sobre o funcionamento interno do modelo. Isso porque a estratégia geral desses métodos é iniciar uma amostragem aleatória e, em seguida, usar o conhecimento derivado dessa amostra para obter outras amostras com melhores desempenhos. Seguindo esse processo, o algoritmo busca as melhores opções sem saber nada sobre como o espaço de projeto funciona e sem ter que testar todas as alternativas possíveis (NAGY, 2018).

Os algoritmos meta-heurísticos são particularmente interessantes porque suas regras e operações são inspiradas nos processos evolutivos da natureza, o que nos permite explorar o *design* de maneira semelhante ao que ocorre no *design* natural (SHIFFMAN, 2012).

O significado de meta-heurístico relaciona-se ao conjunto de regras que orientam este processo. Nesse caso, as regras são meta porque são aplicadas a um algoritmo geral, e não a um problema específico. Isso significa que as abordagens meta-heurísticas podem ser generalizadas a qualquer problema de otimização (SHIFFMAN, 2012).

O Discover[©], de Danil Nagy, é uma estrutura flexível e modular para projeto generativo e

exploração do espaço de projeto baseado neste tipo de algoritmo. Especialmente adaptado aos problemas de *design* físicos, foi projetado para funcionar em conjunto com uma plataforma paramétrica, tal como o Grasshopper®. O Discover® consiste em uma biblioteca modular para otimização mono-objetivo e multiobjetivo, escrita em Python®, e uma interface escrita em JavaScript® que permite explorar visualmente o processo de otimização, capturando imagens de todas as opções.

Embora o Discover® seja uma boa ferramenta para resolver problemas complexos de otimização de projeto, o algoritmo é conduzido por apenas quatro operadores básicos, baseados na seleção natural:

- (a) – geração – o algoritmo gera uma amostra aleatoriamente dos designs do espaço de projeto formando a geração inicial;
- (b) – seleção – o algoritmo seleciona os indivíduos melhor posicionados para serem usados na próxima geração, criando assim, um "pool de acasalamento" contendo os melhores designs, de acordo com seu objetivo;
- (c) – cruzamento - os algoritmos selecionados como projetos promissores são recombinados para criar uma nova população de projetos;
- (d) – mutação – assegura que a busca explore soluções não contidas nas

gerações passadas por meio de variação pseudorrandômica.

Este caminho é bastante semelhante ao da evolução natural. Em ambas, as informações genéticas dos genitores são aleatoriamente recombinadas para criar um novo indivíduo. A ideia básica é que, uma vez que os genitores sobreviveram tempo suficiente para se reproduzirem, ambos devem ter algum material genético que possa ser útil para a sobrevivência da espécie em geral. Ao se recombinarem, provavelmente o novo indivíduo herdará características vencedoras de ambos os pais – fator de convergência, aumentando suas chances de sobreviver e de se reproduzir em uma próxima geração.

Entretanto, apenas com esses métodos, podemos ficar sem a melhor solução. Isso acontece quando as alternativas da primeira geração, realizada aleatoriamente, apresentam nenhuma ou poucas características fundamentais em sua genética, impedindo a transferência de informações importantes para a nova geração. Então, como ocorre na natureza, precisamos de um mecanismo que possa inserir novas informações de forma aleatória ao fundo genético. Isso é realizado pelo operador de mutação, responsável pela alteração das entradas de um número aleatório de filhos, normalmente uma pequena porcentagem, antes de entrar na próxima geração – fator estímulo à diversidade.

4.2.1 Configurando o Algoritmo Generativo

No Sublime Text 2®, um editor de código com interface de programação Python®, foi aberto o *template Run*, disponibilizado como parte do pacote Discover©. Neste *template*, foram configurados todos os parâmetros de entrada⁵ (linha 04). Abaixo, vemos o exemplo do primeiro *input* preenchido (linha 05):

```
[01] from src import job
[02] jobDescription = {
[03]     "jobName": "untitled",
[04]     "inputsDef": [
[05]         { "name": "input00",
[06]             "type\": continuous\",
[07]             "range": [10,20]},
[08] ...
[30]     ],
[31] }
```

Neste mesmo arquivo, também foram definidos os parâmetros de saída (linha 31). Nestas linhas, serão definidas as funções objetivas do experimento.

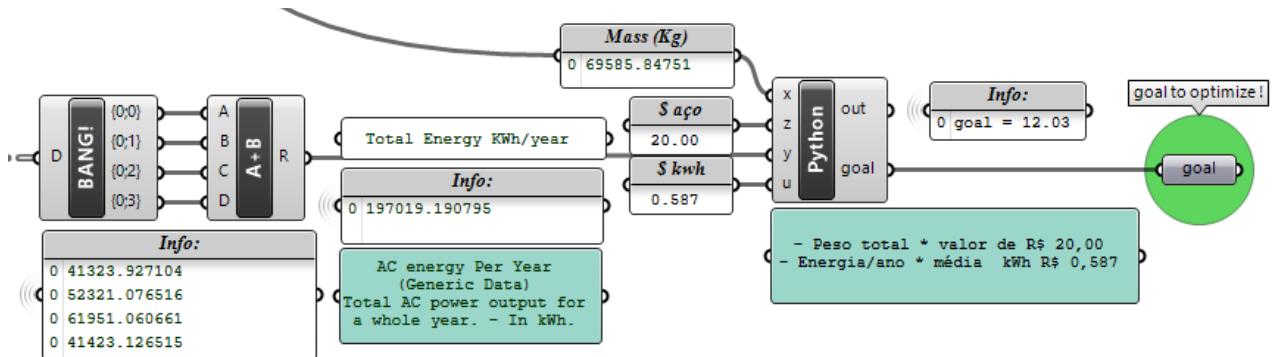
Na primeira etapa do experimento, os problemas que nos interessavam eram aqueles com apenas uma função objetiva, ou seja, aqueles descritos como mono-objetivo.

Para isso, foi construída uma função composta que deveria minimizar (linha 32) o resultado entre a divisão do produto da multiplicação entre o peso total da estrutura (em *kg*) e o preço médio do aço (em R\$) pelo produto da

⁵ Ver Anexo A

multiplicação entre o potencial energético (em *kwh*) e o preço médio do *kwh* (em R\$) (Figura 39).

Figura 39: Construção da função composta do primeiro experimento



Fonte: O autor, 2018.

```
[31] "outputsDef": [
[32]     { "name": "goal",      "type": "objective",
[32]       "goal": "min"},
```

Já para a segunda etapa, o peso total da estrutura e o potencial energético da cobertura foram divididos em funções objetivos distintas, criando um problema multiobjetivo⁶:

- (a) – minimizar o peso total da estrutura (linha 32).
- (b) – maximizar o potencial energético da cobertura (linha 33).

```
[32]     { "name": "weight",    "type": "objective",
[32]       "goal": "min"},
```

```
[32]     { "name": "weight",    "type": "objective",
[32]       "goal": "max"}
```

Definidos os parâmetros de entrada e saída, as opções necessárias para o trabalho geral foram preenchidas como se segue:

⁶ Ver Anexo B

- (a) – número de gerações: 100 (linha 40);
- (b) – número de população: 59 (linha 41);
- (c) – taxa de mutação: 0.05 (linha 42) e
- (d) – número de elite salvas: 10 (linha 43).

```
[38] algo: GA,
[39] "algoOptions": {
[40]     numGenerations": 100,
[41]     numPopulation": 59,
[42]     mutationRate": 0.05,
[43]     saveElites": 10,
[44]     DOE": random,
[ ... ]
```

Preenchido o *template*, o trabalho foi executar o algoritmo (linha 53).

```
[47] "jobOptions": {
[48]     "screenshots": True
[ ... ]
[52]#job.createInputFile(jobDescription)
[53]#job.run(jobDescription)
```

Executado o algoritmo, pode-se acompanhar o progresso do trabalho de otimização sendo exibido no editor de texto Sublime® e os desenhos sendo geradas automaticamente alterações como resposta na interface Rhinoceros®/Grasshopper®.

Quando a otimização foi encerrada, o pacote Discover oferece um arquivo do tipo *.tsv*, contendo o banco de dados de cada opção explorado no espaço de projeto durante a otimização. Este arquivo contém as informações de cada projeto, com seu ID exclusivo e todos seus *inputs* e *outputs*. Finalmente, também é disponibilizado um arquivo do tipo *.html* que é usado para iniciar a interface *Explorer*. Nesta interface foi possível visualizar os resultados

plotando-os em relação aos objetivos, em um gráfico de dispersão, como mostrado a seguir.

4.3 Análise do resultado – Evolução do Projeto

Os experimentos desenvolvidos geraram um conjunto de 11.800 opções diferentes, 5900 alternativas em cada etapa. À medida que foram analisados esse conjunto de resultados, percebeu-se como é frágil a certeza de uma resposta (solução) assertiva para um projeto. Isso acontece porque uma vez que uma decisão tenha sido tomada, acaba-se limitando grande parte do espaço de exploração das possibilidades nas fases posteriores, tornando altamente improvável que o processo de *design* alcance um projeto verdadeiramente "ótimo".

A implementação desta estratégia permitiu abrir mão do controle da definição específica de uma solução em favor da exploração de uma variedade maior de opções do que seria possível usando métodos de projeto tradicionais.

Nos experimentos apresentados, a perda de controle se deu pela modelagem matemática do problema e na busca automática de soluções por meio de heurísticas.

