

A modelação paramétrica e a geração de imagens por IA como ferramentas de Projeto em Arquitetura

A Computação e a AI como apoio
do arquiteto no trabalho de atelier

Fábio Alexandre Soares Remoaldo

Dissertação de Natureza Científica para a obtenção do Grau de Mestre
em Arquitetura, especialização em Arquitetura

Orientação Científica:

Professor Doutor Luís Mateus

Professor Doutor Victor Ferreira

Júri:

Presidente Doutor Luís Romão

Vogal Doutora Ljiljana Cavic

Vogal Doutor Luís Mateus

Documento Definitivo

Lisboa, FA ULisboa, julho, 2024

A modelação paramétrica e a geração de imagens por IA como ferramentas de projeto em Arquitetura

A Computação e a AI como apoio
do arquiteto no trabalho de atelier

Fábio Alexandre Soares Remoaldo

Dissertação de Natureza Científica para a obtenção do Grau de Mestre
em Arquitetura, especialização em Arquitetura

Orientação Científica:

Professor Doutor Luís Mateus
Professor Doutor Vítor Ferreira

Júri:

Presidente Doutor Luís Romão
Vogal Doutora Ljiljana Cavic
Vogal Doutor Luís Mateus

Documento Definitivo

Lisboa, FA ULisboa, julho, 2024

A Computação e a IA como apoio

A modelação paramétrica e a geração de imagens por IA como ferramentas de projeto em Arquitetura

A Computação e a AI como apoio ao
arquiteto no trabalho de atelier

Fábio Alexandre Soares Remoaldo

Dissertação de Natureza Científica
para a obtenção do Grau de Mestre em Arquitetura

Orientação Científica:

Professor Doutor Luís Mateus

Professor Doutor Victor Ferreira

Júri:

Presidente Doutor Luís Romão

Vogal Doutora Ljiljana Cavic

Vogal Doutor Luís Mateus

Documento Definitivo

Lisboa, FA Ulisboa, julho, 20

A Computação e a IA como apoio

RESUMO

A presente Dissertação de Mestrado investiga o potencial da modelação paramétrica e da inteligência artificial (IA) gerativa no âmbito de projeto e design arquitetónico, enfatizando a integração destes processos de trabalho computacional como ferramentas de apoio ao arquiteto em ambiente de atelier. Um trabalho elaborado sob uma componente teórica de investigação de bibliografia e conhecimentos consolidados e uma componente prática de testagem de processos alternativos de trabalho com base em experiências de trabalho do autor em ambiente de atelier.

A componente teórica explora a aplicabilidade e o valor do trabalho projectual sobre um pensamento computacional, que apesar de ser um dos pilares base da Ciência dos Computadores não está diretamente vinculado ao uso do computador. Tendo em conta que Computação é o estudo de processos de “problema-solving”, sendo o acumular multidisciplinar de conceitos, métodos e teorias presentes nas mais variadas áreas de estudo, salientando-se a teoria de sistemas, a teoria da cibernética aplicada a projeto e os processos algorítmicos, que fornecem as bases necessária para compreender as complexidades da arquitetura algorítmica, com um foco particular em metodologias que possibilitam a integração da modelação paramétrica no âmbito de projeto arquitetónico. Uma área de estudo interdisciplinar que têm as suas teorias e temáticas transpostas para a Arquitetura pelos investigadores e estudiosos do pós Segunda Guerra Mundial, uma época marcada pelo desenvolvimento tecnológico que permitiu o surgimento das metodologias de trabalho computacional atualmente praticadas na arquitetura e do atual computador, ferramentas de trabalho indispensável em qualquer atelier.

Além disso, a Dissertação explora algumas das possibilidades do uso das emergentes ferramentas de Inteligência Artificial com base em algoritmos gerativos, particularmente com o objetivo de gerar imagens, de certo modo automatizado, para fins de apresentação e comunicação do projeto ou exploração de ideias não consolidadas. Discute-se deste modo, o potencial da IA no processo criativo, com uma breve análise de alguns dos desafios e considerações éticas associados à adoção destas ferramentas. O estudo apresenta uma visão, que pretende demonstrar tanto as oportunidades como as limitações desta nova ferramenta, bem como a sua utilidade para a arquitetura.

Adicionalmente e como referido anteriormente, este trabalho assenta-se numa componente prática com base em experiências de trabalho pessoais do autor durante a sua participação no atelier Tomás Taveira e Associados. Partindo da análise de cinco casos de estudo, que correspondem a situações onde o problema arquitetónico foi resolvido por processos de trabalho convencionalmente praticados no atelier, mas que revelaram dificuldades e limitações no processo para a obtenção do resultado e/ou no resultado obtido. Deste modo, foram propostos processos alternativos que pretendem colmatar as fraquezas dos processos utilizados pelo atelier.

Em conclusão, esta dissertação destaca a importância de uma avaliação equilibrada e crítica das novas tecnologias na arquitetura, em particular a modelação paramétrica e a IA. Um trabalho que não pretende convencer aqueles que o leem a adotar de uma forma acrítica estas ferramentas, mas sim motivar a procura de novas possibilidades de trabalho em que as conclusões sobre as mesmas devem resultar da experiência real e prática avaliada em contexto, e não de especulações baseadas em preconceitos, por vezes arrogantes, sobre o desconhecido.

Palavras-chave: Pensamento Computacional | Arquitetura algorítmica | CAD (Computer Aided Design) | Modelação paramétrica | Geração de imagens por IA

ABSTRACT

This Master's Dissertation investigates the potential of parametric modeling and generative artificial intelligence (AI) in the context of architectural project and design, emphasizing the integration of these computational work processes as tools to support architects in a studio environment. The work is developed with a theoretical component of literature research and consolidated knowledge, and a practical component of testing alternative work processes based on the author's work experiences in a studio environment.

The theoretical component explores the applicability and value of design work based on computational thinking, which, although one of the foundational pillars of Computer Science, is not directly tied to the use of computers. Considering that computation is the study of "problem-solving" processes, involving the multidisciplinary accumulation of concepts, methods, and theories from various fields of study, particular attention is given to systems theory, cybernetics applied to design, and algorithmic processes, which provide the necessary foundations for understanding the complexities of algorithmic architecture. There is a particular focus on methodologies that enable the integration of parametric modeling within architectural design. This is an interdisciplinary field of study whose theories and topics were brought into Architecture by post-World War II researchers and scholars, a period marked by technological development that allowed for the emergence of the computational work methodologies currently practiced in architecture, as well as the modern computer, indispensable tools in any studio.

Furthermore, the Dissertation explores some of the possibilities of using emerging Artificial Intelligence tools based on generative algorithms, particularly for the purpose of generating images, in a somewhat automated manner, for project presentation and communication or the exploration of unrefined ideas. The potential of AI in the creative process is discussed, with a brief analysis of some of the challenges and ethical considerations associated with the adoption of these tools. The study presents a perspective aimed at demonstrating both the opportunities and limitations of this new tool, as well as its usefulness for architecture.

Additionally, as mentioned earlier, this work is based on a practical component drawn from the author's personal work experiences during participation in the Tomás Taveira and Associates studio. Based on the analysis of five case studies, which correspond to situations where the architectural problem was solved through conventionally practiced studio methods but revealed difficulties and limitations either in the process of obtaining the result or in the result itself. Consequently, alternative processes were proposed to address the weaknesses of the methods used by the studio.

In conclusion, this dissertation highlights the importance of a balanced and critical assessment of new technologies in architecture, particularly parametric modeling and AI. It is a work that does not aim to convince its readers to adopt these tools uncritically, but rather to motivate the search for new work possibilities, where conclusions about these tools should result from real and practical experiences evaluated in context, rather than from speculations based on sometimes arrogant prejudices about the unknown.

Keywords: Computational Thinking | Algorithmic architecture | CAD (Computer Aided Design) | Parametric modeling | AI-driven image generation

AGRADECIMENTOS

O presente trabalho conclui uma fase da minha vida que se demonstrou mais sofrida e longa do que havia antecipado, mas que felizmente termino com muito orgulho por ter conquistado um sonho. No entanto, esta vitória não é só minha, pois sem a ajuda, apoio e carinho dos meus amigos e familiares não teria conseguido concluir este ciclo da minha vida com sucesso.

Agradeço assim aos meus pais, Sérgio Remoaldo e Vanda Remoaldo por me terem apoiado e investido tudo o que tinham de seu em mim sem nunca pedirem algo em troca, que não fosse apenas o meu bem-estar. Agradeço ao meu irmão por todos os momentos de alegria que partilhou comigo, por me ter ajudado apenas por marcar a sua presença ao meu lado quando mais precisei. Agradeço a minha namorada, Beatriz Pedro, por ter sido o meu maior porto de abrigo e pela paciência que teve comigo, sem nunca recusar me auxiliar quando mais precisava.

Agradeço aos meus amigos Sérgio Hilário e Beatriz Gonçalves, que me deram a motivação para continuar a lutar, sem nunca me deixarem sentir só nesta jornada que parecia sem fim. Agradeço aos meus amigos Pedro Leitão e Tomás Campos, duas pessoas que conheci na faculdade, mas que rapidamente se tornaram companheiros para o resto da vida.

Agradeço aos meus orientadores, professor Doutor Luís Mateus e professor Doutor Vítor Ferreira, por terem direcionado o trabalho no caminho certo, mas principalmente por se terem entusiasmado com este trabalho tanto quanto eu. Agradeço ao professor Doutor Tomás Taveira, por me ter aceitado no seu atelier e me ter dado total liberdade para usar o seu atelier como caso de estudo para a componente prática desta Dissertação.

Agradecer a todos os docentes que ajudaram a formar o arquiteto que virei a ser. Agradeço por fim a todos aqueles que infelizmente ficaram por mencionar, mas que sempre desejaram o meu bom sucesso.

ÍNDICE

RESUMO	V
ABSTRACT	VII
AGRADECIMENTOS	IX
ÍNDICE	XI
ÍNDICE DE FIGURAS	XIII
ÍNDICE DE ACRÓNIMOS	XVII
1 INTRODUÇÃO	1
1.1 Enquadramento Temático	2
1.2 Objetivos	4
1.3 Metodologia	6
1.4 Estrutura e organização	8
2 COMPUTAÇÃO APLICADA À ARQUITETURA	11
2.1 Recorte Temporal	12
2.2 Pensamento Computacional	22
2.2.1 Processo Algorítmico	31
2.2.2 Teoria de Sistemas	40
2.2.3 Teoria da Cibernética	48
3 ARQUITETURA ALGORÍTMICA	57
3.1 AAD (Algorithm Aided Design)	58
3.2 Sistema Algorítmico	60
3.3 Modelação Paramétrica	65
4 INTELIGÊNCIA ARTIFICIAL (IA) APLICADA À ARQUITETURA	75
4.1 IA – Inteligência Artificial	76
4.2 IA como ferramenta de produção de imagens	85
4.3 IA como ferramenta de produção de projetos	93
4.4 Autor? PC vs. Operador	94
5 O CASO DO ATELIER TT	101
5.1 Fluxo de trabalho no Atelier	102
5.2 Estrutura de avaliação e enquadramento de casos de estudo	106

5.3 Casos de estudo	107
5.4 Síntese	124
6 DA TEORIA PARA A PRÁTICA: APLICAÇÃO DA MODELAÇÃO PARAMÉTRICA E DA IA	127
6.1 Dificuldades VS Oportunidades	128
6.2 Estrutura de apresentação dos processos alternativos de trabalho..	130
6.3 Processos Alternativos de Trabalho: Modelação Paramétrica	131
6.4 Processos Alternativos de Trabalho: Inteligência Artificial (IA)	162
6.5 Síntese	178
7 CONSIDERAÇÕES FINAIS	185
8 BIBLIOGRAFIA	189

ÍNDICE DE FIGURAS

Tabela

Tabela 1 | Tabela comparativa de sistemas abordados (Tabela do autor) - p.63

Tabela 2 | Tabela comparativa das dificuldades verificadas com as oportunidades de resolução (Tabela do autor) - p.129

Tabela 3 | Tabela comparativa de Casos estudados com Processos alternativos executados. (Tabela do autor) - p.178

Tabela

Fig. 0 | Obras de referência feitas por Projeto computacional. (Neves et al., 2023, p.5) - p.5

Fig. 1 | Obras de referência e temáticas estudadas. (Imagem de autor) – p.7

Fig. 2 | Projeto The Bombe de Alan Turing (1939). (<https://www.cryptomuseum.com>) - p.13

Fig. 3 | Electronic Numerical Integrator and Computer (ENIAC) (1943). (<https://www.britannica.com>) - p.13

Fig. 4 | Processo convencional de desenho técnico. (howtorhino.com) - p.16

Fig. 5 | Processo atual de desenho técnico. (www.worthyhardware.com) - p.16

Fig. 6 | DAC - Design Automated by Computer (1957). (time.graphics) - p.18

Fig. 7 | "Sketchpad: A Man Machine Graphical Communication System.". (bimaplus.org) - p.18

Fig. 8 | Utilização de Sketchpad. (en.wikipedia.org) - p.18

Fig. 9 | IBM PC 5150, lançado agosto 1981. (arstechnica.com) - p.20

Fig. 10 | Catia V1 utilizado no IBM station em 1982. (www.reddit.com) - p.20

Fig. 11 | Evolução do desenho de projeto. (www.fiverr.com) - p.25

Fig. 12 | Receita de bolonhesa. (www.pingodoce.pt) - p.33

Fig. 13 | A palavra Sistema gera duas ideias que se interligam (Imagem do autor) - p.42

Fig. 14 | Relação de sujeito com 1º, 2º, 3º Ordem de Cibernética (Ashby, 2020) - p.49

Fig. 15 | Diferença entre 1º ordem de Cibernética com 2º segunda ordem (Autor Desconhecido) - p.450

Fig. 16 | Generator Project (eliza-pert.medium.com) - p.51

Fig. 17 | Metodos de estudo por *feedback* ("Schemes for Healthy Buildings," 2018) - p.53

Fig. 18 | Processo de análise algorítmica (spbim.com.br) - p.66

Fig. 19 | Modelação digital de elemento arquitetónico da sagrada família (www.semanticscholar.org) - p.68

Fig. 20 | Maqueta pendular da Sagrada família de A.Gaudi (www.flickr.com) - p.68

Fig. 21 | Diagrama de Voronoi (Imagem do autor) - p.69

Fig. 22 | The Fry Building (awrn.co.uk) - p.69

Fig. 23 | Galeria Warminska (framestudio.eu) - p.69

Fig. 24 | *Etapas de modelação digital convencional de um cilindro (Imagem de autor)* - p.70

Fig. 25 | Modelação paramétrica do projeto NEF Kagithane 03 (www.youtube.com) - p.70

Fig. 26 | Lógica de ligação de Componentes em programação por *nodes* (Autor Desconhecido) - p.71

Fig. 27 | Construção de linha via Grasshopper (Imagem de Autor) - p.72

Fig. 28 | Três Leis da robótica de Isaac Asimov (www.researchgate.net) - p.77

Fig. 29 | Deep Blue (phamoxtech.com) - p.78

Fig. 30 | Alpha Go (abertoatedemadrugada.com) - p.78

Fig. 31 | Classe de algoritmo: Deep Learning (De Molina Rius, 2023, p.12) - p.83

Fig. 32 | Exemplo de Render 1 (www.cuubstudio.com) - p.86

Fig. 33 | Exemplo de Render 2 (www.realspace3d.com) - p.86

Fig. 34 | Resultados de DALL-E (zapier.com) - p.88

Fig. 35 | Resultados de Midjourney (zapier.com) - p.88

Fig. 36 | Resultados de Stable Diffusion (zapier.com) - p.89

Fig. 37 | Processo de difusão vs Redução de ruído (www.linkedin.com) - p.89

Fig. 38 | Modelo de Difusão (www.researchgate.net) - p.90

Fig. 39 | Pintura por técnica de pontilhismo (www.peonyandparakeet.com) - p.90

Fig. 40 | Lógica do modelo CLIP (<https://lilianweng.github.io>) - p.91

Fig. 41 | IA para geração de imagens estilo "Van Gogh" (getimg.ai) - 91

- Fig. 42 | Polêmica em volta da IA (edition.cnn.com) - p.92
- Fig. 43 | Exemplo de processo de esboço manual (www.behance.net) - p.95
- Fig. 44 | Processo de sketch pelo uso de IA, feito por Tim Fu (Wadhwa, 2023a) - p.96
- Fig. 45 | Frames do trabalho de Refik Anadol (Cianciolo & Chia, 2023) - p.98
- Fig. 47 | Obras de referência do arquiteto Tomás Taveira: a) Torres das Amoreiras (imperdivel.pt); b) Sede do BNU (pbase.com); c) Estação de metro das Olaias. - p.102
- Fig. 48 | Organigrama de logica de trabalho com xRef (Imagem do autor) - p.104
- Fig. 49 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Modelo original em Sketch Up. - p.108
- Fig. 50 | Processo de construção de caso em estudo (Imagem do autor) - p.109
- Fig. 51 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion - p.113
- Fig. 52 | Processo de construção de caso em estudo (Imagem do autor) - p.114
- Fig. 53 | Processo de construção de caso em estudo (Imagem do autor) - p.116
- Fig. 54 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion - p.117
- Fig. 55 | Processo de construção de caso em estudo (Imagem do autor) - p.119
- Fig. 56 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion - p.120
- Fig. 57 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Imagem renderizada e trabalhada em Twinmotion - p.122
- Fig. 58 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Imagem renderizada e trabalhada em Twinmotion - p.123
- Fig. 59 | Organograma processo alternativo 04 | Processos de produção de estudo prévio - Otimização de área (Imagem do autor) - p.132
- Fig. 60 | Processo de otimização de solução (imagens do autor): a) Resultado Final com volumetria e área otimizadas + Sistema Grasshopper gerador do objeto; b) Solução inicial a otimizar + Parâmetros implementados; c) Solução com planta de implantação otimizada + Informações obtidas. - p.134
- Fig. 61 | Organograma processo alternativo 01 | Processos de produção de estudo prévio – Proposta de múltiplos edifícios (Imagem do autor) (Imagem do autor) - pp.136-137

Fig. 62 | Variantes de solução gerada (Imagens do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4. - p.141-142

Fig. 63 | Organograma processo alternativo 02 | Hipótese de Fachada 1. (Imagem do autor) - p.144-145

Fig. 64 | Variantes de solução gerada (Imagem do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4. - pp.148-149

Fig. 65 | Organograma processo alternativo 03 | Hipótese de Fachada 2 (Imagem do autor) - pp.152-153

Fig. 66 | a) Variantes de solução gerada (Imagem do autor); b) Sistema grasshopper gerador do Objeto - p.155

Fig. 67 | Organograma processo alternativo 04 | Teto acústico Escola de Dança - p.157

Fig. 68 | Sistema grasshopper com cluster e "interface" de utilização planeada. (Imagem do autor) - p.159

Fig. 69 Variantes de solução gerada (Imagem do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4. - p.160-161

Fig. 70 | Organograma processo alternativo 05 | Renders (Imagem do autor) - p.162

Fig. 71 | a) Primeira abordagem; b) Segunda abordagem. (Imagem do autor) - P.163

Fig. 72 | Tipos de algoritmo testados (Imagens do autor): a) Content control; b) Segmentation; c) Scribble_xdog; d) Mlsd. - p.164

Fig. 73 | Recolha de resultados com maior interesse. Esta amostra pretende mostrar os bons e os resultados "caóticos" gerados do esboço base do arquiteto Tomás Taveira (Imagem do autor) - p.165

Fig. 74 | Resultados de Prompt Crafting (Imagens do autor) - p.166

Fig. 75 | Resultados de Interactive Design (Imagens do autor) - p.167

Fig. 76 | Melhores resultados de Interactive Design testados no *Render mode* (Imagem de autor) - p.167

Fig. 77 | Resultados obtidos com o software LookX AI (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido. - p.169

Fig. 78 | Resultados obtidos com o software rendair (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido. - p.170

Fig. 79 | | Resultados obtidos com o *software* PromeAI (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido. - p.171

Fig. 81 | Comparação de a) objeto de estudo (Imagens com direitos autorais de atelier TT) com b) resultado final (Imagem de autor). - p.173