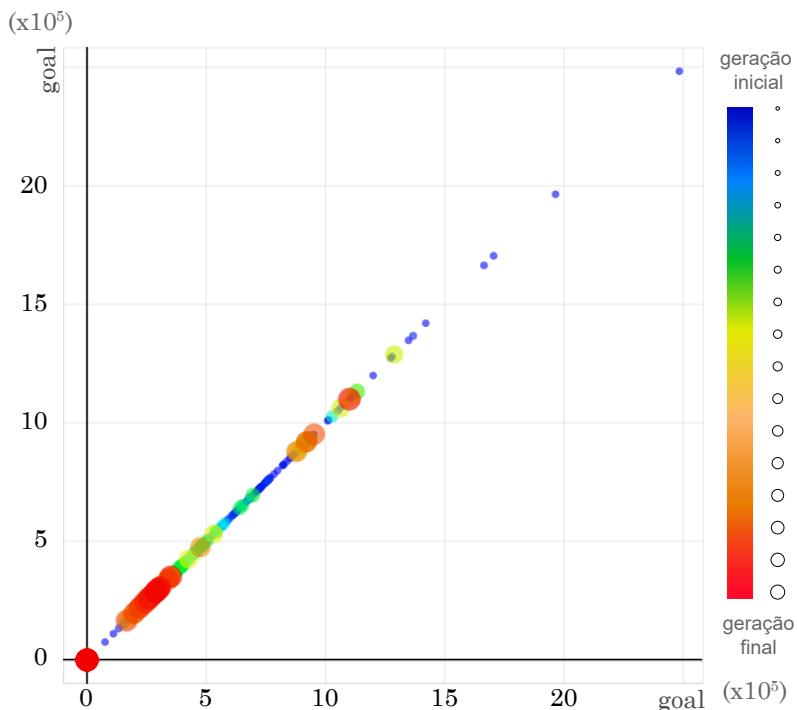
Nas próximas seções, serão apresentados análises detalhadas dos experimentos de otimização através da análise quantitativa dos dados obtidos, através de gráficos de dispersão e de estratégias para a seleção dos melhores indivíduos (opções de projeto) em cada caso.

4.3.1 Experimento 01: Otimização Mono-objetivo (função composta)

Na primeira etapa foi gerado um conjunto de 5.900 opções. A tarefa neste estágio foi filtrar os conjuntos de dados pelas pontuações e selecionar a alternativa com maior desempenho.

Na Figura 40 apresenta-se o gráfico de dispersão do espaço de projeto pesquisado pelo Discover[©]. Cada círculo representa uma opção.

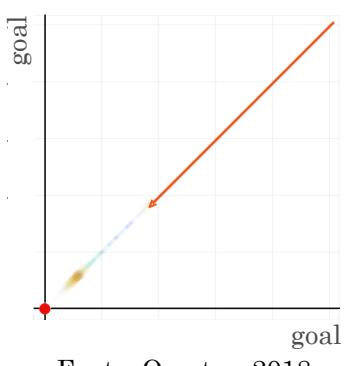
Figura 40: Gráfico de dispersão mono-objetivo



Fonte: O autor, 2018.

Nota-se que por ser uma otimização mono-objetiva, o gráfico apresenta todas as opções alinhadas, já que a abscissa e a ordenada assumem o mesmo valor (*goal*). Isso torna sua interpretação bastante intuitiva, pois quanto menor o peso do projeto, mais próximo do canto inferior esquerdo ele estará (Figura 41).

Figura 41: Percurso do projeto

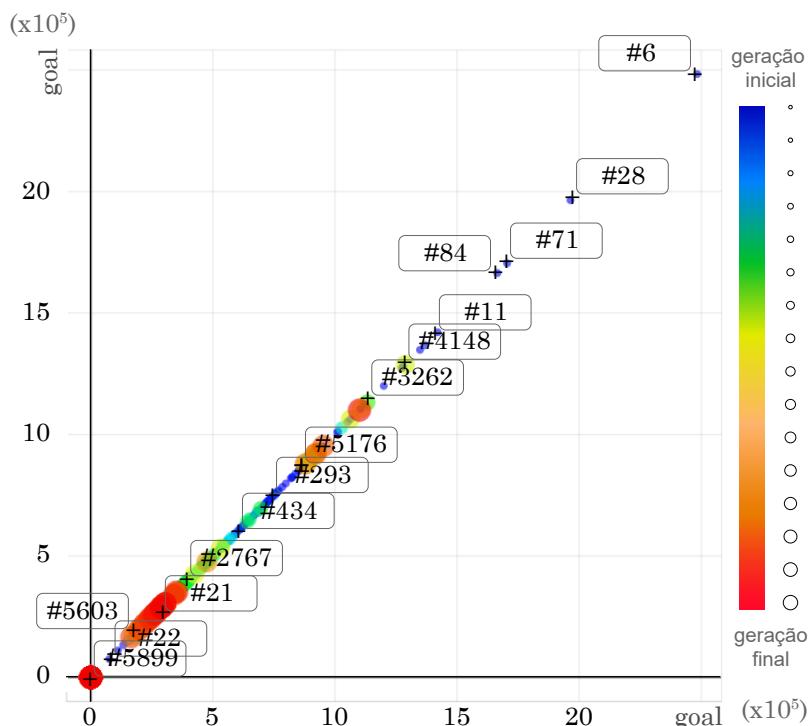


Fonte: O autor, 2018.

Neste gráfico, também é possível ilustrar o processo evolutivo descrito anteriormente. Nota-se que com o decorrer dos cruzamentos, os projetos foram progressivamente empurrados para o canto inferior esquerdo do gráfico (ponto ótimo).

Para comparação, consideremos as opções seguintes (Figura 42).

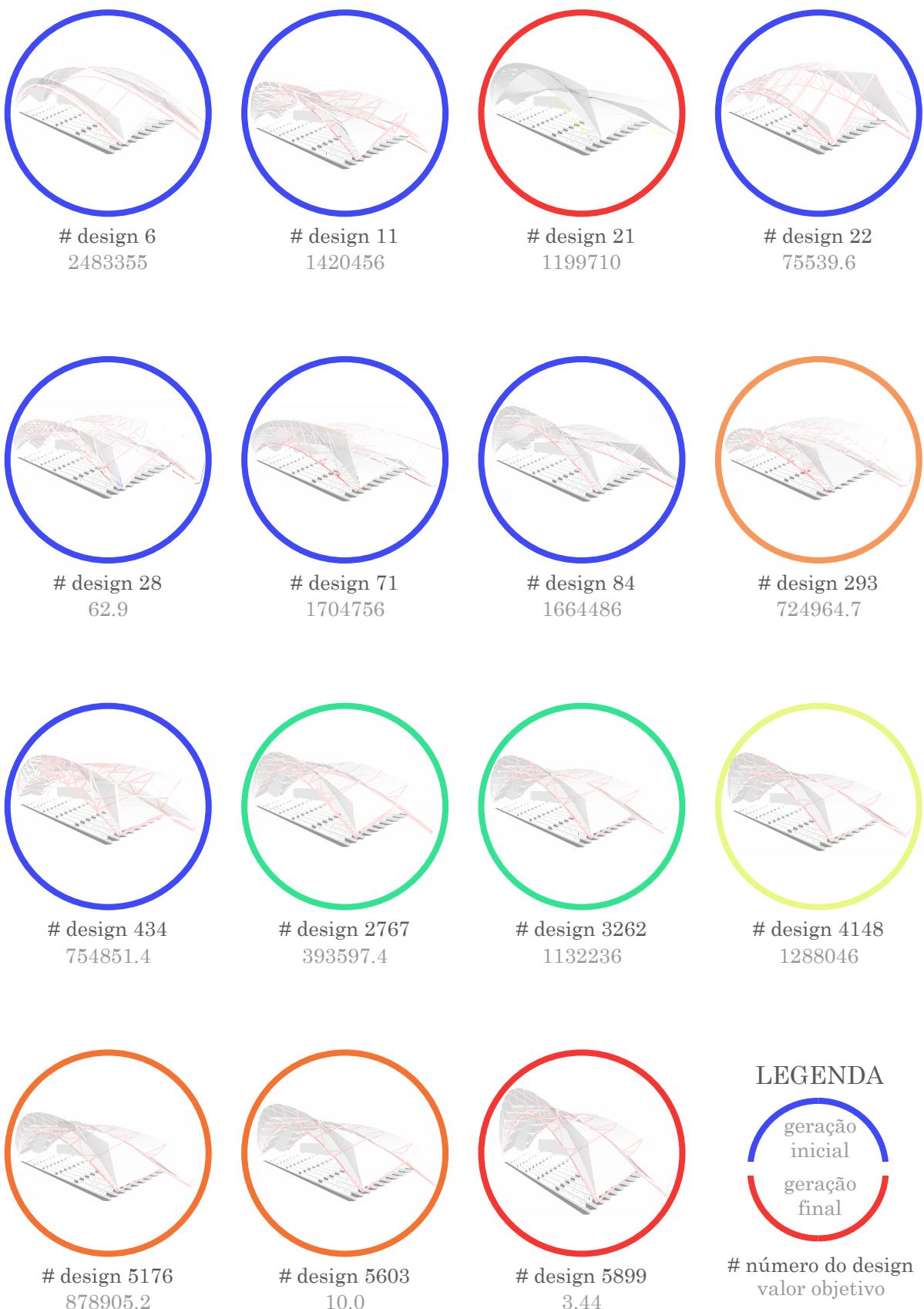
Figura 42: Opções escolhidas no gráfico de dispersão



Fonte: O autor, 2018.

Criado ainda na primeira geração, a opção #6 é a pior. Isso porque, nesta fase, as alternativas são completamente aleatórias, assim, os indivíduos podem conter poucos componentes certos para alcançar o objetivo. Ainda por isso, esta opção foi reprovada em mais de um critério na análise estrutural, empurrando-a para a extremidade oposta do ponto ótimo. Na Figura 43 é possível compará-las de forma mais clara.

Figura 43: Progresso do projeto no gráfico de dispersão mono-objetivo



Fonte: O autor, 2018.

Nota-se, que mesma na fase aleatória, as opções #11, #21, #22 e # 28 apresentam melhorias progressivas. Destes, a opção #28, chegou bem perto do ponto ótimo, próximo até mesmo do indivíduo #5603, das ultimas gerações, demonstrando o maior potencial para cruzamento futuro, quando comparada aos demais citados.

As alternativas #71, #84, #293, #434, #267, #3262, #4146, #5176 mostram, por sua vez, que apesar do avanço do processo, o cruzamento intenso ainda gera indivíduos ruins, que possivelmente foram descartados para os respectivos futuros cruzamentos.

Neste caso, a opção #5899 é considerada como ótima. Isso porque não tem nenhuma outra que tenha o objetivo (*goal*) menor do que 3.44.

Acessando o ID desta opção no arquivo do tipo *.tsv*, disponibilizado pelo pacote Discover[©], tem-se acesso a todos os parâmetros de entrada (*inputs*) e saídas (*outputs*) deste modelo, como por exemplo, as seções dos perfis (Tabela 1), o valor de seu objetivo, entre outros.

Tabela 1: Perfis da opção de projeto #5899

Barras¹	A	B	C	D
d²	168.3	33.4	33.4	114.3
t³	7.1	3.2	3.2	8.8

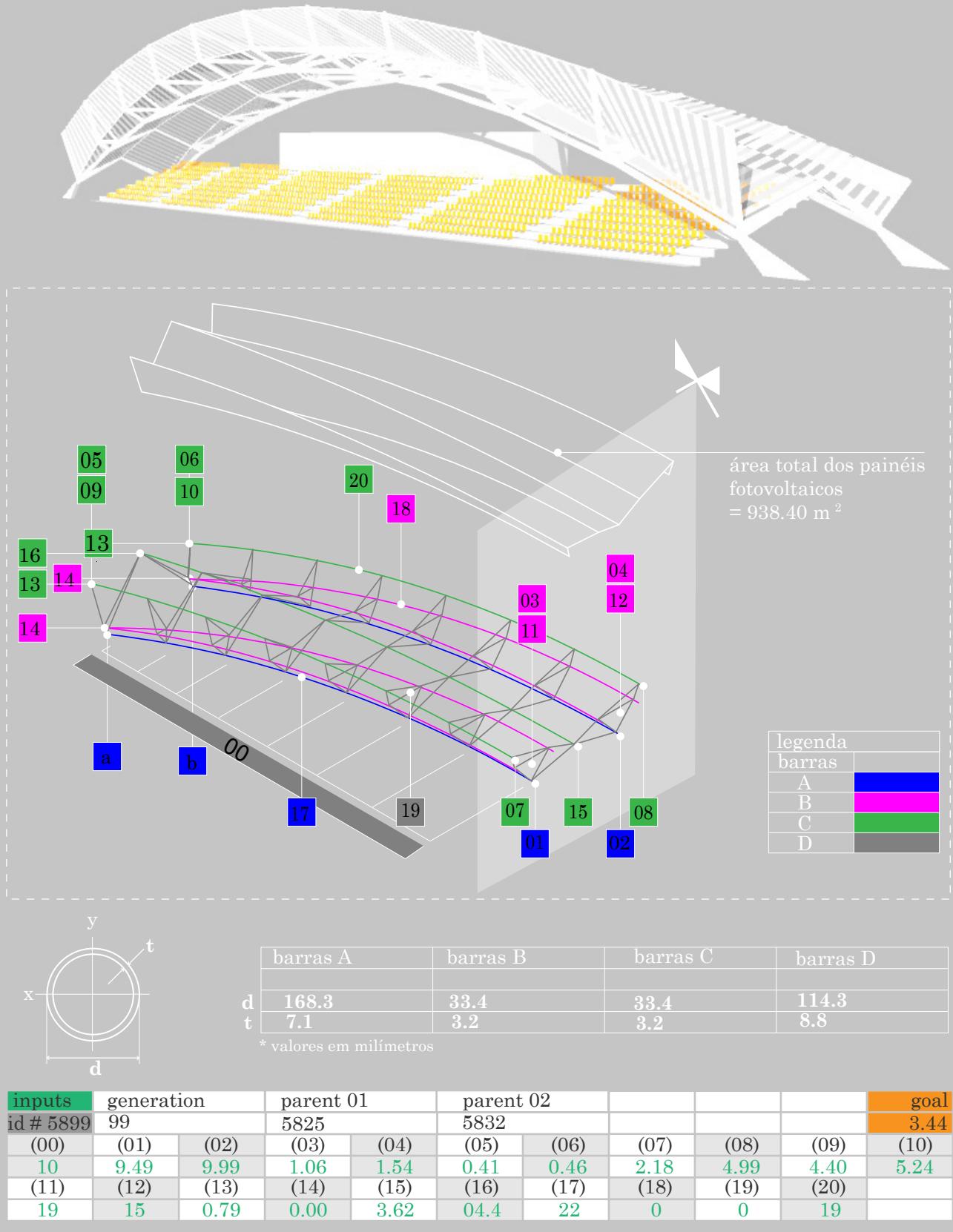
¹ fonte: Vallourec (2018)

² valor de *d* em milímetro

³ valor de *t* em milímetro

Na Figura 44 pode-se ver esta alternativa de maneira ampliada, assim como todos os seus parâmetros de entrada e suas saídas.

Figura 44: Opção número #5899 e seus parâmetros de entradas

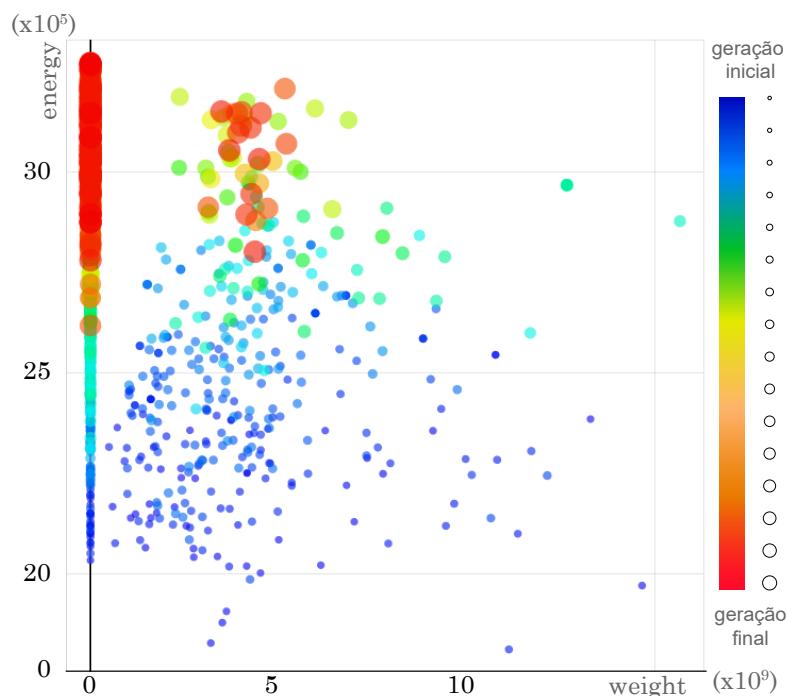


Fonte: O autor, 2018

4.3.2 Experimento 02: Otimização Multiobjetivo

Na Figura 45 mostra-se o gráfico de dispersão da segunda etapa do experimento com 5.900 alternativas.

Figura 45: Gráfico de dispersão multiobjetivo



Fonte: O autor, 2018.

O objetivo desta etapa foi encontrar opções que atendam à restrição estrutural com um peso mínimo (objetivo *weight*) e o maior potencial energético possível (objetivo *energy*). Como esses dois objetivos estão em competição um com o outro, não existe uma solução única, como apresentado no experimento anterior, mas sim um conjunto com várias soluções consideradas satisfatórias.

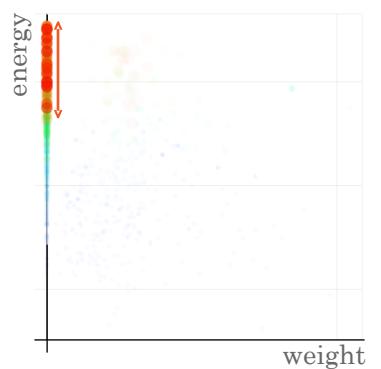
Isolando as alternativas de alto desempenho no espaço de projeto, notamos que os desenhos formam uma linha movendo-se do canto inferior

esquerdo para cima. Isso porque a estratégia adotada como restrição os deslocou para esta posição (Figura 46).

Neste caso, em que a negociação entre ter mais superfície de cobertura contraria o critério do menor peso, as opções consideradas ótimas se concentram em uma linha perpendicular à abscissa (objetivo minimizado) e paralela à ordenada (objetivo maximizado).