Fig. 82 | Principais mudanças aplicadas por ordem alfabética: a), b), c) e d) (Imagens do autor) - pp.174-175

Fig. 83 | Imagens geradas em render (Imagem do autor): a) Hipótese de solução 1; b) Variante 1 de solução 1a; c) Variante 2 de solução 1a; d) Hipótese de solução 2; e) Hipótese de solução 2. - pp. 176-177

Fig. 84 | Processo alternativo para Regio Difference. (Imagem do autor) - p.179

ÍNDICE DE ACRÓNIMOS

2D - Duas Dimensões

3D - Três Dimensões

AAD - *Algorithm Aided Design* (Design com apoio de Algoritmos)

AEC - Arquitetos, Engenheiros e Construção

AGI - *Artificial General Intelligence* (Inteligência Artificial Generalizada)

AI - *Artificial Intelligence* (Inteligência Artificial)

ANI - *Artificial narrow intelligence* (Inteligência Artificial Limitada)

AR - Augmented Reality (Realidade Aumentada)

ASI - *Artificial super intelligence* (Super Inteligência Artificial)

BIM - Building Information Modeling (Modelagem de Informação de Construção)

CAAD - Computer-aided Architectural Design (Projeto de Arquitetura com apoio de computador)

CAD - Computer Aided Design (Design com apoio de computador)

CAM - Computer-aided Manufacturing

CNC - Computer numerical control (Controle Numérico Computadorizado)

DAC - (Design Automated by computer)

ENIAC - *Electronic Numerical Integrator and Computer* (ENIAC)

FDM - Fused Deposit Modeling (Modelagem de Depósito Fundida)

GRASP - (Generation of Random Access Site Plans)

MIT - (Massachusetts Institute of Technology)

PC - Portable Computer (Computador Portatil)

SLA - Stereolithography (Estereolitografia)

SRS - *String Rewriting System*

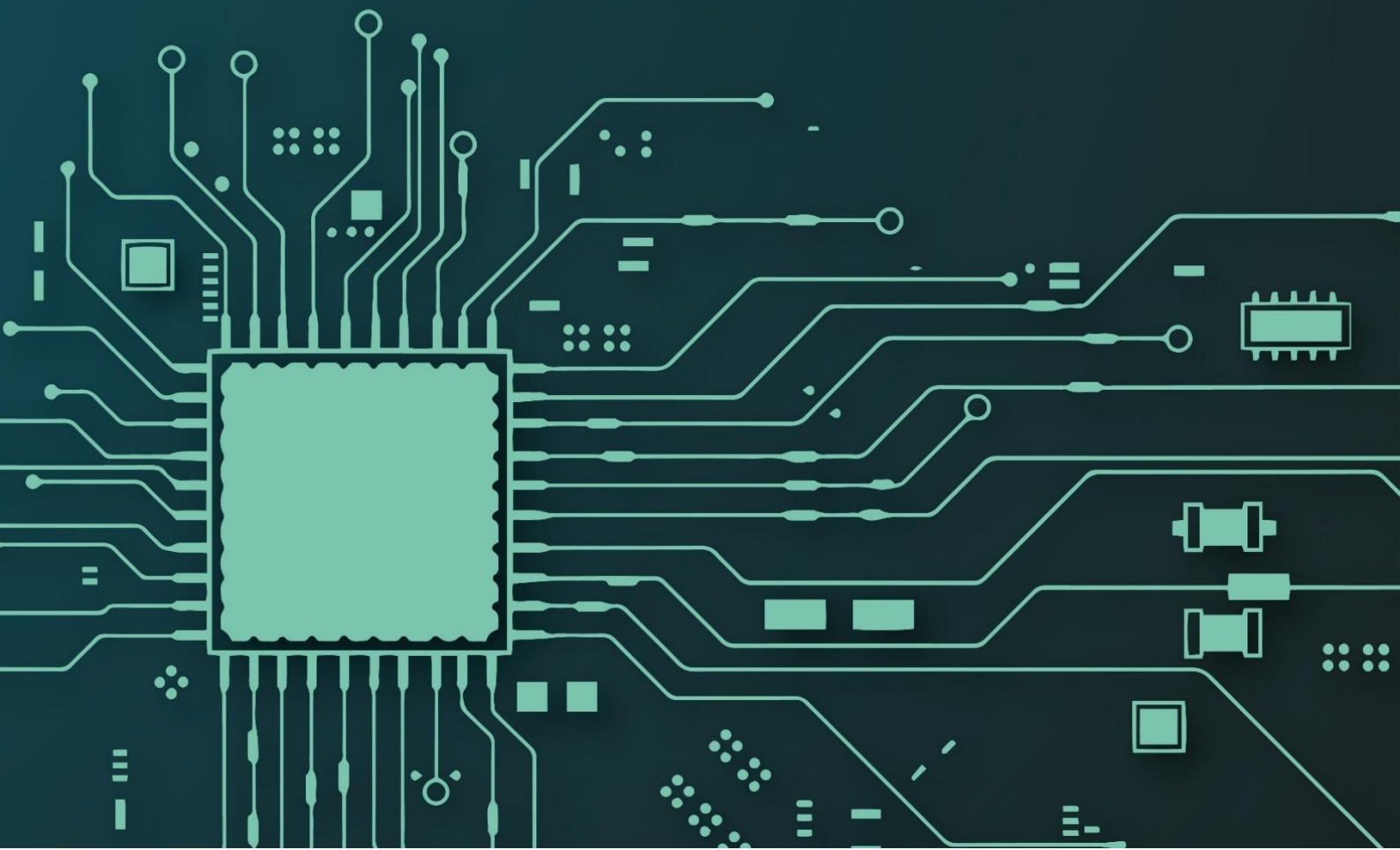
STL - Stereolithography (Estereolitografia)

TT - Tomás Taveira e Associados

VR - Virtual Reality (Realidade Virtual)

"It is not the strongest of the species that survives, not the most intelligent that survives. It is the one that is the most adaptable to change."

Darwin, C. (1859). The origin of species





1. INTRODUÇÃO

1.1 | Enquadramento Temático

Arquitetura sem arquitetos é um conceito abordado numa exposição no MoMA de Nova Iorque por Bernard Rudofsky em 1964. Este evento funcionou como uma antestreia ao livro que lançou no mesmo ano "*Architecture without Architects (A Short Introduction to Non-Pedigreed Architecture)*" (Crook & Crook, 2023). Uma exposição constituída maioritariamente por fotografias em preto e branco de uma arquitetura não formalizada, descaracterizada, vernacular ou primitiva realizada pelo mundo inteiro em diversas épocas, onde a arquitetura era feita sem especialistas na área e licenciados (Crook & Crook, 2023). Esta ideologia é corroborada por Bernard Rudofsky no seu livro com "Nós temos muito que aprender disso que foi a arquitetura antes de se tornar uma arte de especialistas" (Cronologia do Pensamento Urbanístico, s.d.). Uma exposição realizada com a intenção de romper com as ideias já consolidadas, no contexto dos anos sessenta, de uma arquitetura comercial, universal e de certo modo presa na exploração do espaço. Apesar de ter sido uma exposição que gerou controvérsia ela teve a capacidade de expor o espectador moderno, a romper e questionar padrões pré-estabelecidos, abrindo os horizontes sobre a arte que é o "fazer arquitetura", "uma arte comunitária produzida não por alguns intelectuais ou especialistas, mas pela atividade espontânea e contínua de um povo inteiro, depositados numa herança comum e obedecendo as lições de uma experiência comunal." (Cronologia do Pensamento Urbanístico, s.d.).

Atualmente a arquitetura está a passar por uma nova fase, ou tentativa, de "Arquitetura sem arquitetos". Não se pretende afirmar que o arquiteto seja excluído do processo, mas sim enfatizar a presença, cada vez mais frequente, de processos computacionais semi-automatizados e automatizados no processo de projeto. Com a introdução do computador nos ateliers de arquitetura, como uma ferramenta de produção, fabricação e ideação digital, que permite romper com os limites de conceção de um projeto, onde o arquiteto realiza a ideia, o conceito e a máquina fábrica segundo as ordens deste. Inclusive já são inúmeros os projetos a nível internacional que apenas foram possíveis de conceber, pois os ateliers adotaram os métodos computacionais, para auxiliar o arquiteto na conceção do projeto. Exemplo disto é Zaha Hadid, uma arquiteta pioneira no uso de *software* para a fabricação das suas propostas, conhecida pela sua arquitetura caracterizada pela fluidez, abstração e desconstrução da forma.

Zaha Hadid tirou proveito da computação para conceber a estrutura, bem como a malha que produz o acabamento dos seus projetos, como por exemplo o Hyder Aliyev Centre em Baku (Azerbaijan), City of dreams Hotel Tower em Macau e o Centro Heydar Aliyev em Baku (Azerbaijan). Tal como Zaha Hadid imensos são os arquitetos que tiram proveito da computação nas suas obras internacionais como Frank Gehry e a sua obra Walt Disney Concert Hall em Los Angeles (USA), considerada um dos primeiros edifícios a ser criado com o auxílio de modelação paramétrica, só assim foi possível viabilizar o design interior do espaço para maximizar o potencial acústico, uma cooperação feita com Minoru Nagata um engenheiro acústico (Sawantt, 2021).

Atualmente os arquitetos dispõem, graças a evolução tecnológica, de uma panóplia de possibilidades de design e estrutura, onde a concessão é realizada pelo arquiteto, mas o resultado e a fabricação é realizado pela máquina. Atualmente já existem programas que por avaliação paramétrica, conseguem fornecer uma série de respostas às adversidades que o arquiteto enfrenta, um exemplo disso é o trabalho realizado pelo arquiteto José Beirão publicado no livro "*CityMaker: Designing Grammars for Urban Design*", que ao avaliar os padrões urbanos de uma cidade consegue fornecer uma solução para um espaço, que depois pode ser ajustada pelo arquiteto.

Para além, dos novos IA gerativos que estão a inundar a comunidade mundial com novas formas de trabalho. Onde o seu impacto na arquitetura por enquanto, está focado na geração de imagens, mas gradualmente estes *softwares* irão passar a abordar a fabricação de projeto. Uma situação em que alguns arquitetos temem pela sua profissão, enquanto outros veem na IA novas formas de abordar os processos de trabalho projectual.

Os novos *softwares*/ferramentas apresentam um novo capítulo na história da arquitetura, em que este paradigma poderá ser a nova “Arquitetura sem arquitetos”, que rompe com as limitações da conceção arquitetónica, abrindo os horizontes sobre a arte que é o “fazer arquitetura” pelo uso de processos computacionais. Deste modo, tendo em conta o contexto atual, o seguinte trabalho incide sobre o potencial pratico das novas tecnologia como ferramentas que auxiliam o arquiteto na prática da sua profissão.

1.2 | Objetivos

A Dissertação incidirá sobre a presença da tecnologia na arquitetura, mais concretamente a utilização do computador – ferramenta cada vez mais presente e indispensável para qualquer arquiteto - em ambiente de atelier como suporte do processo de trabalho para além da emulação de processos tradicionais. Uma máquina que, pela infinidade de aplicações existentes e outras em desenvolvimento, tem capacidade para fornecer ao arquiteto uma série de oportunidades e possibilidades de uso, em parte limitadas apenas pela criatividade do seu utilizador. Estas novas tecnologias podem auxiliar o arquiteto no fluxo de projeto desde o estudo conceptual, geralmente presente nas fases iniciais, bem como no auxílio à produção de toda a documentação necessária à construção ou, inclusive, influenciar o modo como a própria construção se desenrola. O trabalho de atelier não se resume apenas à concepção do projeto de arquitetura. Deste modo, é importante perceber também, qual a utilidade das ferramentas digitais em diferentes momentos de trabalho, como por exemplo, a comunicação do projeto com o cliente. No entanto, a utilização do computador no âmbito da arquitetura é, muitas vezes, considerada como uma barreira à criatividade, retirando a capacidade de produzir espaços únicos e de elevada qualidade arquitetónica. Apesar desta ser uma questão subjetiva, a utilização do computador não deve ser percecionada como uma barreira à criatividade, mas sim como um potenciador e gerador de novos horizontes.

“As generally known, computers are pretty dumb and only perform what they are programmed to perform...” (Herr & Fisher, 2001)

Assim, e seguindo a premissa exposta acima, pretende-se incidir e refletir sobre as capacidades da utilização do computador, bem como as vantagens e desvantagens inerentes à sua utilização em relação ao ato de “fazer arquitetura” em ambiente de atelier. Deste modo, será realizada a testagem e a prática em primeira mão de diversos métodos e possibilidades de aplicação, baseadas em diferentes processos de trabalho com conceitos próprios (modelação paramétrica, fabricação digital, IA gerativos). As alternativas propostas serão sempre para uma melhoria dos métodos de trabalho aplicados pelo arquiteto. A escolha da ferramenta será baseada na sua capacidade para responder, com maior eficácia, às diversas problemáticas do trabalho de atelier, desde fases iniciais de concepção à apresentação final do projeto.

1.3 | Metodologia

Estado do Conhecimento

O teor deste trabalho implicou uma análise de diversas bibliografias de autores e investigadores dos temas principais, muitos nunca abordados durante o ciclo de estudos de arquitetura, havendo um maior foco na componente teórica inerente aos processos de trabalho com base na modelação paramétrica e na inteligência artificial com aplicabilidade na área de trabalho. A grande maioria dos autores e investigações estudadas surgem do pós segunda guerra mundial, onde inclusive os avanços tecnológicos desse tempo influenciaram o desenvolvimento das tecnologias presentes na atualidade. Este estudo e trabalho de análise permite um maior entendimento sobre os processos de trabalho utilizados no atelier Tomás Taveira e Associados, mas principalmente permitem uma avaliação crítica sobre os processos de trabalho alternativos propostos nesta Dissertação Final de Mestrado, com o intuito de colmatar as dificuldades e limitações de trabalho verificadas em ambiente de atelier.

Análise de Caso

A análise de caso foi conduzida pela participação em ambiente de atelier de modo a averiguar as dificuldades dos profissionais durante a conceção e desenvolvimento do projeto. Em concreto, remeto para a participação do autor da tese na equipa do Atelier do Arquiteto Tomás Taveira entre 2022 e 2023. Durante esse período participou em projetos que serviram, na fase seguinte, como inspiração para o desenvolvimento de processos de trabalho visando a otimização da produção de soluções arquitetónicas. Dessas experiências, foram selecionados cinco casos de estudo em que os problemas de projeto foram resolvidos por processos convencionais de trabalho, no contexto do atelier em questão, que apresentaram limitações ou dificuldade, durante a execução da tarefa e/ou no resultado obtido. Deste modo, a análise e avaliação dos mesmos, permitiu uma visão com maior amplitude das possibilidades e limitações dos processos utilizados em contexto real de trabalho em ambiente de atelier, salientando-se a não substituição do importante papel do arquiteto, mas sim o valor sobre o acumular de conhecimentos e desenvolvimento de metodologias que permitem ao arquiteto beneficiar da constante evolução tecnológica.

Aplicação Prática

A avaliação prática é composta pelos diferentes métodos e ferramentas computacionais que foram desenvolvidos e testados a partir de uma revisitação crítica dos casos de estudo de modo a inferir processos alternativos de trabalho. Incluindo a utilização de software de modelação paramétrica para criar formas arquitetónicas inovadoras e a aplicação de algoritmos de IA para a geração de imagens, com o intuito de otimizar os processos de projeto. A experimentação prática foi fundamental para validar as teorias e conceitos inerentes à computação e a sua aplicabilidade no âmbito da arquitetura.

Avaliação e Síntese

Após a aplicação prática, os resultados foram avaliados em termos de eficácia, automatização e impacto na obtenção da solução para o problema arquitetónico. Esta avaliação considerou tanto aspectos técnicos como a possibilidade de exploração de variantes e hipóteses de solução, de modo a inferir as vantagens e desvantagens das ferramentas utilizadas, mas mais importante ainda o processo e o modo como, neste, se aborda o problema a resolver. A síntese dos resultados proporciona uma visão consolidada sobre como as tecnologias digitais (modelação paramétrica e IA) quando integradas no processo de projeto de forma a potencializar a criatividade e a produtividade do arquiteto.

Conclusão

A metodologia adotada nesta pesquisa permitiu uma abordagem holística e prática sobre o uso de tecnologias computacionais na arquitetura, com ênfase na modelação paramétrica e na IA. Através da revisão bibliográfica, análise de caso e aplicação prática, foi possível desenvolver um melhor entendimento sobre as capacidades e limitações dessas tecnologias, fornecendo assim algumas referências para uma possível adoção de novas práticas no campo da arquitetura digital.



Fig. 1 | Obras de referência e temáticas estudadas. (Imagem de autor)

1.4 | Estrutura e organização

A Dissertação foi organizada de modo a fornecer ao leitor uma progressão lógica dos diferentes temas e subtemas abordados, desde a introdução do tema até a aplicação prática terminada com as conclusões acerca dos resultados obtidos e considerações finais.

1. Introdução

A introdução define o enquadramento temático da pesquisa, estabelecendo o contexto em que se insere o estudo, para além dos objetivos gerais e específicos deste trabalho. Adicionalmente é apresentada a lógica da metodologia de trabalho utilizada para o bom sucesso desta Dissertação Final de Mestrado. De seguida é apresentada uma visão geral da estrutura da tese, com uma breve descrição do conteúdo de cada capítulo.

2. Computação aplicada à Arquitetura

O capítulo explora o impacto da tecnologia na arquitetura, fornecendo uma visão sobre o paradigma tecnológico. Em seguida, é abordado o pensamento computacional aplicado à arquitetura, onde é detalhado o processo algorítmico e as implicações do mesmo no processo de resolução de problemas. A Teoria de Sistemas e a Teoria da Cibernética são analisadas como fundamentos teóricos que influenciam a prática arquitetónica, fornecendo uma base para o entendimento de como a computação pode ser integrada ao processo de criação arquitetónica.

3. Arquitetura Algorítmica

Focado no Design Assistido por Algoritmos (AAD), este capítulo introduz o conceito e discute as aplicações na arquitetura. Os sistemas algorítmicos são descritos, com destaque sobre o modo como estes podem ser utilizados para resolver problemas complexos de design e projeto arquitetónico. É feito um aprofundamento da modelação paramétrica, pela demonstração das capacidades e limitações como ferramentas de design, na produção e criação de formas arquitetónicas inovadoras a partir de processos de trabalho otimizados.

4. Inteligência Artificial aplicada à Arquitetura

Este capítulo investiga a utilização da Inteligência Artificial (IA) na arquitetura. Inicialmente, a IA é introduzida como uma ferramenta para a produção de imagens arquitetónicas, mostrando como pode ser utilizada para gerar representações visuais. Em seguida, a IA é explorada como uma ferramenta para a produção de projetos arquitetónicos, discutindo as vantagens e desafios destas abordagens. Adicionalmente são abordadas as questões éticas de direitos de autor, analisando o papel do operador humano em comparação com o computador no processo de criação arquitetónica.

5. O caso do atelier TT

Neste capítulo, o fluxo de trabalho do atelier Tomás Taveira e Associados é descrito, proporcionando uma visão dos processos de trabalho utilizados no atelier. De seguida são apresentados 5 casos de estudo provenientes de 2 projetos de arquitetura, que ilustram diferentes aspectos do processo de projeto e design, incluindo variantes de fachadas e um teto acústico para uma escola de

dança. Estes casos de estudo são avaliados e discutidos salientando a eficácia, as dificuldades e limitações dos métodos utilizados, fornecendo as informações necessárias de modo a inferir os processos alternativos que irão resolver as mesmas problemáticas. As imagens presentes neste capítulo dos resultados obtidos (autoria do atelier Tomás Taveira e Associados), bem como as informações fornecidas sobre o atelier foram autorizadas pelo arquiteto Tomás Taveira para fins académicos de estudo e avaliação desta Dissertação Final de Mestrado.

6. Da Teoria para a Prática: Aplicação da Modelação Paramétrica e da IA

“If you are a student learning parametric design, your aim is the practitioner’s craft. Everything relevant to the practitioner applies to you too. Designs can only be learned by doing. “Talkitecture ” is a derogatory term, reserved for those who discuss but do not draw. Don’t draw it onto yourself. You need more; particularly, you need to understand how parametric systems work; how their structure makes them perform and how people have used and are using them to do design.” (Woodbury, 2010, pp.8-9)

Inspirado pela citação acima, este capítulo faz a transição da teoria para a prática, discutindo as dificuldades e oportunidades encontradas na implementação dos processos alternativos em questão e o uso das “novas” tecnologias na arquitetura. São apresentados processos alternativos de trabalho, focados na modelação paramétrica e na Inteligência Artificial, com exemplos específicos que mostram como a problemática foi abordada, e o modo como o processo serviu para otimizar, automatizar a produção arquitetónica. O capítulo conclui com uma síntese das alternativas apresentadas, destacando os principais benefícios e desafios associados a cada uma. Optou-se por incluir a conclusão do trabalho no final deste capítulo por ser aqui que se centra o foco da nossa pesquisa.

7. Considerações Finais

A conclusão resume os principais achados da pesquisa, destacando as contribuições teóricas e práticas. São discutidas as implicações dos resultados obtidos e apontadas possíveis direções para futuras pesquisas, sugerindo como os conceitos e técnicas explorados na tese podem ser desenvolvidos e aplicados em estudos futuros. Esta secção finaliza a tese, reforçando o valor que reside na integração da computação e da Inteligência Artificial no âmbito da arquitetura e a sua potencialidade para transformar a prática arquitetónica.





2. COMPUTAÇÃO APLICADA À ARQUITETURA

2.1 | Recorte Temporal

The human ecosystem so obviously layer upon layer of electronic systems and those systems are no longer simply outside and around the limits of the body but deep inside the body, moving and multiplying its limits. (Wigley, 2010, p.57)

O ser humano atualmente, já não “existe” sem o uso das tecnologias. O desenvolvimento tecnológico permitiu a criação de diversas máquinas e aparelhos, como o telemóvel, o computador, o cartão bancário, ferramentas essenciais num mundo cada vez mais informatizado. Elas vieram para ficar, ou pelo menos até serem dadas como obsoletas e assim serem substituídas por outras. O recorrente avanço das novas tecnologias, novas ferramentas (digitais e físicas) e as metodologias de aplicação “inovadoras”, têm-se vindo a manifestar a um ritmo cada vez mais acelerado, ou seja, reduzindo o intervalo de tempo entre cada nova “invenção” ou update/upgrade das já existentes. Este paradigma é verificado por Maggie Toy (1995) em *“Changes in the cyberworld are made everyday; a week is a very long time in computers.”* (p. 7).

São vários os exemplos deste progresso tecnológico, onde muitos destes desenvolvimentos rapidamente são disseminados na sociedade, pela necessidade e/ou do acesso generalizado ao público (como por exemplo sistemas open source). O fácil acesso às tecnologias permite o desenvolvimento de produtos e subsequentemente o surgimento de diversas variações, que atendem a áreas específicas de aplicação. Uma tecnologia recente que se alinha com a situação acima referida é o ChatGPT desenvolvido pela OpenAI. Neste sentido, empresas como a Salesforce.com (2004) são exemplo da aposta no desenvolvimento constante de novas tecnologias. Fundada por Marc Benioff um dos pioneiros da exploração de cloud computing para a venda de *software* “como um serviço”, abandonando a ideia de um elemento físico (*hardware*) como distribuidor do digital (*software*). *“The only constant in the technology industry is change.”* é uma afirmação recorrentemente pronunciada por Marc Benioff, que dificilmente irá ficar desatualizada (Simonite, 2011). Um dos desafios da sociedade do século XXI é proporcionado por esse perpétuo desenvolvimento que implica uma atualização constante nas diversas áreas. Esta ideia é corroborada por Ganesan (2017) quando afirma que *“In our present-day world, time moves very quickly. Technology, economics and politics have accelerated the pace in which we are expected to move.”* (p.10). As previsões do futuro são sempre incertas, mas o impacto da tecnologia sente-se em todo o lado.

(...) fueled by wartime advances that had taken place in electronics, then accelerated explosively by the appearance of new technologies ... This revolution has spread throughout the world in just a few decades. It has been an order of magnitude faster than the Industrial Revolution (...) (Mitchell & McCullough, 1995, p.3)

Com um olhar crítico sobre a história da Humanidade, globalmente, observamos que existe uma grande tendência por parte do ser humano, em gerar as suas maiores descobertas e avanços tecnológicos em tempos de guerra ou crise. São vários os exemplos que demonstram esta correlação, que ocorre da

necessidade de sobrevivência, numa tentativa de ganhar vantagens sobre o inimigo, neste caso através de avanços tecnológicos que possam ser canalizados para o esforço de guerra. A Segunda Guerra Mundial é o principal exemplo de um período de desenvolvimento tecnológico súbito, e com grande influência no contexto contemporâneo atual (2023).

*Even though the scientist Alexander Fleming discovered the antibacterial properties of the *Penicillium notatum* mold in 1928, commercial production of penicillin did not begin until after the start of World War II. As American and British scientists worked collectively to meet the needs of the war, the large-scale production of penicillin became a necessity. (Burton, 2020)*

O pós-guerra permitiu uma reavaliação das tecnologias aplicadas ao serviço militar, em prol da sociedade em geral. Assim, inovações como o *Cavity Magnetron* (1940), que forneceu um importante *upgrade* aos sistemas de radar da época, atualmente faz parte do comum microondas existente em grande parte das habitações. O computador como conhecemos hoje, foi influenciado por certos projetos militares que decorreram durante a Segunda Guerra Mundial. Salientam-se o famoso projeto *The Bombe* (A Bomba) liderado pelo britânico Alan Turing em 1939. O projeto tinha como objetivo desenvolver uma máquina que tivesse a capacidade de decifrar a Enigma (a máquina de encriptação Nazi), revolucionando o processamento de dados algorítmicamente para a resolução de um problema específico (Cox, 2018). No entanto aquele que é considerado o primeiro computador, teve origem nos projectos Electronic Numerical Integrator and Computer (ENIAC) desenvolvido em 1943 pelos norte-americanos John Eckert e John Mauchly, e tinha a capacidade de rapidamente processar centenas de dados (Burton, 2020).

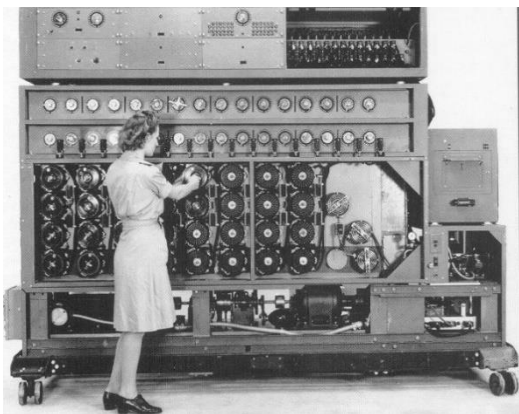


Fig. 2 | Projeto The Bombe de Alan Turing (<https://www.cryptomuseum.com>).



Fig. 3 | Electronic Numerical Integrator and Computer (ENIAC) (1943). (<https://www.britannica.com>)

Uma guerra que, para além de motivar o desenvolvimento tecnológico, deixou para trás uma Europa destruída e sem condições habitacionais. Pela necessidade de uma solução rápida e por um otimismo já pré-estabelecido quanto ao desenvolvimento tecnológico, nas décadas seguintes surgiu uma

época pós-industrial proporcionada pela computadorização (*computerization*) (Mitchell & McCullough, 1995, p.3).

Just as the Industrial Revolution replaced human muscle power by energy-consuming machines, the Computer Revolution is replacing human brain power by information-processing machines, the Computer Revolution is replacing human brain power by information-processing machines. (Mitchell & McCullough, 1995, p.3)

A Revolução do computador trouxe novas perspectivas de trabalho e mudanças de prioridades, afetando a economia mundial. Passou assim a existir uma economia focada na indústria de disseminação, processamento e armazenamento da informação. A segunda metade do século XX passa a estar assente em três pilares económicos, a apropriação e desenvolvimento dos recursos naturais, desde a produção de alimentos à extração de energias fósseis; a produção industrial de bens materiais acompanhada do interesse pelos meios de transporte de mercadorias e passageiros; e por último, a indústria relacionada com as atividades computacionais, a produção de componentes e dispositivos eletrónicos, o desenvolvimento de armazenamento de informação (capacidade de memória) e a partilha acessível e de acesso rápido a essa mesma informação (Mitchell & McCullough, 1995, p.3).

Carl August Sandburg poeta e historiador, escreveu em 1957 *“Money is power, freedom, a cushion, the root of all evil, the sum of blessings.”* esta frase reflete um mundo cada vez mais dominado pela economia de mercado, uma frase que reflete o impacto da economia do século XX. O sistema financeiro é uma força que atua como um impulsionador do crescimento, com impacto direto neste caso da cidade. Este organismo (a cidade) que se desenvolve no tempo e no espaço, não é apenas moldado pela força das atividades económicas, mas sim motivado por estas. John e Julia Frazer (1998) afirmam que:

“(...) the economy is an ‘engine’, a constant ‘metabolic process’ of exchange of goods and financial capital which is regulated by, and enables the manifestation of the city’s internal ‘codescripts’ its culture, its politics, its social conditions, its physical form.” (p.12).

A economia no seu estado tanto macro (economia global) ou micro (economia da cidade) irá atuar como um catalisador do desenvolvimento da sociedade, assim o mercado, ou seja, a procura faz sobressair as prioridades de investimento (Frazer & Frazer, 1998, pp. 12-15).

O computador, através do seu desenvolvimento, demonstra a capacidade catalisadora da economia, quando comparamos os primeiros computadores com a atual tecnologia de informação que carregamos nos nossos bolsos. Os primeiros computadores surgiram nos anos quarenta e cinquenta, como máquinas que ocupavam grandes salas, com um elevado nível de manutenção, de difícil uso, com um grande consumo de energia e só acessíveis em laboratórios de investigação. Nos anos noventa, estas máquinas só marcavam presença em grandes empresas e indústrias capazes de suportar o seu custo. Com o avanço tecnológico impulsionado pelo interesse económico, passou a existir um acesso generalizado a estas máquinas e que na sua configuração mais “básica” habitam os nossos bolsos ou mochilas e são transportadas connosco, validando as “profecias” do passado (Mitchell & McCullough, 1995, pp. 3-5).

Perhaps I may clarify the historical background of the present situation if I say that the first industrial revolution, the revolution of the “dark satanic mills,” was the devaluation of the human arm by the competition of machinery. There is no rate of pay at which a United States pick-and-shovel laborer can live which is low enough to compete with the work of a steam shovel as an excavator. The modern industrial revolution is similarly to devalue the human brain, at least in its simpler and more routine decisions. Of course, just as the skilled carpenter, the skilled scientist and the skilled administrator may survive the second. However, taking the second revolution as accomplished, the average human being of mediocre attainments or less has nothing to sell that is worth anyone’s money to buy. (Norbert Wiener 1948 cited by Mitchell & McCullough, 1995)

A transformação do computador, de elemento exclusivo de um laboratório de investigação para um equipamento de conveniência e de apoio, alterou a visão sobre as qualidades procuradas pelo empreendedor. O impacto económico da presença da máquina de processamento de informação como uma ferramenta de trabalho, é vista como uma desvalorização sobre o valor do conhecimento e da inteligência humana. Esta visão defende que o desenvolvimento de *software* e *hardware* capaz de executar as tarefas do ser humano, leve à substituição deste, reduzindo assim o número de colaboradores numa equipa. Esta evolução das profissões é recorrente e visível desde os primórdios dos tempos. O desenvolvimento tecnológico no mundo do trabalho causa tanto a extinção de alguns trabalhos ou tarefas, como permite a evolução dos atuais e o surgimento de novas oportunidades laborais.

In the next 40 years architects will be faced with fabulous assignments. (...) Volume of work will be extremely heavy and probably quite varied, with architects and engineers called on to meet the fast-changing needs of technological advance. (“AIA Round-table Discussion”, 2014, p. 102)

A citação anterior demonstra como a arquitetura, uma disciplina que abrange as mais diversas temáticas, confronta constantemente os paradigmas contemporâneos. Ela que ao longo do tempo, é obrigada alterar os seus princípios de modo a fornecer uma resposta, para as preocupações que gradualmente ganham o interesse da sociedade. Salienta-se os problemas que envolvem a atual crise ambiental, proporcionada pela falta de gestão sobre os recursos ambientais, que se encontra em paralelo com o aumento da população demográfica mundial. Estas condicionantes afetam a economia mundial, que já se encontrava sob pressão no início do século XXI, a dificuldade em sustentar processos de design de longo prazo, obrigou a estabelecer metodologias práticas que possibilitem a otimização do trabalho, para além da capacidade de resposta interdisciplinar, sem perder a possibilidade de inovação. O computador acompanhado pelos sistemas CAD foram a resposta que surgiu do pós-guerra e que se veio a maturar no final do século XX e início do século XXI (Català, 2010, p.89; Architectural Record, 2014, p.102).

(...) digital computers have turned out to be specially appropriate for the following reasons: To generate design implies a somewhat industrial approach to production insofar as efficient automation is required to output large quantities of solutions. A programmable universal machine is certainly a very helpful tool in this respect. (Herr & Fisher, 2001)

A capacidade de resposta rápida pelo processamento de cálculos e símbolos demonstrada pelo computador, não basta para que esta seja uma ferramenta utilizada pelo arquiteto. O software desenvolvido para a arquitetura, tem de permitir uma tradução deste processamento de informação de um modo gráfico, ou seja a capacidade de visualização em tempo real do produto idealizado. Antes de existirem estas novas metodologias, o conceito e a forma do objeto arquitetónico era desenvolvido com o recurso ao papel, ao lápis e à borracha para a visualização 2D.

Designers are used to working in this mode; add marks and take them away, with conventions for relating marks together. (Woodbury, 2010, p. 11)

O papel nas suas diversas configurações consiste no suporte de visualização, o lápis é o mecanismo de adição do elemento que se pretende visualizar e por fim, a borracha integra a capacidade de subtração, total ou parcial, do resultante indesejado, proveniente do processo criativo. Mas apesar disto, o desenho à mão levantada continua a ser um desenho pouco preciso, sendo usado apenas como método de transmissão de abstrações, conceitos e idealizações. De modo a construir um desenho com carácter técnico, é necessário que o arquiteto use certas ferramentas neste meio. Estas ferramentas de apoio, dando como exemplo a régua, o esquadro, a antiga régua-T, o compasso e a régua de escala não passam de meios para desenvolver um desenho mais exato e preciso do processo mental do arquiteto (Woodbury, 2010, p.11). Em termos comparativos podemos afirmar que as ferramentas primárias do arquiteto (papel, lápis e borracha) são o seu “*tool-kit*” base de desenvolvimento de um projeto. No entanto, estas ferramentas por si só não bastam apesar do seu valor no processo de trabalho, o arquiteto necessita de se apropriar de ferramentas extra (*add-on*), neste caso a régua o esquadro e atualmente o computador, de modo a facilitar ou melhorar o seu trabalho.

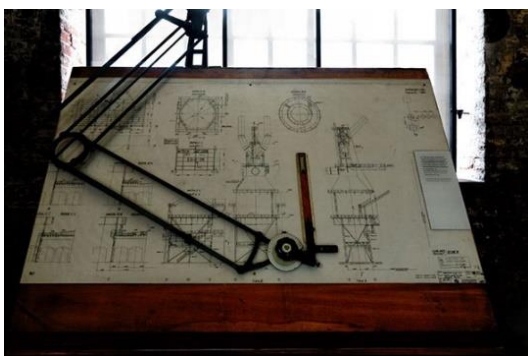


Fig. 4 | Processo convencional de desenho técnico. (howtorhino.com)

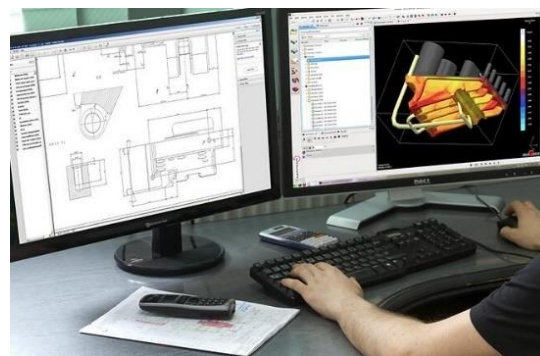


Fig. 5 | Processo atual de desenho técnico. (www.worthyhardware.com)

Deste modo, no final do século XX as metodologias de trabalho em arquitetura passaram gradualmente de uma arquitetura produzida de forma analógica para digital. Os *softwares* de arquitetura de uso mais comum, não passam de emuladores (*emulator*) dos processos de trabalho, descritos em cima, executados pelo arquiteto. O computador em comparação, por si só, integra o “starter-set” e os “add-on” numa só ferramenta que permite a produção e visualização do projeto (Woodbury, 2010, p. 11). É importante ressaltar que neste caso, o ato de “fazer arquitetura” não está a ser substituído pela máquina. A desvalorização e substituição está apenas presente na execução de um desenho técnico, que passa por processos constantes de desenho repetitivos, para obter um desenho preciso regido por regras exatas, e diretamente associado aos *softwares* CAD (*Computer Aided Design*).

In general practice, the acronym “CAD” more often than not translates as little more than Computer Aided Draughting, or, at best, the modelling of that which has already been designed. (Franzer & Rastogi, 1995, p. 8)

Este tipo de software teve e ainda tem um grande valor para o mercado e a indústria da arquitetura, engenharia e construção (AEC), pela sua capacidade de otimização e automatização das práticas (Fischer & M.Herr, 2001, pp.1-3). Foi nos anos sessenta que começaram a emergir os primeiros sistemas gráficos computacionais aplicados ao projeto. A intenção de imitar ou reproduzir as práticas do arquiteto para além de tentar prever as possibilidades de sistemas estruturais desenvolvidos pelos engenheiros, foram as premissas dos sistemas gráficos para projeto com o computador. Apesar de rapidamente se ter verificado que o desenvolvimento de software para a prática da arquitetura, não devia ser definido pela vontade de otimização e automatização como objetivo primário, estas características surgem do próprio uso do computador. Os problemas da arquitetura são variados e provêm de uma série de restrições, requisitos e intenções que dificultam a conceptualização de um programa que resolva tudo (Menges & Ahlquist, 2011, p.11).¹

Termed automated design, there was a holistic vision of computer programs which could process complex design briefs into specific architectural solutions. This alluded to the question of how a computer can, or even should, mimic human thought processes which form such knowledge. (Menges & Ahlquist, 2011, p. 11)

Na mesma altura que se marcava o desenvolvimento do computador aplicado à arquitetura, também surgiu a popularização da teoria da cibernética que permitiu desenvolver as noções da teoria de sistemas para a compreensão da nova relação entre homem-máquina. Uma relação estruturada sobre a premissa de expandir o conhecimento humano, em vez de aplicar um trabalho redundante sobre o processo que formou esse conhecimento (Menges & Ahlquist, 2011, p. 11). Fomentando o desenvolvimento de novos softwares. Patrick Hanratty em 1957, desenvolveu o DAC (*Design Automated*

¹ Esta era a visão e a dificuldade em 1960 e novamente expressa por Sean Ahlquist e Achim Menges em 2011, apesar de atualmente (2024) o problema, poder ser ultrapassado pelas tecnologias de geração à base de Inteligência Artificial.

by Computer), o primeiro sistema que envolvia interação gráfica. Sendo o primeiro sistema de *software* comercializado que se aproximava daquilo que chamamos hoje CAD/CAM pelo uso de programação de controle numérico numa ferramenta chamada PRONTO (Geddes, 2020).



Fig. 6 | DAC - Design Automated by Computer (1957). (time.graphics)

Apesar de Hanratty ser considerado o pai do CAD/CAM, foi apenas em 1963 que surgiu o primeiro *true computer-aided design software* intitulado de Sketchpad, desenvolvido por Ivan Sutherland na sua tese de doutoramento no MIT (*Massachusetts Institute of Technology*) (Woodbury, 2010, p.11). O produto final desta dissertação, o estudo inovador que proporcionou o desenvolvimento do “Sketchpad”, foi tão importante que, a mesma dissertação foi reimpressa em 1980 com o título “Sketchpad: A Man Machine Graphical Communication System.”, um título apropriado para a primeira verdadeira ferramenta de comunicação por sistema gráfico (Reese, 2014).