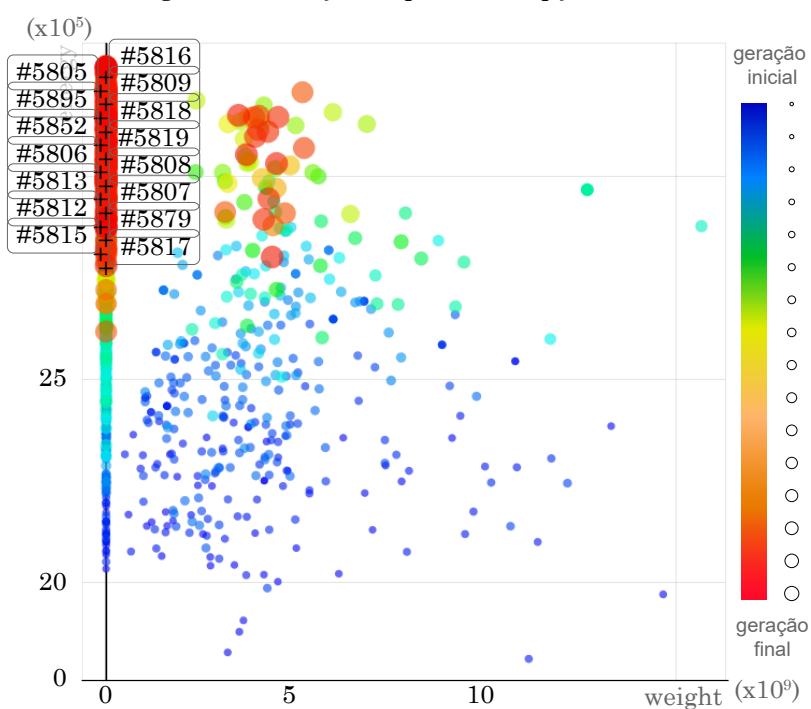
As opções ao longo desta linha podem ser consideradas todas ótimas. Isso porque melhorar qualquer um dessas em um objetivo é torna-lo, necessariamente, pior em outro. Dessa forma, uma análise profunda de várias alternativas se faz necessária. Para demonstração, selecionamos algumas dessas opções (Figuras 47 e 48) para uma análise mais detalhada.

Figura 46: Percurso dos projetos



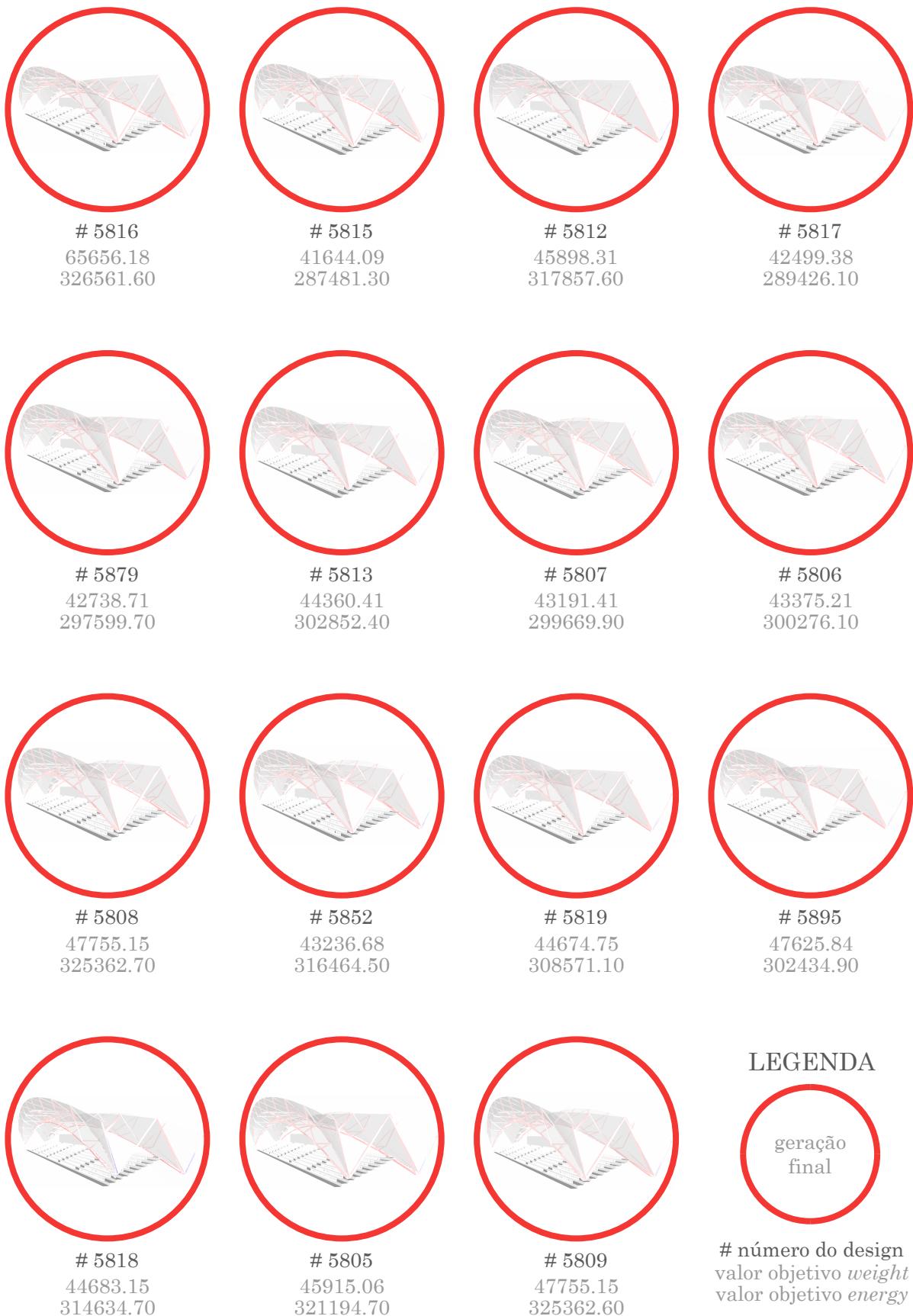
Fonte: O autor, 2018.

Figura 47: Seleção de parte das opções ótimas



Fonte: O autor, 2018.

Figura 48: Opções ótimas selecionadas do segundo experimento



Fonte: O autor, 2018.

Uma vez que esse conjunto de opções é selecionado, a tarefa passa a ser discutir sobre elas. Aqui, ressalta-se mais uma vez o projeto como processo, entidade aberta, onde a participação dos diversos profissionais de projetos e demais envolvidos, direta e indiretamente, com o espaço a ser construído é fundamental para a tomada das decisões aproveitamento de uma correta exploração das características de cada uma das alternativas consideradas ótimas.

Como exemplo desta tarefa, imaginemos uma relação direta do potencial energético como iniciativa para pagar o custo total do aço, em peso, utilizado na construção da cobertura (Tabela 2).

Tabela 2: Tabela comparativa

ID	Custo (R\$)¹	kWh (R\$)²	Meses³	30 anos (R\$)⁴
5816	1.313.123,60	191.691,60	82	4.437.626,18
5815	832.881,80	168.751,52	59	4.229.663,89
5812	917.966,20	186.582,41	59	4.679.506,14
5817	849.987,20	169.891,02	60	4.246.806,02
5879	854.774,20	174.691,02	59	4.385.956,52
5813	887.208,20	177.774,36	60	4.446.022,56
5807	863.828,20	175.906,23	59	4.413.358,74
5806	867.504,20	176.262,07	59	4.420.357,92
5808	955.103,00	190.987,90	60	4.774.534,15
5852	864.733,60	185.764,66	56	4.708.206,25
5819	893.495,00	181.131,24	59	4.540.442,07
5895	952.516,80	177.529,29	64	4.373.361,79
5818	893.663,00	184.690,57	58	4.647.054,07
5805	918.301,20	188.541,29	58	4.737.937,47
5809	955.103,00	190.987,85	60	4.774.532,39

¹ custo total da estrutura = (peso estrutura total) x (valor preço aço – R\$ 20,00/kg)

² receita anual energia elétrica = (potencial anual de energia fotovoltaica) x (média anual valor kWh – R\$ 0,587)

³ tempo para pagar a estrutura com as receitas da energia fotovoltaicas = (custo total da estrutura) / (receita / 12)

⁴ rendimento bruto em 30 anos (excluindo-se o valor da estrutura).

Temos que a opção #5852 é, neste caso, a melhor escolha. Acessando o ID desta opção no arquivo do tipo *.tsv*, tem-se acesso a seus parâmetros de entrada e saídas. Na Tabela 3 vemos as seções de perfis desta opção.

Tabela 3: Perfis da opção de projeto #5852

Barras¹	A	B	C	D
d²	219.1	42.2	33.4	168.3
t³	8	5	3.2	7.1

¹ fonte: Vallourec (2018)

² valor de *d* em milímetro

³ valor de *t* em milímetro

Na Figura 49 pode-se ver esta alternativa ampliada, seus respectivos parâmetros de entrada, assim como suas saídas.

Entretanto, essa lógica muda quando se tem um planejamento a longo prazo, como por exemplo, a redução de gastos com o consumo energético ao longo de 30 anos.

Nesta nova perspectiva, temos a opção #5808 como melhor escolha. Acessando o ID desta opção no arquivo do tipo *.tsv* vemos as seções de perfis desta opção (Tabela 4).

Tabela 4: Perfis da opção de projeto #5808

Barras¹	A	B	C	D
d²	219.1	43.8	38.1	168.3
t³	8	5	4	7.1

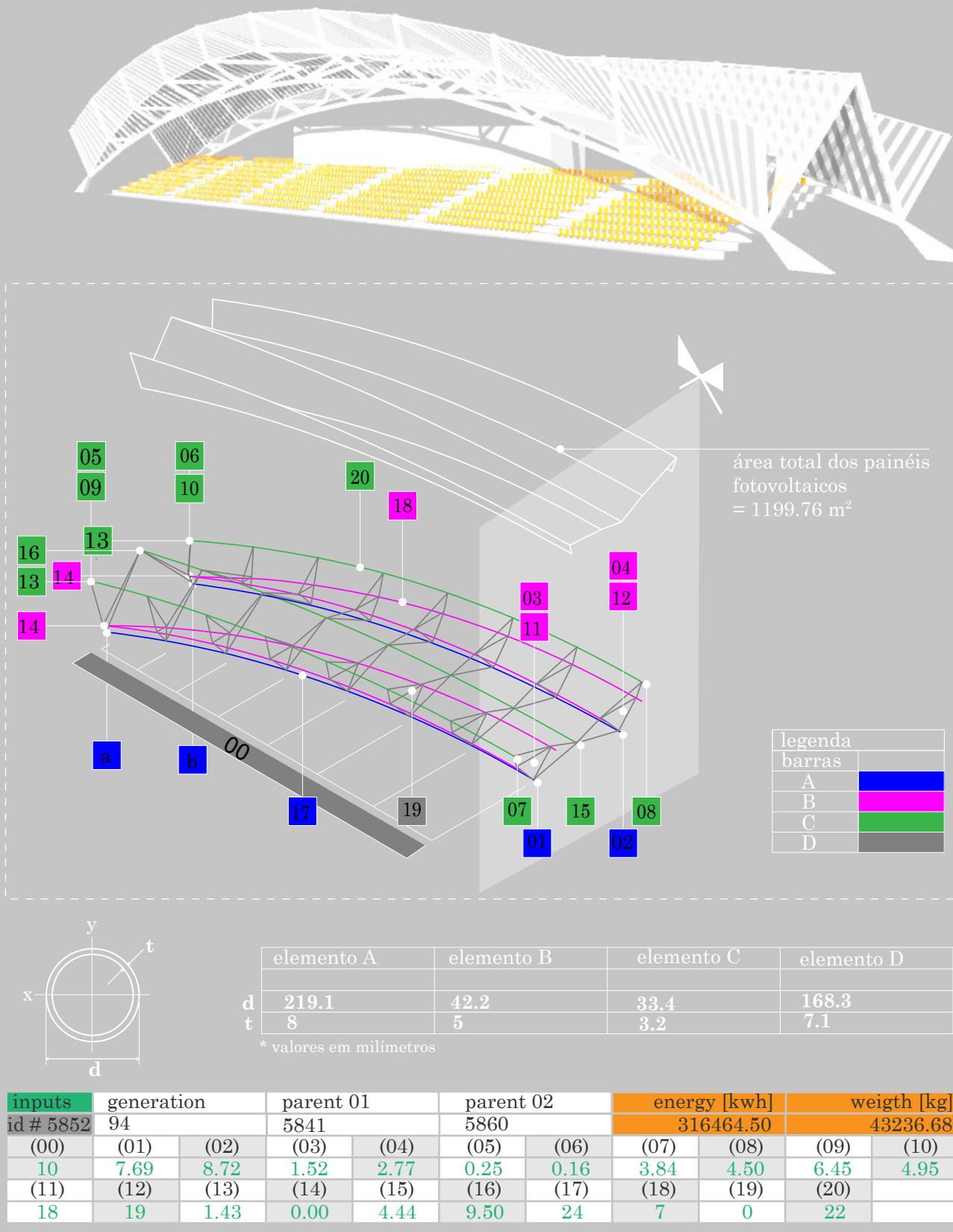
¹ fonte: Vallourec (2018)

² valor de *d* em milímetro

³ valor de *t* em milímetro

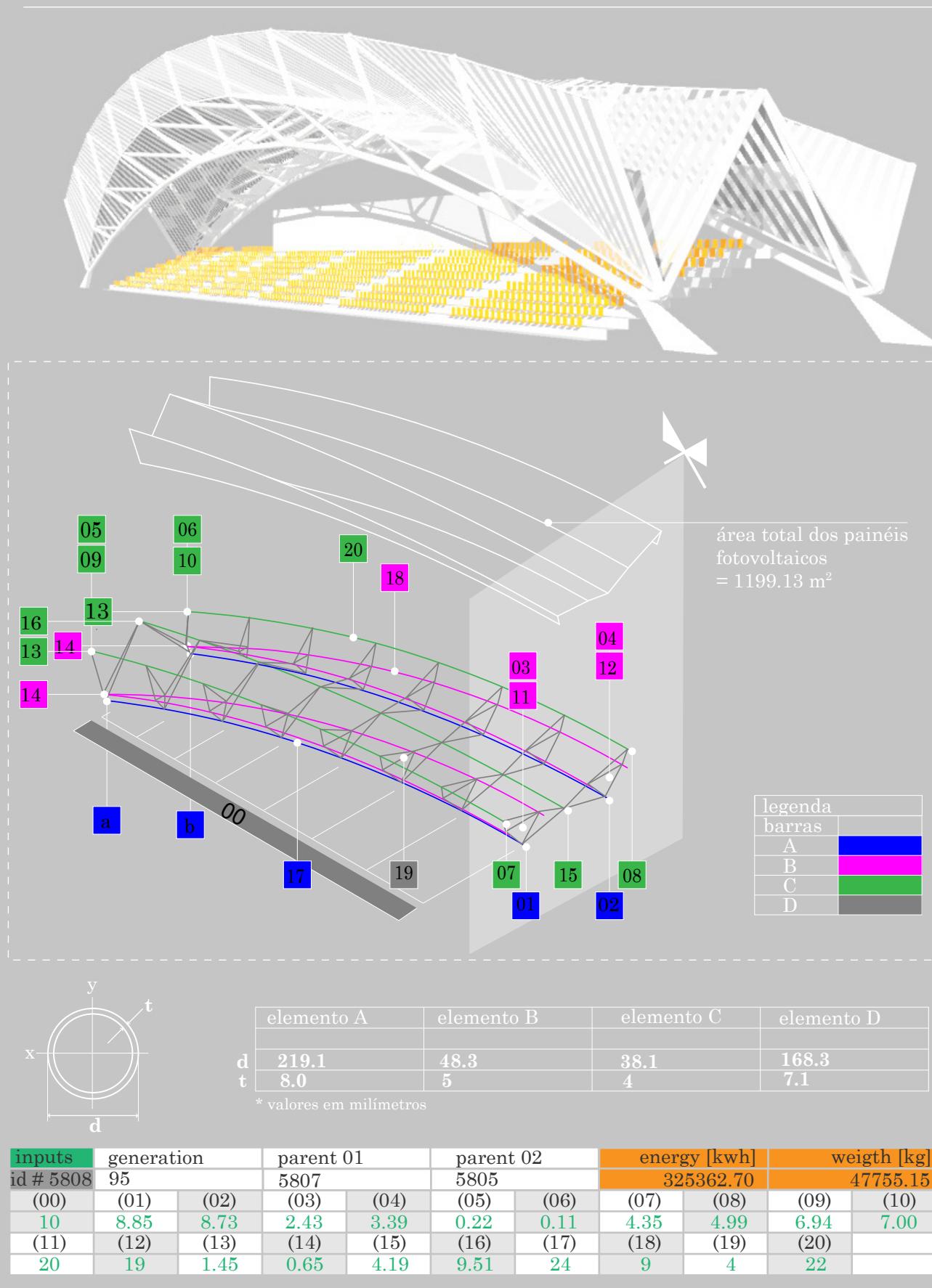
Na Figura 50 é ilustrada esta alternativa assim como seus respectivos parâmetros de entrada e saída.

Figura 49: Alternativa número #5852 e seus parâmetros de entradas



Fonte: O autor, 2018

Figura 50: Alternativa número #5808 e seus parâmetros de entradas



Fonte: O autor, 2018

5 CONSIDERAÇÕES E SUGESTÕES

5.1 Considerações finais

Neste trabalho destaca-se o projeto como prática em que arquiteto e projeto dialogam em um fluxo de informações que precedem o projeto em si, orientado por objetivos comuns. Para isso, buscou-se avaliar esse processo de projeto, discutindo, criticamente a perda de controle como estratégia.

Por meio dos experimentos apresentados destaca-se a relação entre otimização e o projeto aberto, onde o projetista perde controle da definição específica da solução, pois passa a operar na formalização do problema. Essa perda de controle se dá justamente pela modelagem matemática do problema que se constitui como uma prática analítica, e na busca automática de soluções por meio de heurísticas.

Como esses procedimentos estão sendo utilizados há pouco tempo, este trabalho espera contribuir como suporte para arquitetos que pretendam seguir por este caminho de projeto, apresentando-lhes uma descrição detalhada sobre os procedimentos adotados na aplicação deste conteúdo em projeto arquitetônico de construção metálica: (a) na descrição de um modelo de espaço de projeto arquitetônico em aço; (b) na definição de um conjunto de diretrizes para avaliar as qualidades de um espaço de projeto baseado na análise estrutural e na capacidade deste de gerar energia fotovoltaica e (c) na abordagem de método para avaliar os processos generativos, com um ou

mais de um objetivo, por meio de amostragem e visualização em gráfico de Pareto.