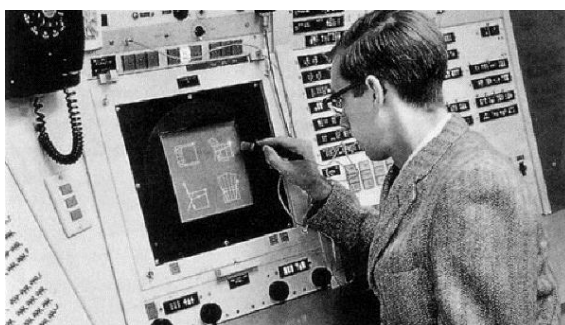


Fig. 7 | “Sketchpad: A Man Machine Graphical Communication System.”. (bimaplus.org)

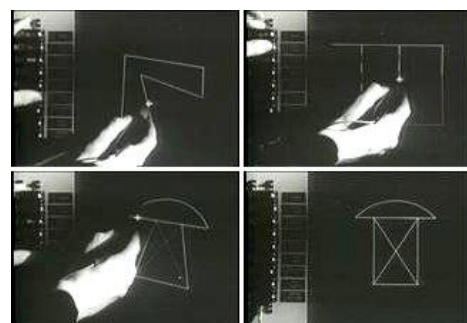


Fig. 8 | Utilização de Sketchpad. (en.wikipedia.org)

Construction of a drawing with Sketchpad is itself a model of the design process. The location of the points and lines of the drawing model and lines of the drawing model the design constraints which limit the values of design variables. (Ivan Sutherland, 1963, cited by Menges & Ahlquist, 2011, p. 12)

O Sketchpad foi a mudança da interação entre a comunicação de *user-machine* (operador-máquina). O operador usava uma *Light-pen* (o antepassado do atual rato) para interagir com o computador e manipular os objetos visíveis no ecrã. A *Light-pen* permitia desenhar no ecrã, e a interface de utilizador incorporava as ferramentas para a produção do trabalho (Reese, 2014). Aplicando a ideia de restrições que permitiam variar e manipular as interações entre as geometrias na formação de um sistema global. As geometrias eram criadas pela *Light pen* a partir de pontos que ao mesmo tempo criavam dependências entre elas (Menges & Ahlquist, 2011, p. 12). Robert Woodbury (2010) descreve o Sketchpad como o primeiro CAD que permitiu o design paramétrico:

(...) provided both a propagation-based mechanism and a simultaneous solver based on relaxation. It was the first report of a feature that became central to many constraint languages - the merge operator that combines two similar structures into a single structure governed by the union of all the constraints on its arguments. (p.11)

O trabalho de Ivan Sutherland introduziu a interface ao mesmo tempo que mantinha a criação de geometrias associada a parâmetros, permitindo a geração de sistemas. Este *software* serviu assim como um catalisador para o desenvolvimento de outras aplicações baseadas em sistemas gráficos como o GRASP (Generation of Random Access Site Plans) de Eric Teicholz desenvolvido em Harvard's Graduate School of Design. Este incorporava, para além da base de projeto de um espaço concebido, o desenvolvimento da forma como o resultado de interações externas pré-estabelecidas como estrutura, exposição solar e programática do projeto. O LOKAT, também desenvolvido em Harvard GSD, permitia a avaliação do processo evolutivo pela procura de um resultado específico, *goodness of fit* ou *fitness*, pela associação e proximidade dos parâmetros especificados com o resultado. Este progresso nas interações de sistemas foram cruciais para as definições de computação com métodos à base de parâmetros, que descrevem as regras base para sistemas generativos. Fornecendo respostas quanto à visão cibernética da máquina como um organismo e a sua relação entre homem-máquina, onde o processo de controle por *feedback* é o resultado da interação entre sistemas (Menges & Ahlquist, 2011, p. 12).

Os programas CAD apenas começaram a integrar os ateliers de arquitetura nos anos oitenta, apesar de no final dos anos sessenta já serem usados na área da engenharia, principalmente na indústria automóvel. Os softwares CAD juntaram-se aos sistemas CAM que já eram usados nos anos cinquenta pelo processo de *G-Code*² criado com o computador para o controlo das máquinas durante a manufaturização. A sua conjugação permitiu que as máquinas conseguissem ler os desenhos técnicos para produzir objetos físicos.

His invention was designed to aid the design and tooling of cars by integrating drawing machines, computer control, interactive free-form curves, surface design and 3D milling for manufacturing clay models and masters. (Geddes, 2020)

² G-code é a linguagem de programação mais comum para CNC (computer numerical control). Utilizada para indicar à máquina todos os procedimentos necessários para a produção de uma peça ou objeto. (TechTarget Contributor, 2017)

Pierre Bézier desenvolveu o UNISURF entre 1966-1968 enquanto trabalhava para a Renault. Um sistema permite criar superfícies 3D sobre software CAD/CAM. Posteriormente, o software CATIA introduzido em 1977, é uma aplicação multi-plataforma concebida para CAD, CAM e permitia a modelação 3D e avaliação do ciclo de vida dos produtos. Mais tarde o investimento passou a ser feito sobre a máquina e o acesso à mesma. Foi então que em 1981, a IBM desviou os seus investimentos para produzir o primeiro computador de secretária (*desktop computer*) direcionado para o consumidor comum, permitindo a disseminação de software e a base de sistemas CAD/CAM. Com o aumento do valor dos *software* CAD surge em 1982 a Autodesk, fundada por John Walker, disponibilizando no mercado, no mesmo ano, o AutoCAD para PC (Portable Computer). Em apenas três anos iria desenvolver a possibilidade de modelação 3D e em 1992 torna-se compatível com o sistema operativo Windows (Geddes, 2020).



Fig. 9 | IBM PC 5150, lançado em agosto 1981. (arstechnica.com)



Fig. 10 | Catia V1 utilizado no IBM station em 1982.

In the next 40 years architects will be faced with fabulous assignments. (...) Volume of work will be extremely heavy and probably quite varied, with architects and engineers called on to meet the fast-changing needs of technological advance. ("AIA Round-table Discussion", 2014, p. 102)

A citação anterior demonstra como a arquitetura, uma disciplina que abrange as mais diversas temáticas, confronta constantemente os paradigmas contemporâneos. Ela que ao longo do tempo, é obrigada alterar os seus princípios de modo a fornecer uma resposta, para as preocupações que gradualmente ganham o interesse da sociedade. Salienta-se os problemas que envolvem a atual crise ambiental, proporcionada pela falta de gestão sobre os recursos ambientais, que se encontra em paralelo com o aumento da população demográfica mundial. Estas condicionantes afetam a economia mundial, que já se encontrava sob pressão no início do século XXI, a dificuldade em sustentar processos de projeto de longo prazo, obrigou a estabelecer metodologias práticas que possibilitem a otimização do trabalho, para além da capacidade de resposta interdisciplinar, sem perder a possibilidade de inovação. O computador acompanhado pelos sistemas CAD foram a resposta que surgiu do pós-guerra e que se veio a maturar no final do século XX e início do século XXI (Català, 2010, p.89; Architectural Record, 2014, p.102). Deste modo, as metodologias de trabalho em arquitetura passaram gradualmente de uma base analógica para uma base digital, com a função de atingir uma produção rápida e automatizada onde de

modo geral, a falta de conhecimentos sobre *computational design thinking*, leva à falsa teoria de que o processo criativo proveniente do processo digital limita a criatividade do arquiteto. Esta arquitetura digital que se apresenta como aberta a novas possibilidades, acaba por se auto limitar no uso da ferramenta como apenas um visualizador de ideias (Frazer & Rastogi, 1998, p.9). De certo modo a básica implementação do programa, ou seja, o uso comum do computador vai influenciar o modo como este é percebido e o seu valor para o trabalho de projeto (Menges & Ahlquist, 2011, p. 11).

2.2 | Pensamento Computacional

(...) we are in the centre of a global, architectural renaissance: that we are in an exhilarating climate where the most advanced and challenging architecture being designed around the world could not have been conceived without the use of the computer. It is hard to deny such a proposition when looking at the changes in our surroundings. This seems to be occurring in a similar although not as meteoric way to the increased use of the car, which had dramatic effects on architecture and planning: the use of the computer affects every aspect of our lives and therefore our built environment. (Marcos Novak cited by Maggie Toy, 1998, p.7)

A arquitetura, como disciplina fundamental na sociedade, tanto no desenvolvimento do indivíduo, da cidade e da civilização, necessita de se adaptar a cada avanço tecnológico em vez de se fechar. Cabe ao arquiteto como representante da disciplina e da profissão, prestar atenção e marcar presença neste mesmo desenvolvimento. Tendo o dever de verificar o potencial destas novas ferramentas e participar no desenvolvimento de novas metodologias de trabalho, que se apliquem às necessidades da profissão. É fundamental compreender os princípios em que a “computação” se baseia, para que o uso da mesma no âmbito da Arquitetura e do Design seja benéfico. Em 2010 a arquiteta Ana Pla Catalá demonstrou o seu apoio nas novas tecnologias, tendo dedicado uma parte fundamental da sua carreira ao uso integral das mesmas no ambiente de atelier. A arquiteta apesar de usufruir e explorar o uso do computador na arquitetura afirmou em 2010 *“What is not clear is what the consequences of an integral use of computation will be for the discipline and the profession of architecture.”* (p.93). Apesar da incerteza, Ana Pla Catalá bem como outros investigadores do seu tempo consideram, no seu processo de trabalho, os conceitos que formalizam as bases da Teoria Computacional ao aplicarem o uso do computador nas diversas áreas de estudo. Apesar de nem sempre estar expressamente indicado, este processo de pensamento de “problem-solving” têm a capacidade de unificar, por fornecer um argumento sólido para a correlação dos fundamentos de conceitos base que caracterizam a biologia, matemática, ciência da computação, ciência dos sistemas e a filosofia, sem esquecer as visões contemporâneas permitindo a translação, neste caso, para uma aplicação no projeto de arquitetura (Menges & Ahlquist, 2011, p. 8).

We increasingly live in a world governed by complex computer and software systems, a world where programming is at the heart of industry and scientific discovery, as well as a basic skill for everyday life. (Rice University, [RICECS], 2022)

Como corroborado pela citação do Departamento de Ciência dos Computadores da Universidade de Rice (2022), a programação está na base do funcionamento destas tecnologias. Tendo em conta a ligação entre a programação e as novas tecnologias, quando é constatado o tema da aplicação e exploração do computador no ambiente da arquitetura, nem sempre esta relação é bem recebida. As entidades mais céticas ao tema afirmam que um dos principais desafios consiste na natureza técnica da

questão. Assim devido a uma pré-concepção, de que a estratégia para desenvolver, usufruir e integrar estas tecnologias no seu trabalho, necessitam de obter conhecimentos de *coding (script)* para além de capacidades de programação por parte dos Arquitetos e Designers (Menges & Ahlquist, 2011, pp. 8-11). A limitada noção dos fundamentos necessários para usufruir e manipular o computador no contexto da arquitetura, pode ter, origem nos próprios conceitos inerentes ao uso do mesmo, ou seja, “não integrem os métodos tradicionais do repertório e do ensino da disciplina.” (Menges & Ahlquist, 2011, p. 8).

Computer Science, Computational Thinking, Programming são três terminologias que aparecem recorrentemente quando se fala do computador. São conceitos diferentes que sofrem de uma errada interpretação de contexto, levando a que estes possam ser identificados como sendo o mesmo.

Computer science (Ciência dos Computadores) engloba os sistemas e processos de desenvolvimento, utilizando as tecnologias e sistemas computacionais, bem como metodologias e pensamentos, dos investigadores desta área. Abrange ainda temas que incluem as competências digitais necessárias para o uso da tecnologia, isto é: habilidades avançadas de programação tais como *scripting* e *coding*, indispensáveis para projetar estas tecnologias (Lcom Team, 2022). Esta ciência oferece ainda uma compreensão holística de todos os aspectos modernos, interconectados em relação aos computadores, bem como o conhecimento escalado de sistemas complexos, como por exemplo, um algoritmo para *machine learning*. Os profissionais formados nas componentes desta ciência aprendem a desenvolver processos à base de *Computational Thinking* (Pensamento Computacional) e noções de “*whole systems*”, ao adquirirem capacidades multidisciplinares que lhes permitem compreender e projetar, em diferentes escalas, sistemas computacionais complexos e participar no desenvolvimento de produtos inovadores (RICECS, 2022).

Em termos comparativos, *Programming* (Programação) é uma pequena parte inserida na ampla temática da *Computer Science*. A programação fragmenta-se na implementação de diversas linguagens de código, alguns exemplos mais conhecidos são Python, Java e HTML (RICECS, 2022).

In short, programming is the act of communicating instructions to different levels of computing devices (for example, hardware or software) so they can carry out tasks intended by the programmer (or coder). (RICECS, 2022)

Computational Thinking (Pensamento Computacional) engloba uma série de conhecimentos e metodologias com base em processos que permitem resolver e compreender problemas complexos (*problem-solving*). Esta teoria, apesar de ser uma das bases da Ciência da Computação, é aplicada nas mais variadas áreas, ou seja, não se classifica como exclusiva a pragmáticas do computador (Menges & Ahlquist, 2011, p. 8-11).

The emergence of this perspective has come at the confluence of different modes of thinking, in alignment with various scientific, technological and cultural advancements spanning several decades if not centuries. (Menges & Ahlquist, 2011, p. 11)

Sean Ahlquist e Achim Menges (2011) explicam que *Computational Thinking Theory* na sua génese é o acumular multidisciplinar de conceitos, métodos e teorias presentes nas mais variadas áreas de estudo (matemática, biologia, ciência de sistemas, filosofia, etc), permitindo transpor esses conhecimentos para as mais diversas disciplinas. Deste modo, esta capacidade multifacetada de aplicação de novos métodos e conceitos, permite estabelecer processos de *problem-solving* (pp. 8-11). Assim sendo, *Computational Thinking* aplicado a *Computer Science* permite desenvolver processos de pensamento computacional, de modo a resolver o problema existente (*problem-solving*). Este procedimento pode contribuir para o desenvolvimento do código. Deste modo, Coding é o processo que permite programar diferentes ferramentas digitais pelo uso de algoritmos. Neste caso os algoritmos servirão como *logic-based steps*, que irão comunicar com estas ferramentas tecnológicas, para que, como já referido anteriormente, executem as diversas ações planeadas pelo programador (Lcom Team, 2022).

Nesta perspectiva, enquadrando estas questões no âmbito da disciplina da Arquitectura, é possível afirmar que o principal desafio do arquitecto ou designer para usufruir do espectro de possibilidades, provenientes da integração do computador no seu processo de trabalho, não está apenas inerente à capacidade do arquitecto apresentar conhecimentos de programação (*coding, scripting*).

Yet, over many years of teaching computational design, we have realised that the main challenge does not lie in mastering computational design techniques, but rather in acculturating a mode of computational design thinking. (Menges & Ahlquist, 2011, p. 8)

Atualmente, a “computação” e a arquitetura, não podem ser mais consideradas como duas disciplinas completamente distintas sem qualquer tipo de relação. Apesar do conceito de “computação” ser considerado um tema multifacetado e intrincado, a sua integração no âmbito da arquitetura abre caminho para uma série de novas oportunidades, considerando no entanto, um fluxo contínuo de novos desafios (Menges & Ahlquist, 2011, p. 11).

The dominant mode of utilizing computers in architecture today is that of computerization; entities or processes that are already conceptualized in the designer's mind are entered, manipulated, or stored on a computer system. In contrast, computation or computing, as a computer-based design tool is generally limited. The problem with this situation is that designers do not take advantage of the computational power of the computer. (Kostas Terzidis, 2006, citado por Menges & Ahlquist, 2011, p. 10)

Como verificado anteriormente, a questão económica tem o seu peso na introdução das ferramentas, sendo extremamente fácil uma metodologia de trabalho cair na tendência da industrialização em vez do aspecto exploratório da máquina. O erro está no uso do computador para “computadorização” em vez de “computação” como pode ser verificado na citação anterior (Menges & Ahlquist, 2011, p.10). Esta importante diferenciação dos termos é recorrentemente salientada por diversos autores onde o que muda é a nomenclatura dada aos dois conceitos. Robert Woodbury (2010) marca a diferença identificando o processo Amador vs Profissional (p.9), enquanto que autores como Mateus, Roseta e Monteiro (2013) referem como “*catch the curve*” e “*create the curve*”. Como expressado

anteriormente, a teoria da computação engloba o processo de *problem-solving*, que aplicado ao uso do computador o resultado final vai ser proporcionado pelo processo lógico de pensamento do sujeito/operador da máquina.

Computadorização é o processo mais básico de pensamento e/ou processamento, que parte pela simples compilação de dados ou conjuntos de dados (*inputs*) que no final vai proporcionar tanta informação (*output*) como a que inicialmente já se tinha. Por sua vez, a computação vai deduzir resultados a partir de dados ou conjuntos de dados (*inputs*), o que permitirá um aumento da informação base e/ou a obtenção de informação específica diferente da inicial (Menges & Ahlquist, 2011, p. 10). O aumento do uso da computação na disciplina de arquitetura é maioritariamente para gerar geometrias complexas, imagens, ou uma visualização de ideias. Um uso deveras simples, tendo em conta o amplo espectro de possibilidades.

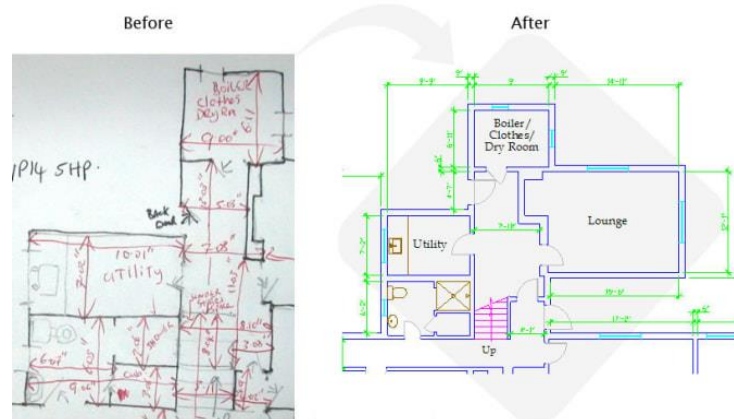


Fig. 11 | Evolução do desenho de projeto. (www.fiverr.com)

A computação permite desenvolver estratégias para que o modelo seja capaz de se interligar com a sua envolvente, permitindo a adaptação sobre o tempo e espaço, em vez da cópia com ajuste ao problema. Permitindo um trabalho ambicioso do computador sobre processos computacionais que vai além das intenções formais e compreensões meramente estéticas, que por vezes entram em conflito com o ecossistema no qual é inserido, tornando-se canceroso em vez de benéfico (Catalá, 2010, pp. 89-93). Em suma, a distinção passa pela abordagem ao processo de design, onde o primeiro termo passa por processo de computer-aided que assume uma estratégia de obtenção do objeto, por simplesmente agrupar a informação para uma representação simbólica (métodos de organização de informação) (Menges & Ahlquist, 2011, p.10).

Estes procedimentos já costumam estar integrados em *softwares* como ferramentas (*tools*) disponíveis na *interface*, mesmo que sejam geradas por métodos paramétricos não passam da obtenção de uma solução final, pelo uso de “tipos” uma parede, janela ou uma porta. Estes elementos proporcionados em *softwares* como o ArchiCAD ou o Revit (Mateus, Roseta e Monteiro, 2013). Em contraste, o método computacional permite o estudo do processo na obtenção de informação (*data*) da abstração inicial. A particularidade do *code* na metodologia de design contém os valores (*inputs*) e as ações (regras, parâmetros), que vão integrar e afetar o resultado final permitindo que se adapte ao ambiente e não que o ambiente se adapte ao resultado (Catalá, 2010, p. 90). O computador fornece ao

arquiteto a capacidade de processar informação em níveis que vão para lá dos limites humanos, exemplo de softwares que permitem este processo exploratório são o Grasshopper um plug-in do Rhino (Mateus, Roseta e Monteiro, 2013) ou o Dynamo da Autodesk.