Destaca-se o potencial da utilização dos algoritmos genéticos como ferramenta auxiliar integrada ao *design* generativo como um método de projeto permitiu uma aplicação customizada de procedimentos científicos (modelagem) para a síntese da forma (*design* final). Assim, o projeto se desenvolveu ancorado numa visão mais criativa do uso de procedimentos da ciência.

Outro ponto a ser destacado é a comparação de dois cenários de otimização de projeto para um exemplo concreto. A formulação do problema e dos dois cenários de otimização são pertinentes e acredita-se que a possível contribuição do trabalho seria justamente mostrar como as diferentes formulações para o mesmo problema se comportam em um processo de projeto.

Por fim, destaca-se também um caminho transdisciplinar no sentido da capacidade de compreender o processo e de diálogo entre as diferentes áreas do saber. Este caminho foi estabelecido por meio da crítica da fragmentação do conhecimento. Basarab Nicolescu, em seu livro intitulado *O Manifesto da Transdisciplinaridade* (1999), sugere reconhecer a existência de complexas pluralidades no mundo. Assim, o autor propõe abordagens alicerçadas na compreensão das múltiplas dimensões da realidade.

Para Nicolescu, a transdisciplinaridade é um dos caminhos para compreender a complexidade do

universo e do mundo. Os procedimentos metodológicos específicos de projeto abordados nesta dissertação nos obrigou encarar o projeto com maior complexidade projetual. Esta complexidade se deu no dialogo entre as diversas disciplinas de arquitetura e de engenharia, além de outras, como da matemática e da ciência da computação, num esforço capaz de pensar um processo que integrasse as várias disciplinas e campos de conhecimento, rompendo as fronteiras de cada uma delas.

Frente aos múltiplos desafios enfrentados neste trabalho, foi necessário estimular um olhar criativo capaz de integrar estas disciplinas de maneira inteligente, e não apenas sobrepondo-as. A ideia de se dar a mesma importância a cada uma das disciplinas, unindo-as em um sistema que uma vez construído não se pode desfazer, é outra referência a transdisciplinaridade presente neste trabalho. Para Erich Jantsch (1970), a transdisciplinaridade é também o reconhecimento dessa interdependência de todos os aspectos de uma realidade, que neste caso, foi um processo de projeto.

A aposta nesta abordagem projetual permitiu a transgressão das fronteiras imposta pelas disciplinas acadêmicas (NOWOTNY, 2006) e também foi à causa da potencialização das ferramentas digitais e da redefinição destas como metodologias específicas. Este olhar transdisciplinar nos permitiu um dialogo constante entre a parte e o todo, onde o *design* despontou num cenário onde o saber se propôs a explorar e integrar

distintas áreas do conhecimento numa prática de projeto.

5.2 Sugestões para trabalhos futuros

Fica como principais sugestões para trabalhos futuros as seguintes sugestões:

- (a) – o desdobramento metodológico desse trabalho através de oficinas práticas como reflexão sobre o método adotado, nas quais este pudesse ser testado e discutido com estudantes e profissionais, de modo a aferir como ele seria recebido pelos projetistas.
- (b) - um maior aprofundamento nas questões sobre quem define o que é problema, ou ainda, o que entra nessa delimitação. Esta questão é importante porque quando um problema é mal formulado, o projeto não resolve a situação-problema;
- (c) – uma maior abertura do processo projetual. Para além dos outros agentes envolvidos na projetação (parceiros projetistas), o projeto como pergunta também abre espaço para incorporar ao processo os futuros usuários. Um dos caminhos seria dotar este sistema, cada vez mais, de repertórios que o direcione para o *metadesign*;
- (d) – por fim, abordar, discutir e testar este método adotando critérios e

parâmetros não-monetários para a escolha do projeto ótimo. Isto seria um desafio interessante, pois este trabalho adotou apenas critérios (peso da estrutura e potencial de geração de energia) amoedáveis como objetivos dos procedimentos de otimização testados.

REFERÊNCIAS

ASHBY, W. R. **Uma introdução à cibernetica.**

São Paulo: Editora Perspectiva, 1970.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6123: Forças devidas ao vento em edificações.** Rio de Janeiro. 1988.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 8681: Ações e segurança nas estruturas - Procedimento.** Rio de Janeiro. 2003.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 8800: Projeto de estruturas de aço e de estruturas mistas de aço e concreto de edifícios.** Rio de Janeiro. 2008.

BUCHANAN, R. Wicked Problems in Design Thinking. **MIT Press Journal**, 1992. p. 5-21.

DARWIN, C. **Origem das espécies.** Belo Horizonte: Villa Rica, v. v.7, 1994.

DAVIS, D. Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. Melbourne: School of Architecture and Design College of Design and Social context RMIT University, 2013. p. 243.

DELEUZE, G.; GUATTARI, F. **Mil platôs: capitalismo e esquizofrenia.** Vol. 5. São Paulo: Editora 34, 1995.

FERRANTE, I. P. G. **Entre a arquitetura experimental e a arquitetura para a experiência:** máquinas, corpos e próteses de Lars Spuybroek/NOX. São Paulo: Faculdade de

Arquitetura e Urbanismo da Universidade de São Paulo, 2013.

GIACCARDI, E.; FISCHER, G. Creativity and evolution: A metadesign perspective. **Digital Creativity**, p. p. 19 - 32, March 2008.

GLANVILLE, R. On being out of control. **Lemende Systeme**, v. 11, January/February 2002.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. 2nd Edition, 1992. ed. Ann Arbor: MIT Press, 1975.

JANTSCH, E. Inter- and Transdisciplinary University: A systems approach to Education and Innovation. **Higher Education Quarterly**, p. p. 403-428, February 1970.

JUSTINA, L. A. D; CALUZI, J. J.; MEGLHIORATTI, F. A.; CALDEIRA, A. M. A. A herança genotípica proposta por Wilhelm Ludwig Johannsen. **Filosofia e História da Biologia**, São Paulo, n. 1 v.5 2010. p. 55-71.

LAWSON, B. **How designers think**. London: Butterworth-Heinemann, 1983.

MENGES, A. Pluripotent Components and Polymorphous Systems: an alternative approach to parametric desing. **AA Files**, 2005. 63-74.

NAGY, D. Generative Design. **Medium**, 23 jan 2018. Disponivel em: <<https://medium.com/generative-design>>.

NICOLESCU, B. **O manifesto da transdisciplinaridade**. São Paulo: Triom, 1999.

NOWOTNY, H. The potencial of transdisciplinary, Wien, 2006.

OXMAN, R.; OXMAN, R. M. Refinement and adaptation in design cognition. **Design Studies**, p. p. 117-134, 1992.

PREISINGER, C. Karamba - User Manual for Version 1.2.2. [S.l.]: [s.n.], 2016.

RIBEIRO, C.; PRATSCHKE, A. **Transdisciplinaridade e complexidade na arquitetura.** Universidade Federal de Espírito Santo. ed. Vitória: II Congresso Mundial de Transdisciplinaridade, v. v. CD-Rom, 2005.

ROCHA, B. M. **Novos olhares e desafios da complexidade na epistemologia projetual.** n. 39. ed. São Paulo: Pós, v. v. 23, 2016.

RUTTEN, D. Galapagos: on the logic and limitations of generic solvers. **Architectural Design**, n. 83 (2), p. p. 132-135, 2013. ISSN doi: 1002/ad. 156.

SCHUMACHER, P. The Autopoiesis of Architecture: a new agenda for Architecture, Chichester: Wiley, 2012.

SHIFFMAN, D. **The Nature of Code.** California: Magic Book Project, 2012.

SPASIC, D. Ladybug Photovoltaics components released. **grasshopper3d**, 2015. Disponível em: <<https://www.grasshopper3d.com/forum/topics/ladybug-photovoltaics-components-released?groupUrl=ladybug&groupId=2985220%3A>

Group%3A658987&id=2985220%3ATopic%3A13095
23&page=4#comments>. Acesso em: 25 jun. 2018.

VASSÃO, C. A. **Metadesign:** ferramentas, estratégicas e ética para a complexidade. 1^a. ed. São Paulo: Blücher, 2010.

YANG, X.-S. **Nature-Inspired Metaheuristic Algorithms.** Second Edition. ed. United Kingdom: Ed. Luniver Press, 2010.

YENIAY, Ö. Penalty Function Methods For Constrained Optimization With Genetic Algorithms. **Mathematical and Computational Applications,** Beytepe, v. Vol. 10, p. pp. 45-56, 2005. ISSN No. 1.

GLOSSÁRIO

A

ALGORITMO

Algoritmo é a sequência lógica e finita de instruções que tem como finalidade executar uma tarefa.

ALGORITMO GENÉTICO

Algoritmo genético é a sequência lógica e finita de instruções que, inspiradas na biologia evolutiva como hereditariedade, mutação, seleção natural e recombinação, tem como finalidade executar uma tarefa.

ALGORITMO META-HEURÍSTICO

Algoritmo meta-heurístico é um método heurístico que tem como finalidade resolver problemas de otimização.

C

CARÁTER PUNITIVO

Qualidades que distingue um comportamento que castiga ou tem relação com o castigo.