People do design. Planning and implementing change in the world around us is one of the key things that make us human. Language is what we say, design and making is what we do. Computers are simply a new medium for this ancient enterprise. True, they are the first truly active medium. As general symbol processors, computers can present almost limitless kinds of tools. (Woodbury, 2010, p. 7)

Os computadores são máquinas de processamento de símbolos onde a programação é usada como uma linguagem de comunicação com a máquina existente em qualquer software. Semelhante à programação, o designer no uso do software decide o nível de abstração que irá implementar no uso do mesmo. O trabalho com base em alto nível de abstração permite a produção de um maior número de possibilidades, onde o computador é usado como um meio de interpretação dos símbolos, por mapear ou projetar, formando elementos que são atribuídos a um produto dessa metodologia de design, o “objeto” resultante é apenas uma das milhentas possibilidades dentro do seu sistema dinâmico. Enquanto o uso do computador como gerador de conjuntos de símbolos fica-se apenas por uma abordagem de obtenção de um objeto ou informação que nada acrescenta à imposição inicial - Projetar o já projetado. De qualquer modo, Thomas Fisher e Christiane M.Herr (2001) fornecem uma explicação sobre o tema *“The semiotic relationship between symbol production and symbol interpretation can be anywhere from strict, intentional and meaningful (rational generative design approach) to random (irrational generative design approach).”*

A fundamental transformation is taking place in both scientific culture and epistemological culture. New theories of nature that emphasise the multiple, the temporal and the complex, and technology that promotes connection as opposed to separation, are radically redefining our conception of space. (John Frazer & Mani Rastogi, 1998)

As bases e os aspectos de interesse na concepção de espaço sofreram mudanças significativas durante o pós-guerra do século XX. Esta evolução de um pensamento sobre a relação entre homem-edifício que por sua vez é transposto para o contexto do edifício na cidade (Frazer & Rastogi, 1998, p.8). Um processo contínuo de pensamento que analisa o que origina/necessário para um resultado específico, um foco sobre o *“know how”* para classificar, controlar, aceder, distribuir e direcionar a informação sobre o ambiente de ecossistemas complexos, em vez de uma mera representação. Rompendo com a visão que a ligação entre lógica e causa não são uma metodologia viável para a produção de modelos. Assim nos anos 60s diversas publicações nas mais variadas áreas de estudo começaram a focar as relações dos agentes externos e o impacto sobre os micro-sistemas que por sua vez é projetado no macro-sistema, rompendo com a visão de sistemas fechados (Kallipoliti, 2010, pp.15-23), que por sua vez aplicando ao contexto arquitetónico, um edifício não existe isolado da envolvente em que é inserido (*whole systems*).

Space and time are no longer separate, not even in an everyday sense: a spacetime vernacular has developed. (Marcos Novak, 1995)

Este entendimento de sistemas dinâmicos que se transformam pelo tempo e espaço, concede à arquitetura uma visão mais humanista, natural, de um modelo biologicamente evolutivo. Em contraste com as regras da arquitetura vernacular do passado, uma visão do edifício como um sistema estático, arquitetura de monumentos. Uma arquitetura proporcionada pela estrita racionalização matemática do espaço, levada aos limites pelos artistas frutos do seu tempo durante o século XIV e XV com bases nos entendimentos da arquitetura de Vitruvius (Carl, 2011, p.38-39).

Uma postura que se assume sobre a arquitetura como um organismo dinâmico encontra fundamentos na teoria da cibernética, que inicialmente viu o seu desenvolvimento expandido para compreensão da relação entre homem-máquina. Douglas Engelbart (1964) argumentava que a relação entre máquina e homem deveria ser interativa e responsiva em tempo real, para que a inteligência humana pudesse ser expandida. A tecnologia torna-se num meio de resposta aos complexos problemas da humanidade. Engelbart estende este pensamento de uma forma quase radical, sobre a intenção que homem e máquina sejam um só permitindo uma co-evolução, ligando o cérebro humano ao computador. No entanto o pensamento de co-evolução em contraste com o único propósito de usufruto por uma das partes, foi a visão dos anos 60s, trabalhos de referência como “Man-Computer Symbiosis” desenvolvido pelo psicólogo JCR Licklider abordam este mesmo tema. A cibernética do pós-guerra aplicada à tecnologia como uma afirmação da máquina como um organismo e os organismos como máquinas, onde o organismo humano é estendido pela tecnologia numa ideia de human-aided machines (Wigley, 2010, pp. 53-54).

Por sua vez, investigadores dos anos 60s como Gordon Pask aplicam a teoria da cibernética na arquitetura, corroborando a ideia de uma arquitetura como um sistema que integra e gera sistemas, uma arquitetura que oferece respostas às mudanças do tempo. O seu trabalho providenciou modelos para um design interativo aplicável tanto a ambientes virtuais (digital) como reais (físico). (Hunt, 1995, p.53)

... the computer, material and all, engages (the user-participant) in dialogue and within quite wide limits is able to learn about and adapt to his behaviour. (Gordon Pask, 1969, p.495)

O entendimento da cibernética e o seu envolvimento com a computação e por sua vez com a arquitetura, permite uma compreensão teórica sobre o funcionamento dos sistemas e como estes devem ser abordados e trabalhados. No entanto, para o desenvolvimento do projeto, aplicado à arquitetura, tirar partido da computação é necessariamente uma atitude racional. O arquiteto, à semelhança do programador, tem de adotar diferentes níveis de abstração para permitir uma visão mais ampla do problema, obtendo várias soluções em vez de uma única solução. O algoritmo, uma componente importante da computação, com origem na matemática, é o meio para definir o processo de resolução do projeto numa lógica de passo a passo, no processo de tomada de decisão (Mateus, Roseta e Monteiro, 2013). Sean Ahlquist e Achim Menges (2011) definem algoritmo “... as a set of procedures consisting of a finite number of rules, which define a succession of operations for the solution of a given problem.” (p.11).

Mateus, Roseta e Monteiro (2013) defendem que estas metodologias de trabalho permitem ao arquiteto expandir as suas capacidades de abordagem ao problema, culminando em novas formas de

conceção arquitetônica. O seu uso não é um obstáculo à criatividade do arquiteto, pelo contrário, fornece um entendimento mais controlado sobre o seu design, pois para que estes possam ser corretamente aplicados, o arquiteto terá de ter em consideração um maior número de variáveis ao projeto. Aqui entra a vantagem do computador que compensa as limitações humanas em calcular todas estas variáveis que influenciam o projeto. O Sketchpad foi o primeiro exemplo de desenvolvimento tecnológico sobre uma lógica de inter-relações entre geometrias num sistema gráfico, onde a associação causa um efeito em cascata que irá desenvolver e manipular a forma. Um avanço que atribui ao design a capacidade de pensar na forma como um acúmulo de forças, requisitos e restrições provenientes de agentes externos que irão influenciar a forma, quebrando com o foco do design de objetos (Menges & Ahlquist, 2011, p. 13).

Deste modo a computação aplicada ao projeto deve ser introduzida como um processo de pensamento. Os autores Sean Ahlquist e Achim Menges (2011) fornecem uma definição clara da computação aplicada ao projeto:

In relation to design, computation is the processing of information and interactions between elements which constitute a specific environment, the pivotal word being interaction. This definition pulls computation out of only the virtual realm. In a greater relevance to design and the designer, the most general application of computation is in producing outcomes released from the processing of internal and external properties. Computation provides a framework for negotiating and influencing the interrelation of these datasets of information, with the capacity to generate complex order, form and structure. Computation as a design methodology is to formulate the specific. Where computer-aided processes begin with the specific and end with the object, computational processes start with elemental properties and generative rules to end with information which derives form as a dynamic system.

No entanto, a disciplina de arquitetura necessita de assentar bases que integrem as novas tecnologias no ensino da mesma, acabando com exclusividade dos arquitetos autodidatas onde as suas capacidades são provenientes da vontade de aprender. Tornando a produção de arquitetura com o uso do computador um ato responsável e fundamentado. O arquiteto assim é apresentado a uma série de novas possibilidades interdisciplinares, mas com grande complexidade para serem facilmente abordadas. É necessário entender como trabalhar corretamente com elas, para que o seu uso seja benéfico. Em que o uso das ferramentas com total liberdade, mas sem uma base que fundamente o processo e o resultado, poderá originar uma escultura feita com apoio do computador, mas que não resolve o problema arquitetónico. Por sua vez, restringir o uso do computador e impedir a sua utilização na procura da forma no processo do projeto da arquitetura, resulta no uso simplificado deste, abordando apenas processos de computadorização. O computador passa a ser usado exclusivamente como um método de visualização ou em processos orientados à produção de objetos. O arquiteto precisa de encontrar um meio termo, em que o uso do computador no processo de projeto, seja tão intuitivo como o uso do lápis no desenhar ou o uso do cartão na maquete, para desenvolver uma ideia, resolver um problema e conceber um projeto de arquitetura (Woodbury, 2010, passim).

Radical architecture pedagogies aimed to challenge the status quo by attempting to destabilise the very institutions they depended on, and in so doing they generated forms of institutional critique. (Colomina, Choi, Ignacio et al, 2012, p. 79)

Os *softwares* com capacidade de aplicação ao projeto arquitetônico são fruto, como anteriormente estabelecido, do desenvolvimento tecnológico que abre portas para novas possibilidades. Como referido anteriormente, os *softwares* começaram por ser emuladores de processos analógicos, mas atualmente as suas características conseguem ser estendidas para lá dessa primeira premissa. O que é facto, é que, nem todos os arquitetos que terminam os seus estudos no ensino superior (2024), detém tais capacidades e conhecimentos no seu curriculum. Maggie Toy, em 1995, afirmou que os professores precisavam de bases concretas, para poderem ensinar e entender as possíveis aplicabilidades do computador na profissão e no meio do ensino da arquitetura. Onde a dificuldade parte pelo facto do avanço tecnológico não estabilizar, devido ao constante desenvolvimento e inovação tecnológica. Apesar da dificuldade, demonstra a importância do ensino da arquitetura incorporar o computador, referenciando trabalhos de alunos de arquitetura da Bartlett School of Architecture, UCL, como prova do potencial e demonstrando que existe solução para o problema (p.7).

Very little has been done so far to develop methodologies, materials and curricula for generative design teaching and to clarify terms and techniques for teaching purposes. (Fisher, 2001)

Sean Ahlquist e Achim Menges (2011) afirmam que existe falta de integração de conhecimentos técnicos no repertório de ensino da disciplina de arquitetura (pp.8-9). Estes autores não são os únicos a afirmar tal falha no programa educativo, em que a devida exploração e aquisição de conhecimentos referentes ao tema durante a formação dos jovens arquitetos não é formalizada. Thomas Fisher e Christiane M.Herr (2001) reconhecem que a recorrente produção de trabalho com apoio do computador, tem levado a uma maior procura por parte dos jovens arquitetos em adquirir mais conhecimentos teóricos e principalmente práticos para utilizarem tais ferramentas na sua área de trabalho. É importante desenvolver um programa curricular no ensino superior que integre metodologias que incluam a compreensão da componente teórica, sem desvalorizar a importância da componente prática. A mudança deve começar por educar os futuros arquitetos sobre como usar essas ferramentas de uma forma responsável, criativa e inovadora, para que a arquitetura resultante do uso dos mesmos seja exploratória e não industrializada. É preciso acabar com a ideia de uma *"Talkitecture"*, um termo que Woodbury define como - *"...reserved for those who discuss but do not draw. Don't draw it onto yourself."* (2010, p.9). O arquiteto precisa de saber como funciona o aplicar e praticar as qualidades da implementação do pensamento computacional. *"Ideas need tempering. There is nothing like a room of 200 practical people to tell you when your ideas and explanations don't work."* (Woodbury, 2010, p.3-4).

Today, genetic algorithm, cellular automaton or shape grammar are very important and very common keywords in international discourses on CA(A)D but due to their relative novelty in design and their complexity, these approaches are largely neglected in

undergraduate design studies. This is not only due to the interdisciplinary involvement of generative design work. (Fisher & M.Herr, 2001)

Muitas das bases e conhecimentos já são abordados na disciplina, como é o caso da matemática, em que o ensino filtrado para uma aplicação prática em CAAD (*Computer-aided architectural design*), permitirá um avanço para a resolução do problema.

The niche of exploration that opens up ahead of us as architects lies in the mathematics of relationships between ecology and computation as a means to describe a disciplinary shift from the study of objects to the study of processes. (Catalá, 2010, p. 90)

A natureza, uma musa do artista, uma fonte de inspiração do arquiteto desde os primórdios da arte. O arquiteto que recorrentemente recorre a esta fonte de inspiração por conjugação com a forma, pode agora pela integração da computação no seu currículo, estender o seu âmbito de possibilidades e aplicabilidade. A computação que reproduz no seu funcionamento diversas teorias e estudos da natureza, permite ao arquiteto aplicar teorias de evolução pelo uso de algoritmos para identificar e produzir as soluções mais “fortes”, adaptáveis.

Many different search algorithms exist, but evolutionary searches are particularly relevant as the algorithms computationally evolve solutions without explicitly specifying the direct evolution process, which becomes an emergent property of the algorithm. (Menges & Ahlquist, 2011, pp.20-21)

Deste modo a arquitetura pode co-evoluir com o design computacional, mas para que tal aconteça, é importante deixarmos de rejeitar a evolução tecnológica e a integração desta na prática. Um aspecto da arquitetura que se tem vindo a verificar com regularidade. Já nos anos sessenta até aos anos oitenta, os próprios estudantes exigiam mais das suas instituições, culpando as mesmas por não interligarem as necessidades do tempo com o currículo educativo. *“Radical Architecture pedagogies”* é uma pesquisa colaborativa que decorre há vários anos sobre o intuito de analisar as experiências revolucionárias que ocorreram nos anos sessenta e setenta. Os alunos do doutoramento da faculdade de arquitetura da Universidade de Princeton é que realizam as experiências, registando os resultados das mesmas para uma exposição sobre o tema. A revista *The Architectural Review*, em 2012, aborda a investigação para salientar o problema que não é de agora, mas que decorre desde os anos sessenta (Colomina, Choi, Ignacio et al, 2012, p. 79).

Architectural pedagogy has become stale. Schools spin the old wheels as if something is happening but so little is going on. Students wait for a sense of activist engagement with a rapidly evolving world but graduate before it happens. (Colomina, Choi, Ignacio et al, 2012, p. 79)

2.2.1 | Processo Algorítmico

The manifest form - that which appears - is the result of a computational interaction between internal rules and external (morphogenetic) pressures that, themselves, originate in other adjacent forms (ecology). The (pre-concrete) internal rules comprise, in their activity, an embedded form, what is today clearly understood and described by the term algorithm. (Sanford Kwinter cited by Menges & Ahlquist, 2011, p.10)

O processo de informação algorítmica é uma componente importante do trabalho e pensamento computacional, imprescindível na aplicação e execução de linguagens de programação (Woodbury, 2010, pp.34-35). Este processo proveniente da matemática, consiste numa sequência finita e sucessiva de instruções elementares; ações executáveis descritas de um modo geral, mas exacto, com o propósito de obter uma solução para um problema. Deste modo, o processo de informação algorítmica integra a computação por via da configuração técnica, com fundamento teórico para uma *framework*, onde os sistemas virtuais (digitais) e físicos (o mundo real) são identificados como o acumular de conjuntos de informação, desenvolvidos sobre uma metodologia própria. A ideia de algoritmo remota ao trabalho do matemático norueguês Axel Thue em 1914. Thue inventou o *string rewriting system* (SRS), um programa que reescreve cadeias de sinais. Mais tarde, foi demonstrado que o problema de Thue era recursivamente sem solução, mas o seu processo de decisão continha as características bases para a computação algorítmica (Menges & Ahlquist, 2011, p.11).

An algorithm is

a finite procedure

written in a fixed symbolic vocabulary,

governed by precise instructions,

moving in discrete steps, 1,2,3,...

whose execution requires no insight, cleverness,

intuition, intelligence or perspicuity,

and that, sooner or later, comes to an end.

(Berlinski, 1999, cited by Woodbury, 2010)

Para entender melhor, pode-se criar um paralelismo entre o algoritmo e uma simples receita de culinária. Tal como o algoritmo, uma receita consiste na sequência de passos explícita, que culminam num resultado final, neste caso o prato confeccionado. A receita é escrita num idioma específico, com uma fonte específica e descreve as quantidades e medidas usadas passo a passo, para obter o resultado final. Assim, transpondo para a programação de um software ou programa para o computador, este apoia-se numa linguagem de programação, que indica o passo a passo, para a execução das ações pretendidas, que

culminam no resultado final. O pensamento algorítmico é implementado como uma lógica em esboço dos vários passos numa sequência específica, possibilitando diferentes níveis de abstração dos pormenores (Bazilio, 2023). A capacidade de abstração no processo de pensamento computacional e por sua vez no uso do computador, permite compreender o quão adaptável é um processo ou solução. A utilidade prática do pensamento computacional está, de um modo geral, ligada com a ideia de que quanto maior a diversidade e oportunidade de aplicação, maior é o seu valor (Woodbury, 2010, p.30).

O significado da palavra “abstração” é vasto e variado, tendo como referência, o contexto onde é inserida, sendo possível afirmar que a própria palavra permite um certo nível de abstração, ao se adaptar a diferentes usos. A abstração³, que pode ser vista como o ato de descrever um conceito geral, ao invés de fornecer um exemplo específico e detalhado, é geralmente associada a algo vago, indeterminado ou indefinido. Robert Woodbury (2010) compara o significado da palavra “abstração” entre o meio artístico e o mundo da ciência dos computadores. Afirmando que, no meio artístico, nomeadamente no projeto, *“abstract ideas are often protein, that is, they are used as a base from which to generate many alternatives.”* (p.30). Deste modo, o meio artístico beneficia das duas conotações, em que a abstração surge como um conceito geral sem uma realização previamente determinada, permitindo diversos processos de resolução, e por outro lado, também permite que um conceito vago gere diversas interpretações, em que cada uma poderá ter o seu próprio processo.

No entanto, a ciência do computador encara a abstração na sua primeira conotação anteriormente referida: *“an abstraction describes of a class of instances, leaving out inessential detail.”* (Woodbury, 2010, p.30). A Universidade de Rice (2022) afirma que os diferentes níveis de complexidade da linguagem de programação estão inerentes ao nível de abstração do código:

- *High-level (or high abstraction) programming languages are closest to the software and user interface layer*
- *Low-level (or low abstraction) programming languages are closest to the binary code (1's and 0's) understood by machines (RCIS, 2022)*

Assim, a ciência da computação no emprego da programação, procura constantemente formalizar o seu código, na íntegra ou parte, para ser adaptado a diversas situações. Existe um foco em ter o código organizado e claro, de forma a eliminar o trabalho de fazer do zero os passos anteriormente já criados, em suma eliminar a repetitividade. Deste modo, os programadores apoiam-se na abstração, para tornar o seu trabalho reutilizável em vez de específico.

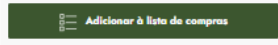
Uma receita de culinária também é escrita com apoio na abstração com a mesma conotação que é usada na ciência dos computadores. Não existe a necessidade de ter os passos extremamente detalhados, a própria omissão (abstração) das informações previamente assumidas, ou sem relevância para o resultado final, é que lhe conferem uma clara generalidade.

³ O site infopédia (Dicionário Porto Editora) define numa visão filosófica como - “separação mental de um ou mais elementos concretos de uma entidade complexa (facto, representação) desprezando outros que lhe são inerentes; conceptualização”, “ato ou efeito de abstrair ou abstrair-se” (2023).

Ingredientes

espaguete	400 g
alho	2 dente
talo de aipo	10 g
cebola	1 unid.
azeite	1 c. de sopa
carne de novilho picada	300 g
sal	½ c. de chá
tomate pelado em cubos	200 g
polpa de tomate	100 g
tomilho seco	1 c. de sobremesa
queijo parmesão ralado	2 c. de sopa

* Considera preço médio e indicativo dos artigos



Preparação

Passo 1

Num tacho, coza o espaguete em água a ferver durante 8 minutos.