CÉLULA FOTOVOLTAICA

Célula fotovoltaica é o dispositivo o capaz de converter a energia solar em energia elétrica por intermédio do efeito fotovoltaico.

CÉLULA FOTOVOLTAICA DE PELÍCULA FINA

Uma das tecnologias da terceira geração de células fotovoltaicas.

COMPUTAÇÃO EVOLUCIONÁRIA

Computação evolucionária constitui uma família de algoritmos do ramo da inteligência computacional e da computação natural inspiradas na biologia evolutiva.

CONDIÇÃO DE RESTRIÇÃO

Característica que determina restrição, que impõe limites.

D

DESEMPENHO

Desempenho é o conjunto de características ou capacidades de comportamento e rendimento de um indivíduo, objeto ou entidade.

DESEMPEÑO ESTRUTURAL

Desempenho da organização, disposição e ordem dos elementos essenciais que compõem o corpo estruturante.

DESIGN GENERATIVO	Design gerativo é a abordagem projetual inspirada na biologia evolutiva.
F	
FENÓTIPO	Fenótipo é o conjunto das características observáveis de um organismo ou população.
FLUXO DE INFORMAÇÃO	Fluxo de informação é qualquer rastreamento de informações, como por exemplo, as informações em um algoritmo.
FORMA	Nesta dissertação, a palavra forma foi empregada com o mesmo significado de forma plástica.
FUNÇÃO COMPOSTA	Problema de otimização representado por dois ou mais objetivos articulados como função objetivo comum.
FUNÇÃO MONO- OBJETIVO	Problema de otimização representado por uma única função objetivo.
FUNÇÃO MULTIOBJETIVO	Problema de otimização representado por duas ou mais funções objetivos.
G	
GENÓTIPO	Genótipo é a constituição genética de uma célula, organismo ou indivíduo.
GRÁFICO DE DISPERSÃO	Gráfico de dispersão é a representação de dados de duas ou mais variáveis organizados em um gráfico de coordenadas cartesianas.
H	
HEURÍSTICA	Heurística é o processo cognitivo empregado em decisões não racionais.
M	
MAXIMIZAÇÃO	Ato ou efeito de maximizar, de elevar ao máximo.
METADESIGN	<i>Metadesign</i> é a modalidade do <i>design</i> que lida com a construção do contexto agregando a coparticipação como questões indispensáveis.
MÉTODO DETERMINÍSTICO	Método determinístico é totalmente determinado pelos valores dos parâmetros e pelas condições iniciais.

MÉTODO ESTOCÁSTICO	Método estocástico possui alguma aleatoriedade inerente, assim, o mesmo conjunto de valores de parâmetros e condições iniciais levará a um conjunto de diferentes saídas.
MINIMIZAÇÃO	Ato ou efeito de minimizar, de elevar ao mínimo.
MODELAGEM MATEMÁTICA	Modelagem matemática consiste na descrição matemática de fenômenos reais a fim de prever seus comportamentos.
MODELO FLUIDO	Modelo fluido é o resultante da substituição do <i>design</i> do objeto pelo <i>design</i> do processo que gera o objeto.
MODELO MUTÁVEL	Empregado no mesmo sentido de modelo fluido.
MORFOGÊNESE	Morfogênese é um processo biológico de modelagem dos organismos.
O	
OTIMIZAÇÃO	Otimização se refere ao estudo matemático de problemas em que se busca minimizar ou maximizar uma função.
OTIMIZAÇÃO ESTRUTURAL	Ato ou efeito que busca minimizar ou maximizar uma estrutura.
P	
PARÂMETROS DE ENTRADA	Parâmetros de entradas são a organização e a estandardização da informação que se insere em um sistema.
PARÂMETROS DE SAÍDA	Parâmetros de saída são a organização e a estandardização da informação que é recebida de um sistema.
PARÂMETROS NÃO-MONETÁRIO	Parâmetros não-monetário são dados amoedáveis em que são considerados para analisar ou valorizar uma situação.
POTENCIAL ENERGÉTICO	Capacidade de produção de energia.
S	
SIMULAÇÃO	Ensaio que consiste na reprodução, artificialmente,

ENERGÉTICA	das condições reais de um fenômeno energético.
V	
VARIÁVEIS	Atributos, mensuráveis ou não, no interior de um sistema.
VIABILIDADE DE PROJETO	Atributo ou condição do projeto que é viável.
<i>VISUAL PROGRAMMING LANGUAGE</i>	Linguagem de programação que permite a criação de programas a partir da manipulação de elementos gráficos.

APÊNDICES

APÊNDICE A - ALGORITMO PARA A COMBINAÇÃO DE AÇÕES

Algoritmo para determinar a combinação mais exigida. Língua: *Python*.
Elaborado pelo autor.

```
[01]fatorM = 0.0098 # fator de multiplicação
[02]CP = celulasPV + membrane
[03]CA = sobrecarga
[04]c1 = (1.4 * CP) + (1.4 * CA)
[05]c11 = c1 * fatorM
[07]print ("combinação 1 = %.2f + PP" %c11)
[08]c2 = (1.4 * CP) + (1.4 * CA) + (1.4 * 0.6 * CV)
[10]c21 = c2 * fatorM
[11]print ("combinação 2 = %.2f + PP" %c21)
[13]c3 = (1.4 * CP) + (1.4 * CV) + (0.70 * CA)
[15]c31 = c3 * fatorM
[16]print ("combinação 3 = %.2f + PP" %c31)
[18]c4 = (1.4 * (CP * fatorM)) + (1.4 * CV)
[19]print ("combinação 4 = %.2f + PP" %c4)
[21]#OUTPUTS
[23]Permanentes = - (CP * 1.4 * fatorM)
[24]Acidentais = - (CA * 1.4 * fatorM)
[25]Vento = - (1.4 * 0.6 * CV * fatorM)
```

APÊNDICE B - ALGORITMO PARA DETERMINAR A FORÇA DE ARRASTO

Algoritmo para determinar F_a . Língua: *Python*. Elaborado pelo autor.

```
[01]s2 = b * (z / 10) ** p
[02]#print ("S2 = %.2f" %s2)

[04]vk = v0 * s1 * s2 * s3
[05]print ("velocidade característica (Vk) = %.2f m/s" %vk)

[07]#pressão dinâmica do vento
[08]q = (0.613 * (vk) ** 2)/1000           #divisão por 1000 --> N para KN
[09]print ("pressão dinâmica do vento (q) = %.2f KN/m2" %q)

[11]# componente global da força Fa na direção do vento
[12]Ca = 1.3
[13]Fa = Ca * q * Ae
[14]print ("força de arrasto (Fa) = %.2f KN" %Fa)

[16]a = Fa
```

APÊNDICE C - ALGORITMO PARA A ESCOLHA DOS PERFIS

Algoritmo para a escolha dos perfis. Língua: *Python*. Elaborado pelo autor.

1/1 - Selection of circular tubular section

```
[01]import rhinoscriptsyntax as rs
[02]
[03]list = []
[04]
[05]f = open(f,'r')
[06]for line in f:
[07]    list_line = line.rstrip().split('\t')
[08]    list.append(list_line)
[09]print list
[10]
[11]W = []
[12]I = []
[13]Z = []
[14]
[15]proA = lista[indexA]
[16]pA = proA[0].split(',')
[17]diameterA = pA[0]
[18]DiameterA = float(diameterA) / 10
[19]thicknessA = pA[1]
[20]ThicknessA = float(thicknessA) / 10
[21]profileA = diameterA,thicknessA
[22]W.append(pA[6])
[23]I.append(pA[5])
[24]Z.append(pA[7])
[25]
[26]proB = lista[indexB]
[27]pB = proB[0].split(',')
[28]diameterB = pB[0]
[29]DiameterB = float(diameterB) / 10
[30]thicknessB = pB[1]
[31]ThicknessB = float(thicknessB) / 10
[32]profileB = diameterB,thicknessB
[33]W.append(pB[6])
[34]I.append(pB[5])
[35]Z.append(pB[7])
[36]
[37]proC = lista[indexC]
[38]pC = proC[0].split(',')
[39]diameterC = pC[0]
[40]DiameterC = float(diameterC) / 10
[41]thicknessC = pC[1]
[42]ThicknessC = float(thicknessC) / 10
[43]profileC = diameterC,thicknessC
[44]W.append(pC[6])
[45]I.append(pC[5])
[46]Z.append(pC[7])
[47]
[48]proD = lista[indexD]
```

```
[49]pD = proD[0].split(',')
[50]diameterD = pD[0]
[51]DiameterD = float(diameterD) / 10
[52]thicknessD = pD[1]
[53]ThicknessD = float(thicknessD) / 10
[54]profileD = diameterD,thicknessD
[55]W.append (pD[6])
[56]I.append (pD[5])
[57]Z.append (pD[7])
```

APENDICE D - ALGORITMO PARA ANÁLISE ESTRUTURAL

Algoritmo para análise estrutural. Língua: *Python*. Elaborado pelo autor.

1/5 – Calculate $N_{t,rd}$ and $N_{c,rd}$

```
[01]import math
[02]
[03]listDiameter = (diameterCm)
[04]listThickness = (thicknessCm)
[05]num = (len(listThickness))
[06]
[07]cont = 0
[08]
[09]momentInertia = I
[10]γa1 = 1.10
[11]
[12]forceTraction = []
[13]forceCompression = []
[14]
[15]while cont < num:
[16]
[17]    D = listDiameter[cont] * 10
[18]    t = listThickness[cont] * 10
[19]
[20]    Ag = math.pi * (D * t - t ** 2)
[21]
[22]    #traction
[23]
[24]    Ntrd = (Ag * fy) / γa1
        forceTraction.append(Ntrd)
[25]    print ('Nt,rd %s: %.2f KN' % (cont,
                                         Ntrd))
[26]
[27]    #compression
[28]
[29]    if D/t <= E/fy:
[30]        Q = 1.00
[31]    else:
[32]        Q = (0.038 * E)
            / ((fy * (D / t)) + 2 / 3)
[33]
[34]    Ne = (((math.pi) ** 2) * (E *
        momentInertia[cont])) / (K) ** 2
[35]    cont += 1
[36]    λ = math.sqrt(Q * Ag * fy) / Ne
[37]    if λ <= 1.5:
[38]        χ = 0.658 ** (λ ** 2)
[39]    else:
[40]        χ = 0.877 / (λ ** 2)
[41]    Ncrs = (χ * Q * Ag * fy) / γa1
        forceCompression.append(Ncrs)
[42]    print ('Nc,rd %s: %.2f KN' % (cont,
                                         Ncrs))
[43]
```

[44] #print λ

2/5 – Search $N_{t,sd}$ and $N_{c,sd}$ maximum

```
[01]L = normalForce
[02]
[03]listMax = map(abs,L)
[04]listMax.sort()
[05]forceMax = (listMax[-1])
[06]
[07]traction = forceMax
[08]compression = forceMax
[09]
[10]print ('A --> Max Mt, sd: %0.2f KNm'%
           traction)
[11]print ('A --> Max Mc, sd: %0.2f KNm'%
           compression)
```

3/5 – Calculate M_{rd}

```
[01]import math
[02]
[03]listDiameter = (diameterCm)
[04]listThickness = (thicknessCm)
[05]num = (len(listThickness))
[06]
[07]ResistanceElastic = W
[08]gamma1 = 1.10
[09]cont = 0
[10]listMrd = []
[11]
[14]while cont < num:
[15]
[16]    D = listDiameter[cont] * 10
[17]    t = listThickness[cont] * 10
[18]
[19]    Ag = math.pi * (D * t - t ** 2)
[20]
[21]    lambda_ = D / t
[22]    lambda_p = 0.07 * (E / fy)
[23]    lambda_r = 0.31 * (E / fy)
[24]
[25]    if lambda_ <= lambda_p:
[26]        Mrd = ((Z[cont]) * fy) / gamma1
[27]        listMrd.append(Mrd)
[27]    elif lambda_p < lambda_ < lambda_r:
[28]        Mrd = (((1/gamma1) * ((0.021 * E) / (D /
[28]            t)) + fy) * ResistanceElastic[cont])
[29]        listMrd.append(Mrd)
[30]    else:
[31]        Mrd = (((1 * 0.33 * E) / (gamma1 * (D /
[31]            t))) * ResistanceElastic[cont])
[32]        listMrd.append(Mrd)
[33]    cont += 1
[34]
[35] print ('Mrd %s: %.2f KNm' % (cont, Mrd))
```

4/5 – Search M_{sd} maximum

```
[01]listBM = bendingMoment
```

```

[02]listBM.sort()
[03]maximumBendingMoment = listBM[-1]
[04]
[05]print ('A --> Max Msd: %0.2f KNm'%
           maximumBendingMoment)
5/5 - Structural analysis:  $N_{t,rd} \leq N_{t,sd}$  ,  $N_{c,rd} \leq N_{c,sd}$ 
 $M_{sd} \leq M_{rd}$  and Max. Displacement  $\leq \ell/300$ .

```

[01]*Ntsd = traction*
[02]*Ntrd = forceTraction*

```

[03]num = (len(Ntsd))
[04]cont = 0
[05]gTraction = []
[06]
[07]while cont < num:
[08]  if Ntrd[cont] > Ntsd[cont]:
[09]    print ('Traction %s is OK!' % cont)
[10]  else:
[11]    print ('Traction %s is NOT OK!' %
            cont)
[12]    gTraction.append(Ntsd[cont] * punishment)
[13]  cont += 1

[14]Ncsd = compression
[15]Ncrd = forceCompression
[16]
[17]num2 = (len(Ncsd))
[18]cont2 = 0
[19]gCompression = []
[20]while cont2 < num2:
[21]  if Ncrd[cont2] > Ncsd[cont2]:
[22]    print ('Compression %s is OK!' %
            cont2)
[23]  else:
[24]    print ('Compression %s is NOT OK!' %
            cont2)
[25]    gCompression.append(Ncsd[cont2] * punishment)
[26]  cont2 += 1
[27]

[28]Msd = flecha
[29]Mrd = forceFlecha
[30]
[31]num2 = (len(Msd))
[32]cont2 = 0
[33]gFlecha = []
[34]
[35]while cont2 < num2:
[36]  if Mrd[cont2] > Msd[cont2]:
[37]    print ('Bending Moment %s is OK!' %
            cont2)
[38]  else:
[39]    print ('Bending Moment %s is NOT
              OK!' % cont2)
[40]    gFlecha.append(Msd[cont2] * punishment)

```

```
[41] cont2 += 1
[42]
[43]l = support
[44]
[45]if ((l / 300) > maximumDisplacement):
[46]   print ('Flecha is OK!')
[47]   disp = 0
[48]else:
[49]   print ('Flecha is NOT OK!')
[50]   disp = (l + punishment + 100)
[51]
[52]addition = sum(gTraction) + [53]sum(gCompression) + sum(gFlecha)
[54]goal = (mass + addition + disp)
[55]
[56]print ('Total mass: %.2f kg' % mass)
[57]print ('Mass to optimize: %.2f kg' % goal)
```

ANEXOS

ANEXO A - ALGORITMO PARA OTIMIZAÇÃO MONO-OBJETIVO

Definição em *Python* do algoritmo⁷ genético para *design* generativo.
Elaborado por Danil Nagy. Configurado pelo autor.

Algoritmo para otimização mono-objetivo: função composta.

```
[01]from src import job
[02]
[03]jobDescription = {
[04]"jobName": "untitled",
[05]"inputsDef": [
[06] { "name": "input00", "type":
[07]     "continuous", "range": [10,20]},
[06] { "name": "input01", "type":
[07]     "continuous", "range": [5.0,10.0]},
[07] { "name": "input02", "type":
[08]     "continuous", "range": [5.0,10.0]},
[08] { "name": "input03", "type":
[09]     "continuous", "range": [1.0,4.0]},
[09] { "name": "input04", "type":
[10]     "continuous", "range": [1.0,4.0]},
[10] { "name": "input05", "type":
[11]     "continuous", "range": [0.0,1.0]},
[11] { "name": "input06", "type":
[12]     "continuous", "range": [0.0,1.0]},
[12] { "name": "input07", "type":
[13]     "continuous", "range": [2.0,5.0]},
[13] { "name": "input08", "type":
[14]     "continuous", "range": [2.0,5.0]},
[14] { "name": "input09", "type":
[15]     "continuous", "range": [2.0,7.0]},
[15] { "name": "input10", "type":
[16]     "continuous", "range": [2.0,7.0]},
[16] { "name": "input11", "type":
[17]     "continuous", "range": [10,20]},
[17] { "name": "input12", "type":
[18]     "continuous", "range": [10,20]},
[18] { "name": "input13", "type":
[19]     "continuous", "range": [0.0,1.5]},
[19] { "name": "input14", "type":
[20]     "continuous", "range": [0.0,1.5]},
[20] { "name": "input15", "type":
[21]     "continuous", "range": [1.0,5.0]},
[21] { "name": "input16", "type":
[22]     "continuous", "range": [4.0,10.0]},
[22] { "name": "input17", "type":
[23]     "continuous", "range": [0,30]},
[23] { "name": "input18", "type":
[24]     "continuous", "range": [0,30]},
[24] { "name": "input19", "type":
```

⁷ Disponível em <https://github.com/danilnagy/discover>. Acessado em 24 de abril de 2018

```
    "continuous", "range": [0,30]},  
[25] { "name": "input20", "type":  
      "continuous", "range": [0,30]},  
[26]],  
[27]"outputsDef": [  
[28] { "name": "weight", "type":  
      "objective", "goal": "min"},  
[29]],  
[30]"algo": "GA",  
[31]"algoOptions": {  
[32]"numGenerations": 150,  
[33]"numPopulation": 59,  
[34]"mutationRate": 0.05,  
[35]"saveElites": 10,  
[36]"DOE": "random",  
[37]# "DOE": ["_job name_", -1],  
[38],  
[39]"jobOptions": {  
[40]"screenshots": True  
[41]}  
[42]}  
[43]  
[44]#job.createInputFile(jobDescription)  
[45]job.run(jobDescription)
```

ANEXO B – ALGORITMO PARA OTIMIZAÇÃO MULTIOBJETIVO

Algoritmo para otimização multiobjetivo: 02 funções objetivas.

Idem ao código anterior, exceto o acréscimo da linha [29].

```
[01]IDEM
[...]
[26]],
[27]"outputsDef": [
[28] { "name": "weight", "type":
    "objective", "goal": "min"}, 
[29] { "name": "powerOutput", "type":
    "objective", "goal": "max"}, 
],
"algo": "GA",
[...]
[45]#job.createInputFile(jobDescription)
[46]job.run(jobDescription)
```