Passo 2

Retire do lume, escorra e reserve.

Passo 3

Pique o alho, o talo de aipo e a cebola.

Passo 4

Numa frigideira antiaderente, refogue o alho, o aipo e a cebola no azeite.

Passo 5

Junte a carne de novilho picada, o sal e deixe cozinhar em lume médio.

Passo 6

Acrescente o tomate pelado aos cubos, a polpa de tomate e o tomilho seco.

Passo 7

Misture bem e deixe cozinhar mais um pouco para apurar.

Passo 8

Sirva o espaguete coberto com o molho e polvilhe com queijo parmesão ralado.

Fig. 12 | Receita de bolonhesa. (www.pingodoce.pt)

Tendo como exemplo a receita de espaguete à bolonhesa presente na figura 12 verificamos que, no Passo 1, apesar de claro, a omissão de informação como quantidade de água, dimensão do tacho e intensidade do lume, permite uma redução de etapas e condensação da informação ao estritamente necessário, para a obtenção do resultado final (Bazilio, 2023). É importante deixar claro que apesar dos passos serem resumidos, o “esboço” tem os detalhes, especificações pré-assumidos, em que o espaguete para cozer tem de ter água a ferver previamente, logo o lume tem de estar alto e a quantidade de água necessária é proporcional à dimensão do tacho disponível. O próprio Passo 1, por conter um certo nível de abstração, consegue ser adaptado a uma receita diferente, como por exemplo: Num tacho, coza as batatas em água a ferver durante 8 minutos. Existe assim um benefício claro no uso da abstração para uma melhor compreensão e uso do processo integral ou parcial, sem perder a informação do projeto anterior.

With the recent attunement to information as the basis of continuity, it seems that type has been inscribed in the effort to bring reality to a single horizon of representation, in which, ideally, all relations are explicit, even calculable (as in, for example, a parametric field). That is, “space” as a field has given way to a type of field comprising entities that obey mathematical or logical (algorithmic) operations. (Carl, 2011, p.39)

Os arquitetos abordam o significado da palavra abstração tal como a ciência dos computadores, quando se referem a modelos dimensionais de referência (modular de Le Corbusier), detalhes e formas padrão, incluindo o uso da linha central de referência num projeto de estruturas. Estes conseguem ser aplicados em diversas situações, por dependerem de um número finito de dados/informações prévias (*inputs*), que não se correlacionam (obrigatoriamente) com a aplicação anterior em pormenores específicos (Woodbury, 2010, p.30).

Dwelling, properly understood, is more profound than the efficient or attractive accommodation of a lifestyle - it comprises orientation in reality. (Carl, 2011, p.40)

A arquitetura aborda o uso de elementos tipo de diferentes maneiras no pensamento conceptual, partindo do isolamento de semelhanças e organizando em categorias, como um conjunto de preconceitos aptos a uma manipulação e aproximação ao mundo real. Esta visão e associação sintetizada de tipos, engloba o conhecimento teórico e prático sobre dimensões, sistemas construtivos ou espaços. A forma como o arquiteto aborda a abstração permitirá mais ou menos a exploração do espaço. Um quarto é por si só uma componente da habitação, por sua vez é um tipo. O arquiteto tende a abordar o quarto associando-o a um tipo específico, um pré-conceito que determina um espaço. Ele visualiza um espaço com uma dimensão generalizada ao seu contexto, ou seja, integrando no seu interior elementos específicos como uma cama, mesas de cabeceira, um guarda-roupa, janelas e um acesso a uma casa de banho integrada ou não. Enquanto que, uma visão com maior teor de abstração, interpreta as vivências e situações que acontecem no espaço, proporcionadas pela interação do ser humano (força externa) com o espaço. Esta forma de pensamento, permite uma maior liberdade na exploração do desenho arquitetónico (Carl, 2011, p.39).

Diversos autores demonstram que existe alguma dificuldade por parte dos arquitetos, em aprender e beneficiar do pensamento algorítmico. Robert Woodbury (2010) salienta duas características para essa mesma dificuldade de integração do pensamento algorítmico na conceção do projeto de arquitetura, o “procedimento” e a “precisão”.

The first is “procedure”: an algorithm is a process that must be specified step-by-step. Designers largely describe objects rather than processes. The second is “precise”: one misplaced character means that an algorithm likely will not work. In contrast, designerly representations are replete with imprecision - they rely on human readers to interpret marks appropriately. (Woodbury, 2010, p.35)

Robert Woodbury (2010) usa a citação acima transcrita, como justificação para a dificuldade dos designers em aprender pensamento algorítmico, uma realidade que verificou após 30 anos a ensinar programação em escolas de design. Sendo que existe uma ligação entre o pensamento algorítmico e a programação, Woodbury (2010) sugere que um designer de modo a facilmente ultrapassar essa dificuldade e aprimorar esta metodologia de pensamento, precisa de aprender uma linguagem de programação (p.49).

In almost all of these, to use the language you must remove yourself from the actual task and your accustomed visual, interactive representation. You must work in a domain of textural instructions. (Woodbury, 2010, p.35)

O pensamento algorítmico na sua génese difere dos outros procedimentos de pensamento. Existe uma clara diferença entre a representação de objetos familiares ao design e o processo sequencial necessário para o uso do algoritmo, esta lacuna dificulta a aprendizagem. Passa a ser exigido ao designer uma abstração dos ensinamentos base, que prescinde da componente visual, para tirar partido do algoritmo. Deste modo, é importante reafirmar que um designer ou arquiteto não precisa de ser um programador, nem aprender programação, para poder usar estas ferramentas. Mas existem certos conhecimentos ou noções base, do qual o arquiteto pode beneficiar para expandir os seus horizontes.

It is a way of thinking that some designers may find alien, but the first requirement is an attitude of mind that seeks to express and explore relationships. (...) However, at the highest level of fluency we may yet see a generation emerge who can "sketch with code". (Whitehead, 2010)

Existe uma lógica por detrás destes novos processos que permitem desenvolver novas estratégias de trabalho. A capacidade de desenhar é constantemente relacionada com a qualidade do trabalho e o valor do arquiteto, uma noção básica da disciplina. A aptidão para o desenho ainda hoje é bastante valorizada na arquitetura, permitindo expressar e desenvolver ideias, para além do próprio ato de desenhar ser um processo de pensamento para resolução de problemas (Virilio, 1995, p.61). O desenho é a principal ferramenta do arquiteto, a sua versatilidade permite diversas formas de comunicação, quer seja executada com mais liberdade (*sketch*) ou menos liberdade (desenho técnico). O desenho é um ato com dupla capacidade. A primeira é a capacidade de visualização, que por sua vez atua como um meio de comunicação, um vocabulário do qual o arquiteto tira proveito para transmitir uma ideia, um conceito, um sistema, um projeto. A segunda corresponde ao processo do desenho, como um trabalho mental na conceção de um projeto (Woodbury, 2010, p.36). O desenvolvimento tecnológico permite uma nova expressão do desenho, esta ferramenta imprescindível para qualquer arquiteto ganha novas propriedades de conexão e transmissão do projeto de arquitetura. A existência e o desenvolvimento de softwares de modelação, permite que o arquiteto acrescente mais qualidades ao desenho.

What old-timers like Buxton and I might see as overly determined and graphically definite, the new generation might well add the word dynamic to Buxton's list. (Woodbury, 2010, p.36)

Softwares como por exemplo o Grasshopper com bases em modelação paramétrica, permitem alterações rápidas ao modelo, facilitando a testagem de diferentes abordagens e por sua vez a redução do vínculo do arquiteto à primeira ideia (Mateus, Roseta e Monteiro, 2013). Estes modelos gerados, requerem tempo e conhecimentos base, para serem executados com qualidade e abstração - no contexto da ciência dos computadores (Herr & Fisher, 2001). Deste modo, os *system developers* apontam para o

desenvolvimento de software user-friendly, que permita ao arquiteto desbloquear processos de projeto rápidos, servindo como uma ferramenta de desenho. Cabe aos arquitetos e designers participar no desenvolvimento das ferramentas de computação para que estas correspondam às suas necessidades. Para que tal seja possível, é preciso uma clara transmissão dessas mesmas necessidades aos técnicos que irão desenvolver essas ferramentas, os programadores, *system developers*, *software developers* e designers de computação (Burry, 2011, p.118). Deste modo, o arquiteto não precisa de ser um programador para usufruir do computador e fazer projeto, curiosamente a mesma analogia pode ser estendida ao cientista de computador, que não se afirma como um programador, inclusive nem todos os cientistas de computador são dotados de grandes capacidades de programação (RICECS, 2022).

Like drawing in design, programming is the craft skill by which much of computer science is done. (Woodbury, 2010, p.49)

Assim é necessário entender o papel da programação no contexto da arquitetura. A programação corresponde à implementação da linguagem entre homem-máquina e máquina-máquina. *“Algorithms are realized as programs, which in turn are written in precise and prescribed programming languages.”* (Woodbury, 2010, p.49). O termo “linguagem de programação” demonstra a ligação ao mecanismo de comunicação. A capacidade de dialogar, que provém de uma aprendizagem de comunicação a partir de uma língua natural (português, inglês, espanhol), apresenta diversos níveis de experiência (Woodbury, 2010, p.49). Consequentemente, o método de comunicação escrito, é executado pela junção de letras para formar palavras, que por sua vez, sobre regras de construção frásica, resultam na expressão da língua como um sistema estruturado de comunicação, que consiste em gramática e vocabulário (Woodbury, 2010, p.49).

... without a concrete language, there is no formal language - no logic, no mathematics, no geometry, no “form”, and certainly no capacity to use these analogically (for example, to convert any of it into “architecture”). (Carl, 2011, pp.40-41)

Ser fluente numa dada língua, é resultado da capacidade de comunicar de forma correta naturalmente, uma capacidade de comunicação fora da sua língua nativa, o mesmo é associado a programação, onde o melhor método de aprendizagem é o trabalho intensivo e imersivo de uma linguagem de programação, que exclui outras formas de pensamentos (Woodbury, 2010, p.49).

... we may ask about the nature of the dialogue between concrete richness and formal complexity (that is, between a designer or user and form or code, which is the most common manifestation. (Carl, 2011, p.41)

O código é implementado sobre uma estratégia pré-definida, apesar de abrir portas às mais diversas possibilidades, ele rege-se sobre regras concretas, que definem o funcionamento do *software* ou programa (Catalá, 2010, p. 93). Robert Woodbury define que, um perito na área, é alguém que

compreende o conceito da abstração algorítmica e demonstra na execução prática, quase intuitivamente, as suas qualidades de programação. *“The expert wants a concept explained in general terms.”* (p.50). O pensamento computacional não é instruído com a aprendizagem da programação, existem então, diferenças entre o cientista de computadores e um programador, que refletem uma lacuna entre a compreensão e a aptidão de técnicas à base de habilidades (*skill-based*). (Woodbury, 2010, pp.49-50)

I reach out for it, touching its compact form ever so lightly to wake up my monitor. Yet I never really see the mouse. Even when reaching for it, my eye is already on the screen, drawn towards the imminent glow and heading immediately into the image that appears, pulling the rest of my body into the chair. (Wigley, 2010, p.50)

Wigley (2010) através da citação anterior, transmite como rato do computador que em tempos era de madeira e funcionava com eletrónica rudimentar, hoje é uma ferramenta complexa carregada de apetrechos, que vão para lá da sua configuração e funcionalidades básicas, de modo a ser adaptado às necessidades do usuário. Este dispositivo tão presente no mundo atual, permite ao homem o uso intuitivo e natural do computador. Pelos movimentos da mão sobre o rato, este transmite as ordens do usuário, que navega e explora o computador. Uma peça, que atua como uma prótese, que permite aproximar o homem da máquina. O rato deste modo, afirma-se como um exemplo da evolução do design de *hardware*, que atualmente permite ao arquiteto apropriar-se do computador como uma ferramenta de apoio ao seu trabalho, um facilitador de tarefas. Este dispositivo por sua vez, também evoluiu para novos métodos de exploração do processo inerente ao desenho, através desta mesma evolução do hardware. Uma transição da interface visual do papel para o ecrã, como por exemplo através de um Tablet + Caneta digital, que permitem ao ilustrador, arquiteto, designer usufruir de um desenho com maior eficiência, enquanto mantêm o valor do desenho gestual, à mão levantada. Uma forma de desenho que valoriza a qualidade e a habilidade de quem o pratica (Wigley, 2010, *passim*).

O projeto de arquitetura e o design de computação têm diversos valores em comum, desde a preocupação com a funcionalidade, até ao impacto inerente ao uso do mesmo. O trabalho é regido sobre diretrizes que refletem sobre o desejado e o necessário (*needs and wants*) do usuário ou cliente. O aumento da complexidade do trabalho do arquiteto, proporcionou a procura de ferramentas e metodologias que no início, requerem uma maior destreza na sua utilização, mas com o tempo, o seu uso se torna mais simplificado e intuitivo. Estes desafios são enfrentados pelas mais diversas profissões e disciplinas, partindo da necessidade de fazer mais com menos. O facilitismo é validado pela interação do usuário (*user*) com a *interface* gráfica, quanto mais intuitiva for a navegação mais fácil é o seu uso (Woodbury, 2010, p.65).

The approach for computational design is one which focuses on the execution of variational methods for the purposeful intent of resolving the complexities that exist in the interrelation and interdependencies of material structures and dynamics environments. (Menges & Ahlquist, 2011, p. 13)

O entendimento dos processos para a obtenção de um resultado específico ou que permita diversas respostas, é por sua vez, a base para o desenvolvimento de *software*, incluindo *software* CAD. Os

programadores ou *software developers* especializados em programação *End-user*, necessitam de se afastar da sua área de experiência, para desenvolverem um software que vá de encontro ao público-alvo. Eles precisam de compreender e clarificar o domínio da aplicação para que um processo possa ser abstraído e generalizado a diferentes métodos de aplicação (Woodbury, 2010, p.66).

In developing computer programs one is forced to question how people think and how design evolves. In other words, computers must be acknowledged not only as machines for imitating what is understood, but also as vehicles for exploring what is not understood. (Kostas Terzidis, 1997)

Kostas Terzidis (1997) transmite duas ideias base sobre o desenvolvimento de *Software*, onde o foco deve integrar tanto um pensamento de computadorização, em que as informações pré-existentes têm de ser armazenadas para poderem ser manipuladas, para além do fácil acesso as bibliotecas de informação (*data-center*), sem esquecer a procura de facilitismo na obtenção e construção da informação pré-estabelecida. Para além de capacidades para uma abordagem computacional, que permite ao usuário desenvolver os seus próprios processos e personalizações, na resolução dos seus problemas. Estes procedimentos irão no final culminar na interface apresentada ao user como “*furnish information*” (Menges & Ahlquist, 2011, pp. 27-28). “*End-user programming is thus a form of meta-work, in which the programmer must reflect on the task at hand, develop, test and refine tools to aid it and then use those tools to accomplish the actual tasks.*” (Woodbury, 2010, p.66).

O desenvolvimento da computação não é limitado pela procura focada dos procedimentos algorítmicos. Assim o programador ou *software developer* é beneficiado pela redução da margem de erro, ao mesmo tempo que esta valoriza a exploração abstrata de soluções.

The determinism of algorithms leads to an indeterminism. In this way the character of creativity is an open horizon, even though it is generated through a finite number of rules. Creativity means algorithmic design, which needs to be conceived in a way that even the unpredictable, that is occurrences far beyond the subjective horizon of design, can be designed for. Creativity entails algorithmic design plus the complementarity of designing additional system components. (...) The creative figure will be both the designer of algorithms and the interpreter of their outcomes. (Peter Weibel, citado por Menges & Ahlquist, 2011, pp. 27-28)

O arquiteto não precisa da programação para trabalhar com métodos computacionais ao desenvolver o seu projeto com o apoio do computador. Mas, de facto, existem vantagens nesse conhecimento, pois eleva a especialização dos procedimentos algorítmicos e a evolução de métodos de abstração. Qualidades que valorizam o arquiteto que se dedique a estas área de estudo e que pretenda abordar tais metodologias de trabalho. A capacidade de poder, de forma autónoma, participar no desenvolvimento de *software*, ferramentas de trabalho, para facilitar e desenvolver todas as questões inerentes ao processo de conceção de um projeto de arquitetura, são desbloqueadas com conhecimentos de programação.

Como referido anteriormente, um programador procura ter o seu código organizado e sintetizado, enquanto, o arquiteto usa o código apenas como um bem para cumprir o seu objetivo, um meio para atingir um fim. Em 2007, Brandy Peter, para a conferência de ACADIA, desenvolveu várias alternativas para a cobertura do edifício Patent Office Building, do atelier Foster + Partners. Estas alternativas foram geradas através de modelação paramétrica, onde o código de computador permitia gerar diferentes alternativas para essa cobertura. Esta apresentação causou alguma polémica entre os especialistas em programação, já que o código se encontrava apenas organizado o suficiente para que funcionasse, contendo imperfeições, como blocos repetidos. Questionado sobre o porquê de não estar devidamente organizado, Brandy Peter respondeu: *"I didn't need to do that". "Peter wasn't being lazy or uncraftful; he was being a designer."* (Woodbury, 2010, p.66)

É neste aspecto que o arquiteto se diferencia de um programador, ele complementa o uso da programação, com os seus conhecimentos e usa-o como uma ferramenta para execução das suas tarefas, quer sejam exploratórias, como os sistemas paramétricos, ou precisas ao desenvolver uma ferramenta CAD para computadorização. Um arquiteto que sabe programação ou que a entende, contribui tanto para o desenvolvimento dos meios de conceptualização do projeto, como para a facilitação do seu trabalho.

2.2.2 | Teoria de Sistemas

(...) how architectural problems may be solved through an analogous process where design forms through the interactive readings and responses to interrelational conditions, with the intention of producing environments synchronous with their cultural settings. (Menges & Ahlquist, 2011, p. 58)

A teoria de sistemas genéricos foi formulada como uma disciplina científica universal para decifrar as bases que definem *“organized wholes”* (Menges & Ahlquist, 2011, p.15). É possível remeter as bases da teoria ao trabalho de Aristóteles pela sua definição de Holismo nos seus estudos metafísicos - *“(...)’The whole is more than the sum of its parts.’ Aristotle viewed the world and each being in the world as mutually interrelated and part of a teleologically organized whole.”* (New World Encyclopedia, 2023). Estudiosos como Descartes, físico e filósofo francês, que também trabalhou sobre o tema ao qual adicionou: *“the notion of understanding processes via simple causality, with relationships deduced from, at most, two interacting elements.”* (Menges & Ahlquist, 2011, p.13). A meio do século XX Ludwig von Bertalanffy, introduz a teoria de sistemas como uma crítica às visões clássicas da física, nos seus métodos dedutivos focados em avaliações de fenómenos isolados. O biólogo considerava tais métodos inadequados para estudos na área da biologia, onde nada na natureza existe por si só, isolado e sem qualquer dependência, mas sim como uma parte integrante de um complexo sistema de interações e reações (Menges & Ahlquist, 2011, p.15).

An open system is defined as a system in exchange of matter with its environment, presenting import and export, building-up and breaking-down of its material components. (Ludwig von Bertalanffy citado por Menges & Ahlquist, 2011, p. 15)

Nos anos sessenta com a evolução do computador e o início dos desenvolvimentos do mesmo como uma ferramenta de trabalho, começaram a ser formuladas as primeiras visões de metodologias de trabalho nas áreas da arquitetura, design e urbanismo. Um tema que ainda hoje é abordado e desenvolvido, facilmente é remetido aos autores dos anos sessenta e setenta como Gordon Pask, Cedric Price, John Frazer e Christopher Alexander, que podem ser considerados como *“builders of bridges”* entre ciência e arquitetura que demonstram abordagens aos processos de projeto interdisciplinares.

Every aspect of form, whether piecelike or patternlike, can be understood as a structure of components. Every object is a hierarchy of components, the large ones specifying the pattern of distribution of the small ones, the small ones themselves, though at first sight more clearly piecelike, in fact again patterns specifying the arrangement and distribution of still smaller components. (Christopher Alexander 1964, citado por Menges & Ahlquist, 2011, p. 11))

Arquitetos como Christopher Alexander intitulado pelo The New York Times (2022) como *Architect Who Humanized Urban Design*, ao desenvolver a teoria de sistemas, de modo a ser implementado no processo de projeto e design, abriu portas à ligação direta entre ciência e arte. Durante a sua carreira, Alexander redefiniu constantemente as suas teorias e visões sobre data structures, do qual é possível verificar uma semelhança entre os processos de programação de computadores, com os métodos de organização da informação, por diagramas, no processo de problem-solving inerentes ao projeto. *“The decomposition of the problem should be processed in a standardized manner, following rules, as if a computer performed it.”* (Pertigkiozoglou, 2017).

O arquiteto inicia o seu artigo *“Systems Generating Systems”* para a revista *Architectural Design* (1968), fornecendo as bases para o entendimento da teoria de sistemas:

1- There are two ideas hidden in the word system: the idea of a system as a whole and the idea of a generating system.

2- A system as a whole is not an object but a way of looking at an object. It focuses on some holistic property which can only be understood as a product of interaction among parts.

3- A generating system is not a view of a single thing. It is a kit of parts, with rules about the way these parts may be combined.

4- Almost every ‘system as a whole’ is generated by a ‘generating system’. If we wish to make things which function as ‘wholes’ we shall have to invent generating systems to create them.

In a properly functioning building, the building and the people in it together form a whole: a social, human whole. The building systems which have so far been created do not in this sense generate wholes at all. (Alexander, 1968/2011, pp.58-59)

A palavra sistema, é uma palavra comum, possuidora de um carácter técnico, e que possibilita diversas interpretações. A falta de precisão na sua definição, ou o valor de abstração da palavra, permite o desenvolvimento de novas ideias mesmo que vagas. Esta palavra consegue assim, instruir o desenvolvimento e a exploração de ideias, que por sua vez, poderiam ser refutadas de início por falta de precisão ou fundamento. O entendimento da teoria de sistemas, contribui para a estabilização e entendimento de procedimentos e metodologias de trabalho apoiados no uso do computador. Desviando dos processos de trabalho, baseados na “computadorização” e valorizando o pensamento computacional, em modos exploratórios aplicados ao projeto de arquitetura.

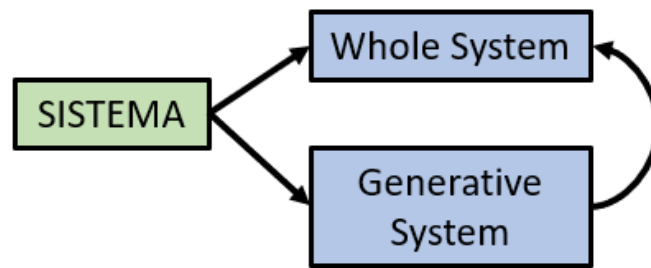


Fig. 13 | A palavra Sistema gera duas ideias que se interligam.
(Imagem do autor)

A teoria de sistemas, deve ser assimilada como uma visão para entender o funcionamento dos sistemas, uma forma de compreensão do problema, um ponto de vista, para impulsionar o desenvolvimento do processo de resolução do problema (*problem-solving*).

A ideia de sistema como um todo (*whole system*), refere-se a uma visão holística em particular de uma única coisa. Por outro lado, a ideia de um gerador de sistemas (*generative system*) não se refere a algo, uma coisa propriamente dita, mas sim, a um conjunto de partes (*kit of parts*), que associado a regras específicas (*combinatory rules*) permite gerar diversas coisas (Alexander, 1968/2011, pp.58-59). Christopher Alexander (1968/2011) tende a fornecer definições com identificações vagas, ao mesmo tempo que realiza associações transpostas para a “realidade”, como exemplos. Estas definições sobre uma caracterização abstrata permitem, por sua vez, anular os pré-conceitos e transmitir ao leitor a lógica por detrás da teoria, em vez de uma associação concreta.

The system viewpoint is a modern, disciplined, version of the sense of wonder. It is that view of things which man takes when he becomes aware of oneness and wholeness in the world. (Alexander, 1968/2011, pp.59-60)

Sistemas como um todo (*whole system*), para serem compreendidos, devem ser vistos como fenómenos holísticos, onde o sistema é o produto da combinação das suas partes. Para uma correta avaliação de um sistema é necessário ter em conta todas as forças externas que o constituem e influenciam. Alexander (1968/2011) fornece exemplos de sistemas, associados a objetos, onde pretende destacar as suas propriedades holísticas:

Exemplo 1:

The great depression is an obvious example of a holistic phenomenon. We cannot understand the depression, except as a result of interaction among rates of consumption, capital investment and savings: the interactions can be specified in the form of equations; if we follow these equations through to their conclusion, we see that under certain conditions they must always lead to a depression. (Alexander, 1968/2011, p.59)

Exemplo 2:

Another example of a holistic property is the relation between input and output in any computer. In the toy computer called ThinkADot, a ball dropped into one of three holes comes out on one of two sides. The output side is not determined by the input hole, but by the input hole and the internal state of the machine, which is itself determined by the sequence of past inputs. In order to understand this behaviour, we must understand the machine as a whole, considering the past inputs and the internal states as parts, and the way that different sequences of inputs and internal states create specific new internal states and outputs as interactions. (Alexander, 1968/2011, pp.59-60)

Em ambos os exemplos verificamos um objecto que só pode ser entendido como a causa da interação das suas partes dentro do próprio objeto, em suma esta é a definição de uma propriedade holística. Devemos ter em conta o conceito de instabilidade ao avaliar os objetos. Esta propriedade é definida para os objetos holísticos vulneráveis à mudança, onde a alteração de uma das suas partes pode desencadear alterações em cadeia das outras partes.

The central point of the whole argument can be stated very simply. The most important properties which anything can have are those properties that deal with its stability. It is stability which gives a thing its essential character. (Alexander, 1968/2011, pp.59-60)

No entanto, a estabilidade, independentemente da sua forma, é uma propriedade holística e a essência da caracterização. *“The strength of an arch, the even burning of a flame, the growth of an animal, the balance of a forest ecology, the steady flow of a river, the economic security of a nation, the sanity of a human individual, the health of a society: these are all, in one way or another, concerned with stability.”* (Alexander, 1968/2011, p.60). Todos os exemplos, só podem ser entendidos pelo produto da interação das suas partes, visto que no centro têm de existir algum tipo de estabilidade, que comprime as interações dentro do todo. No entanto, devemos ter em conta que a palavra sistema não está a ser associada a um objeto em si, ou seja um objeto físico, mas sim a um objeto como um elemento de estudo, um modelo.

Most often common language obscures this very badly. When we speak of the solar system, or a hi-fi system, or an airline system or of a plumbing system, the words are used in such a way as to suggest that the ‘system’ is synonymous with the objects. For instance, when we speak of the Ptolemaic system as opposed to the Copernican system, in each of the cases the word ‘system’ is used correctly: it refers to an abstract way of looking at the interaction among earth, planets, sun and stars — not to the objects themselves. (Alexander, 1968/2011, p.62)

Alexander (1968/2011) afirma que a primeira regra a ter em conta para compreender Sistemas como um todo (*whole system*), é não associar este a uma coisa, (p.64). *“When we view a thing in such a*

way as to reveal its character in holistic terms, we speak of it as a system." (Alexander, 1968/2011, p.60). Deste modo, a ideia de sistema está diretamente associada à ideia de um modelo abstrato com propriedades holísticas. Alexander (1968/2011) dá como exemplo um sistema económico de um país, do qual é possível replicar propriedades holísticas como a inflação e a depressão, a partir de um sistema de equações. Para definir algo como um sistema, é necessário cumprir três pontos:

(1) the holistic behaviour which we are focusing on;

(2) the parts within the thing, and the interactions among these parts, which cause the holistic behaviour we have defined;

(3) the way in which this interaction, among these parts, causes the holistic behaviour defined. (Alexander, 1968/2011, p.60)

Sistema é uma forma de explicar algo, que só pode ser identificado pelas interações dentro do todo (*whole system*). Proveniente da abstração específica de um comportamento com propriedades holísticas, com interações entre as partes e com o processo das mesmas. Para denominar algo de sistema, é necessário um contexto e lógica, para que este não se torne em algo demasiado complicado para ser compreendido. A ideia de sistema só tem utilidade para identificar os comportamentos que resultam das interações entre as partes. Alexander (1968/2011) corrobora esta ideia através do paralelismo entre uma flor e um sistema. Ele demonstra que entender a flor como um sistema, permite compreender os seus comportamentos, como quando esta floresce ou murcha. Ações que são o resultado da interação entre as partes no todo. No entanto, se a mesma flor vista como um sistema, for usada como um presente ou um projétil, deixa de ser possível explicar o desfecho, como proveniente do resultado das interações entre as diferentes partes da flor (p.62).

Furthermore, even though we call a thing a system when we try to view it as a whole, this does not mean that we ever really view the thing in its entirety. (...) The notion of 'whole' refers only to the breadth of vision, not to the inclusion of detail: it is still abstract. (Alexander, 1968/2011, p.62)

A visão de algo como um sistema deve ser focada nas interações que realmente influenciam o todo. A abstração atua como um filtro do ruído desnecessário, à semelhança do uso da abstração na ciência dos computadores, onde *"an abstraction describes of a class of instances, leaving out inessential detail."* (Woodbury, 2010, p.30). O foco incide sobre os comportamentos e interações explícitas e não no objeto. No entanto, a disciplina da abstração também pode proporcionar algumas complicações. Este processo, quando é confrontado com fenómenos, que são o produto de interações com um alto nível de complexidade, torna-se difícil uma visão clara dessas mesmas interações. Nestes casos, uma invocação rígida sobre a ideia de um sistema como um modelo abstrato, poderá levar à abstração de sistemas mais óbvios e superficiais. Causado pela intenção de simplificar e compreender algo que vá para lá da nossa compreensão atual (Alexander, 1968, p.62).

This is exactly what happens when a systems analyst looks at a building: he/she manages to describe the circulation, the acoustics, the heating and the load-bearing structure as systems — and fails to identify the most interesting human and social systems, because he/she can't describe them in explicit terms. (Alexander, 1968/2011, pp.62-64)

O confronto com algo bastante complexo, normalmente leva a uma associação que funciona como um sistema. Esta associação é dada por um sentido ou um instinto, que por sua vez o sujeito vai tentar abstrair esse comportamento holístico, focando apenas nas interações essenciais que provocam o dito comportamento. Este é um processo ativo, que se inicia por um instinto e apenas se torna num pensamento mais tarde.

Alexander (1968/2011) afirma a necessidade de compreender estas duas regras. A primeira já verificada anteriormente, em que um sistema não é objeto (*thing*), a não ser que seja possível identificar o seu sistema abstrato. A segunda regra, é que devemos aprender a primeira regra, sem que esta nos leve a fazer abstrações fáceis e supérfluas. A primeira regra traduz-se no instinto, a necessidade de assimilar primeiro que algo funciona como um sistema, a segunda vai de encontro ao pensamento lógico, por detrás da abstração de propriedades holísticas que causam o comportamento. Uma visão da interação entre as partes no sistema como um todo (*whole system*) (p.64).

A building system is a generating system in this sense. It provides a kit of parts - columns, beams, panels, windows, doors - which must be put together according to certain rules. (Alexander, 1968, p.65)

Como dito anteriormente a ideia de sistema divide-se em dois conceitos: sistemas como um todo (whole systems) e sistemas generativos (generative systems). A palavra sistema é então usada como “*a way to do something*”. Podemos averiguar casos como um sistema de apostas, este vai nos informar, após uma avaliação de dados, onde devemos colocar as nossas apostas. Na sua essência é um sistema de regras que gera algo, neste caso a aposta “melhor”⁴. Um sistema generativo consiste num conjunto organizado de partes ou elementos (*kit of parts*), com regras pré-definidas, que por sua vez, permite gerar algo (Alexander, 1968/2011, p.64).

The formal systems of mathematics are systems in this sense. The parts are numbers, variables, and signs like + and =. The rules specify ways of combining these parts to form expressions, ways of forming expressions from other expressions, ways of forming true

⁴ Melhor é colocado entre aspas porque, no caso de um sistema de apostas dependendo do jogo em avaliação, poderá não haver uma resposta certa, mas sim uma resposta com maior ou menor probabilidade de sucesso.

sentences from expressions, and ways of forming true sentences from other true sentences. (Alexander, 1968/2011, pp. 64-65)

O segredo está no cumprimento das regras de combinação das partes, geradas pelo sistema, que irão produzir algo que é resultado de um fundamento com significado. Um sistema generativo não tem de ser complexo, apenas precisa de ser coerente, podendo ser desenvolvido por um *Kit of parts* simples, combinado por sua vez também por regras simples. A tipicidade e vantagem dos sistemas generativos é não estarem vinculados apenas a uma solução, ou seja, detém a capacidade de fornecer diversas respostas. A versatilidade da abstração das regras até um certo nível lógico, permite o funcionamento do sistema generativo, ao mesmo tempo que possibilita uma maior variedade de combinações entre as partes (Alexander, 1968/2011, p.65). Criando uma variação que se verifica tanto no processo como na resposta. Podemos ter como exemplo de sistemas generativos uma língua, um sistema democrático ou sistemas naturais como o sistema genético

There is a relationship between the two ideas of system which have been defined. Almost every object with behaviour that depends on some 'system as a whole' within the object, is itself created by a generating system. (Alexander, 1968/2011, p.65)

Existe uma relação pré-estabelecida entre um sistema holístico e um sistema generativo. A partir do momento em que um objeto tem alguma propriedade holística, que seja o resultado da interação entre as suas partes. Por sua vez estas partes e as suas interações, só existem, caso exista algum tipo de restrição (regras) das relações entre elas. O objeto é gerado pela combinação das partes de acordo com as regras que irão definir as interações válidas entre as partes, sempre que o sistema entra em funcionamento. Verifica-se assim um sistema generativo (Alexander, 1968/2011, p.66). Alexander (1968/2011) no seu artigo fornece, mais uma vez, exemplos que facilitam esta compreensão dos quais se destacam:

Exemplo 1:

(...) the railroad switch-yard. It plainly functions as a whole. In order to understand it as a device for breaking up and making trains, we must focus on the sequence of switches, and on the fact that the length of track in front of the switches depends on the length of track behind the switches and on the length of trains. At the same time, the switch-yard is also plainly generated by a generating system. The pieces of track, switches, couplings, cars, together with the rules for putting them together, form a kit of parts which generates properly functioning switch-yards.

Exemplo 2:

The most complicated case of all, and the clearest, is that of an animal. A landing seagull certainly needs to be seen as a system: so does almost everything else that seagulls do. At the same time, this seagull is created by a generating system: the genetic system. An animal is both something which needs to be seen holistically, and generated by a generating system. (p.66)

Torna-se evidente, que em grande parte dos casos, um sistema como um todo é gerado, por sua vez, por um sistema generativo, que não precisa de ter uma consciência própria como o *railroad switch-yard* (exemplo 1). Para o arquiteto que por vezes, têm um pensamento focado no design de objetos. O foco de trabalho deve ser sobre a compreensão das propriedades holísticas dos nossos objetos de estudo. Uma cidade precisa de suportar vidas, um edifício precisa de funcionar no seu contexto social ou humano como um todo. O arquiteto precisa de criar objetos com complexas propriedades holísticas. Para isso, é necessário construir sistemas generativos, em que as partes e as regras que o compõem, desenvolvam sistemas com propriedades holísticas, que respondam aos problemas com que lida a arquitetura.

The designer becomes a designer of generating systems — each capable of generating many objects — rather than a designer of individual objects. (Alexander, 1968/2011, p.66)

Uma questão importante para um uso benéfico deste conhecimento é o espírito crítico, pois nem todos os objetos gerados por um sistema generativo, são sistemas com propriedades holísticas com valor útil. Numa aproximação ao contexto arquitetónico, um edifício com propriedades holísticas, considera o próprio edifício e as pessoas como um todo (*whole system*).

Holism is of several types:

a) A functionally interpreted building can only be usefully considered in the context of a city (notice that the city is also functionally interpreted and, as a result, is a dynamic entity).

b) A (functionally interpreted) structure, either a building or an entire city, can only be meaningfully conceived in the context of its temporal extension, ie, its growth and development.

c) A (functionally interpreted) structure exists as part of an intention, ie, as one product of a plan.

d) If (assumed dogma) man should be aware of his natural surroundings, then buildings should be wedded to or arise from these surroundings (Wright's organic thesis). Pask, 1969, p. 494)

Alexander (1968/2011) promove a criação de sistemas generativos, que produzem projetos funcionais com respostas às forças externas que interagem com o edifício como um todo (*whole system*). Estas ideias de controle e compreensão dos sistemas estão diretamente integradas na teoria da cibernética.

2.2.3 | Teoria da Cibernética

Reyner Banham considered that “... architecture is just one of a dozen ways of approaching the design of buildings, but it happens to be the one which western civilization still grants cultural hegemony”. (...) Currently, “intelligent” buildings seem to be defined predominantly by a high level of automated services and what seems to be a preoccupation with increased “control” of information and energy management systems. Arguably, “human centred” and integrated approach is warranted in the utilisation of available technologies in order to facilitate novel experiential domains for “human-human” and “human-machine” interaction. (Hunt, 1995, p.53)

A teoria da cibernética ganhou maior prestígio nos anos sessenta, ao ser transposta para as mais variadas áreas de estudo, inclusive na arquitetura. Norbert Wiener, o matemático americano e conhecido como o fundador da teoria no final dos anos quarenta, definiu-a pela primeira vez, no seu livro lançado em 1948 intitulado *“Cybernetics or Control and Communication in the Animal and the Machine”*. Wiener caracteriza a teoria como *“the study of control and communication in the animal and the machine”*, afirmando-se como um estudo que visava compreender sistemas de grande complexidade. Uma avaliação sobre o estudo de processos de aprendizagem, adaptação, emergência, comunicação, evolução e eficiência proveniente das interações dos sistemas. A cibernética, por ser uma teoria com caráter interdisciplinar, influenciou as mais variadas áreas de estudo, e por sua vez também evoluiu com elas. Podemos destacar a sua aplicação nas investigações inerentes à psicologia e teoria de controle (*control theory*), filosofia e engenharia mecânica, arquitetura e evolução biológica, ciências sociais e engenharia eletrônica (Marinescu, 2017).

Unlike traditional scientific approaches, they are highly interdisciplinary and deal with the transformation of information and feedback in self-regulatory systems... (Pertigkiozoglou, 2017)

Como referido anteriormente, a cibernética surge como uma tentativa de compreender a relação máquina-homem por analogias que visualizem a máquina como um organismo. As avaliações e transformações da informação por feedback permitem sistemas auto-reguláveis, que transpostos para o desenvolvimento tecnológico da máquina, permitem que esta seja mais flexível à adaptação. A máquina passa assim, a se auto-ajustar ao ambiente no qual é inserida, facilitando as interações com o operador ao mesmo tempo que evolui com o aumento da complexidade do mundo em redor (Hunt, 1995, p.54).

(...) *first-order cybernetics – “observable systems”, to second-order – “observing systems”, to third-order cybernetics – “self-developing poly-subject (reflexive-active) environments”. And also on the ascent from the paradigm “subject – object” to the paradigm “subject – subject” and then, in third-order cybernetics, to the paradigm of “subject – metasubject (self-developing poly-subject environment).* (Chaturvedi, 2020)

A teoria da cibernética evoluiu, até a atualidade, em três níveis de ordem relativamente ao controle por feedback. A cibernética estabilizou-se inicialmente como o estudo do sistema complexo e organizacional dos seres vivos, uma vertente que é facilmente correlacionada às temáticas da arquitetura. Instalando-se assim um novo paradigma no desenvolvimento de um projeto, em que este necessita de ter a capacidade de “auto regulação”, uma máquina com auto-controlo no seu comportamento (*homeostatic*). Deste modo, os primeiros desenvolvimentos são focados na primeira ordem cibernética, ideia sumarizada por Francis George (1977) como - *“primarily concerned with certain specific features of communication and control, especially to those that synthesise (copy in terms of ends rather than means) or simulate (copy in terms of ends and means) organismic behaviour.”* (Hunt, 1995, p.54).

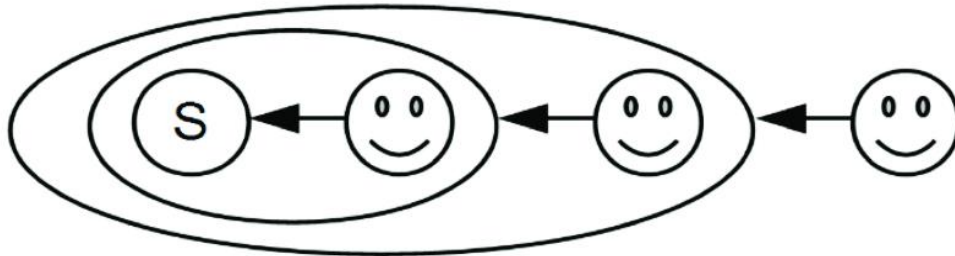


Fig. 14 | Relação de sujeito com 1ª, 2ª, 3ª Ordem de Cibernética (Ashby, 2020)

Após a estabilização do conceito, possibilitado pelo desenvolvimento da teoria durante o pós-guerra, é possível verificar, uma alteração de paradigma na segunda metade do século XX. Estudiosos, arquitetos e inventores iniciam a procura de uma aplicação prática da teoria na disciplina de arquitetura. Nos anos setenta do século passado, Cedric Price com o apoio de John Frazer como consultor computacional, ambos formados nos anos sessenta na Universidade de Cambridge e Associação Arquitetural de Londres, conceberam o projeto *“Generator Project”* em 1976. Este foi um projeto produto do seu tempo, por ser desenvolvido num período de valorização do computador, ao mesmo tempo que eram explorados os processos computacionais, com interpretações provenientes do estudo de processos naturais, influenciados pela cibernética (Pertigkiozoglou, 2017).

Lacking a specific function, the Generator was conceived to serve a number of different uses and consisted of a kit-of-parts which would enable enclosures, gangways, screens and services to be altered in accordance with the changing needs and wishes of the users. (Hunt, 1995, p.54)

A proposta pretendia que os habitantes do edifício fossem encorajados a participar na construção do mesmo. Participando ativamente na reorganização intencional sobre o projeto, adequando este às suas necessidades. Numa primeira versão, não existia nenhum programa específico de otimização e avaliação sobre o projeto, essa tarefa pertencia ao usuário que perante o resultado das suas ações,

fornecia o feedback com as alterações. Por sua vez, a construção era realizada a partir de guias e guindastes que no local de implantação iriam dispor os componentes do edifício (Hunt, 1995, p.54).

“Sited in Florida, the Generator’s services and structure respond to the user’s wishes with help from both carnage and computer”. There was no fixed program, just a “desired effect” .“A menu of items caters for individual and group demands for space, control, containment and delight”. (Pertigkiozoglou, 2017)

Todos os movimentos, decisões e reajustes eram documentados, para verificar os padrões sobre o uso da ferramenta e os desejos e necessidades do usuário. *“Spatially, the need for a continuous readjustment was translated to a grid, a scheme of 150 12’ by 12’ mobile, combinable cubes.”* (Pertigkiozoglou, 2017). O trabalho inicial de Price, mais concretamente os seus diagramas sobre o processo da cibernética aplicado ao projeto, são associados ao circuito elétrico da *Homeostat*, tendo sido um dos primeiros dispositivos com capacidade de adaptação ao ambiente (Pertigkiozoglou, 2017).

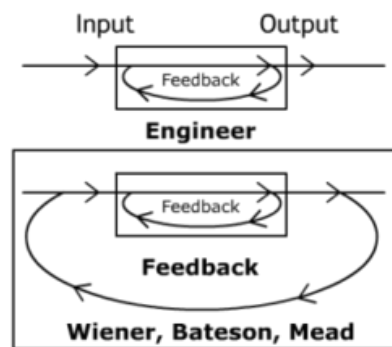
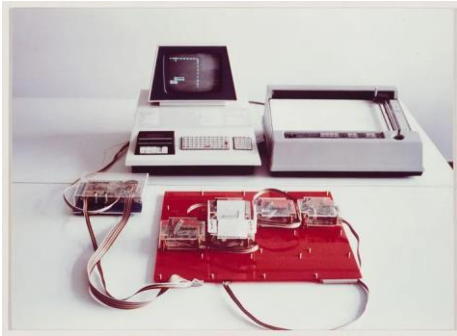


Fig. 15 | Diferença entre 1º ordem de Cibernética com 2º segunda ordem (Autor Desconhecido)

O Generator Project, numa fase inicial foi baseado na primeira ordem cibernética (sujeito-objeto), no entanto o apoio de John Frazer ao projeto permitiu desenvolver processos na segunda ordem cibernética (sujeito-sujeito), afirmando-se como o “primeiro edifício inteligente” (Hunt, 1995, p.54). Semelhante ao trabalho de Gordon Pask com o Musicolor, John Frazer propôs que o próprio edifício fizesse sugestões em relação às configurações existentes (Pertigkiozoglou, 2017). Assemelhando-se à capacidade de evolução “natural”, perante o impacto das forças externas com que interage. Uma analogia biológica entre a ciência e a arquitetura, em que o próprio edifício adapta-se e evolui, consoante o *feedback* recebido ao longo do seu tempo de vida (Hunt, 1995, p.54).



As parts are added to the three- dimensional model the microprocessor control unit interrogates the structure to discover the present state of its configuration. A television monitor screen displays statistical data, analysis or just shows projections of the current state of the model. The tape recorder stores the image of the model for future use for further calculations and the production of working drawings. (Frazer, 1998)

Fig. 16 | Generator Project (eliza-pert.medium.com)

A visão da arquitetura como um criador de sistemas com propriedades holísticas introduz por sua vez a característica do tempo à equação, como um fator impulsionador da mudança. Da mesma forma que os animais evoluem para melhor se adaptarem às necessidades do tempo, também o edifício deve ser visto para resistir e adaptar-se. O foco principal não está na construção de um edifício mutagénico, mas sim num edifício previamente idealizado e pensado sobre as necessidades do tempo. É deste modo que a cibernética introduz na arquitetura uma visão mais humanizada, onde o conceito de arquitetura funcional é aprimorado sob o pretexto de um edifício integrante do sistema.

It follows that a building cannot be viewed simply in isolation. It is only meaningful as a human environment. It perpetually interacts with its inhabitants, on the one hand serving them and on the other hand controlling their behaviour. In other words structures make sense as parts of larger systems that include human components and the architect is primarily concerned with these larger systems; they (not just the bricks and mortar part) are what architects design. I shall dub this notion architectural “mutualism” meaning mutualism between structures and men or societies. (Pask, 1969, p.494)

Gordon Pask (1969/2011) vai contra os conceitos do que define de arquitetura “pura” bastante presente no século XIX, destacando o Temple Meads, Tropical House Kew e o Crystal Palace, como edifícios que apesar de cumprirem com as suas funções, não passam de soluções individuais e idiossincráticas. Talvez influenciado também por Bruno Zevi que em 1949 demonstra estas ideias no seu livro “Saber ver a Arquitetura” (*Saper vedere l'architettura*):

“... consiste no fato de os edifícios serem apreciados como se fossem esculturas e pinturas, ou seja, externa e superficialmente, como simples fenómenos plásticos. Em vez de uma falta de método crítico, trata-se de um erro de postura filosófica.” (p.5).

Projetos com ideais de arquitetura monumental, uma arquitetura que apenas serve como um elemento decorativo e escultórico da cidade, para além de serem projetados com base em pré-conceitos

arquitetónicos, que definem como um edifício deve ser, para albergar uma determinada função. A visão de um projeto com propriedades estáticas é apresentada por Gordon Pask, em contraste com um projeto com propriedades dinâmicas, onde o trabalho do arquiteto deve ter maior ênfase sobre as necessidades do sistema, ou seja, o sistema humano.

We consider all organisms to be deprived when their environments fail to lead them to coordinate themselves to an ever greater degree. It is only through architecture that this privation can be compensated for, or ended. (Arakawa & Gins, 1995, p. 44)

A cidade facilmente pode ser associada a um organismo vivo, o mesmo se aplica a um edifício ou um computador. Um sistema gerado por sistemas, que crescem, desenvolve-se e de um modo geral evolui. O papel do arquiteto como um dos modeladores da cidade, passa por desenvolver sistemas de controle, que de uma forma responsável devem considerar as propriedades holísticas e evolutivas dos sistemas. A cibernética apresenta-se como um processo de controlo, para identificar e selecionar as melhores soluções, para além de permitir a capacidade predictiva, para conceber um projeto planeado com visão sobre o futuro da cidade.

The more coordinated the organism, the less likely it is to become a victim of circumstance. The body that is architecturally augmented to promote a higher degree of self-coordination - the architectural body - has a better chance of continuing indefinitely than does an organism taken as distinct from the world surrounding it, a world that addresses its needs at best only half-consciously. (Arakawa & Gins, 1995, p. 44)

O trabalho do arquiteto fica facilitado quando este compreende que o que produz são sistemas dinâmicos, que não podem nem devem ser vistos por questões isoladas, mas sim avaliados no contexto integrado de um sistema. O desenvolvimento computacional e a facilitação de métodos para obtenção de dados concretos, permitem ao arquiteto um maior controle sobre o desenvolvimento do projeto. As suas decisões são tomadas com base na informação obtida, tornando a conceção da forma, como o produto dos processos que vai para além da estética.

Instead of first designing a building and then analyzing its impact on users, we start from the question of what the people working within a certain programmatic context actually need. Concentrating on people's truly essential needs is difficult - but once you have achieved that, it opens the door to entirely new creative opportunities. (Abdessemed)

A propriedade cibernética da arquitetura fornece ao arquiteto o controle sobre o projeto, onde o resultado final terá em conta as forças externas que irão influenciar o edifício através de feedback, devido a complexidade atual inerente ao desenvolvimento de um edifício. O trabalho colaborativo passa a ser importante no desenvolvimento de um projeto, em que o arquiteto apesar de ter controle sobre a concetualização do projeto, necessita de se aliar a especialistas de diferentes áreas, de modo a obter os dados que lhe permitem a tomada de decisões fundamentada e intencional. A facilidade, bem como a

procura pelo desenvolvimento de metodologias para obtenção de informação, de um modo generalizado, passam por uma análise de resultados através de feedback.

Creating building systems in the present sense is not enough. We need a new, more subtle kind of building system, which doesn't merely generate buildings, but generates buildings guaranteed to function as holistic systems in the social, human sense.
(Alexander, 1968, p.67)

Os exemplos apresentados nas figuras abaixo, são metodologias de avaliação por *feedback* em relação a edifícios construídos. Demonstradas como ferramentas, estão idealizadas sobre estratégias holísticas de avaliação do edifício, cada uma com os seus pontos de foco mais ou menos abrangentes. Tal como Christopher Alexander (1968) refere no seu artigo “*Systems generating Systems*” (p.62), a necessidade de filtrar a informação relevante, para entendimento dos sistemas. Um passo que também é verificado nestas metodologias de avaliação e obtenção de informação.

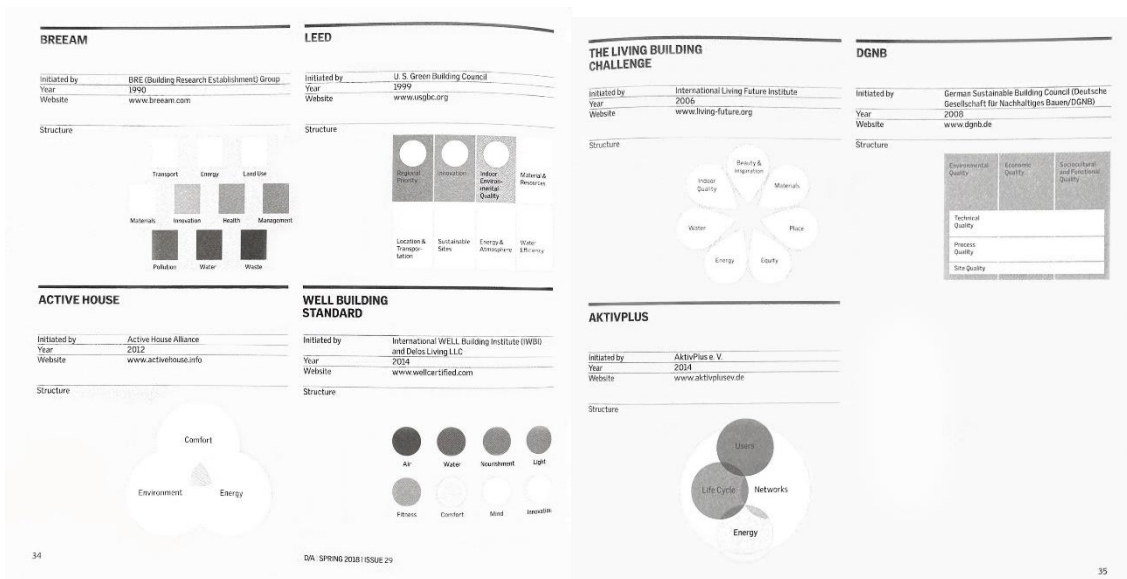


Fig. 17 | Metodos de estudo por *feedback* (“Schemes for Healthy Buildings,” 2018)

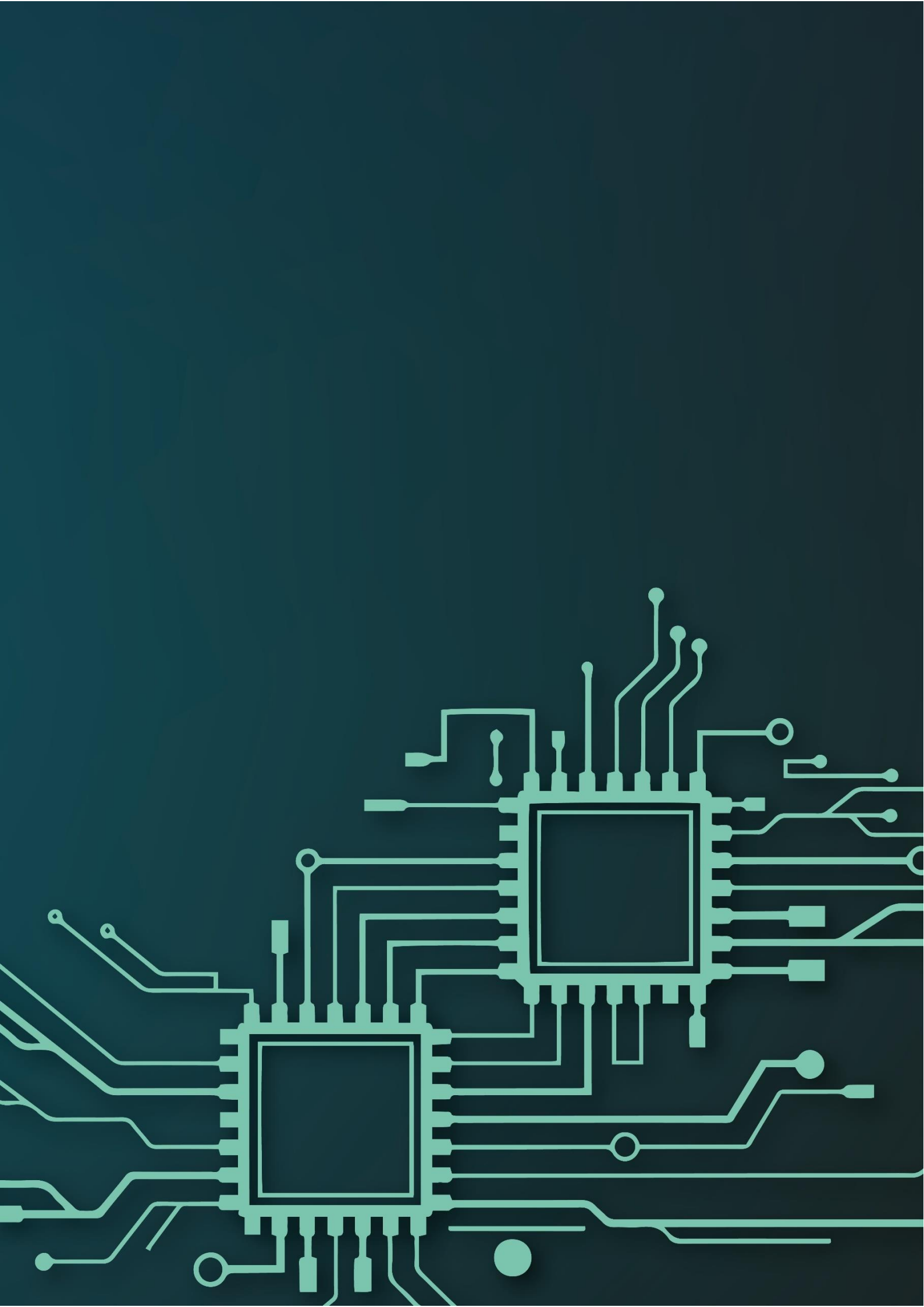
A teoria da cibernética aplicada à arquitetura permite uma avaliação computacional sobre o produto do arquiteto e as suas propriedades holísticas. Como demonstrado por Gordon Pask (1969) e igualmente fundamentado por Christopher Alexander (1968), o arquiteto que cria e desenvolve sistemas de controlo social (edifício, cidade, etc.), precisa de desenvolver um sistema que seja o resultado (*feedback*) da interação entre as partes (pessoas, natureza, animais), ou seja, o sistema gerado por sistemas e integrante do sistema como um todo (*whole systems*). Uma componente importante desta lógica é a obtenção de informação (*data*), ou seja, o feedback como um processo de avaliação crítica do projeto.

People are the best measuring instruments. They are just harder to calibrate. (Gary Row, p.82)

Esta metodologia de trabalho ajuda a verificar e entender como resolver os problemas do projeto neste caso já construído. A grande vantagem, passa pelo simples facto da informação fornecer ao arquiteto, a capacidade “predictiva” para os projetos futuros e permitir que este entre no processo de projeto muito mais cedo (Pask, 1969, p.495). Uma arquitetura evolutiva que encontra inspiração na cibernética para desenvolver os seus conceitos e dar ao arquiteto o controlo sobre o seu sistema, ou seja o seu projeto.

(...) an open relationship with the environment in both a metabolic and socio economic sense. It will maintain stability with the environment by negative feedback interactions and promote evolution in its employment of positive feedback. It will conserve information while using the processes of autopoiesis, autocatalysis and emergent behaviour to generate new forms and structures... Not a static picture of being, but a dynamic picture of becoming and unfolding - a direct analogy with a description of the natural world. (Frazer, 1992, p.21)

Uma componente importante desta lógica, passa pela obtenção de informação (*data*), ou seja, o feedback como um processo de avaliação crítica do projeto. Esta metodologia de trabalho ajuda a verificar e entender como resolver os problemas do projeto neste caso já construído. A grande vantagem, passa pelo simples facto da informação fornecer ao arquiteto, a capacidade predictiva para os projetos futuros e permitir que este entre no processo de projeto muito mais cedo (Pask, 1969, p.495). Uma arquitetura evolutiva que encontra inspiração na cibernética para desenvolver os seus conceitos e dar ao arquiteto o controle sobre o seu sistema, ou seja o seu projeto.





3. ARQUITETURA ALGORÍTMICA



3.1 | AAD (Algorithm Aided Design)

An algorithm is a computational procedure for addressing a problem in a finite number of steps. It involves deduction, induction, abstraction, generalization and structured logic. It is the systematic extraction of logical principles and the development of a generic solution plan. (Terzidis, 2003/2011, p.94)

A arquitetura algorítmica é o resultado da evolução tecnológica com aplicações práticas na profissão, que só foi possível pela elaboração e desenvolvimento de novos processos computacionais. O algoritmo em si é uma parte fundamental dos processos computacionais. Ahlquist e Menges (2010) corroboram esta premissa afirmando - “One critical aspect of working in a computational manner is the processing of information algorithmically.” (p.11) - é importante por sua vez salientar, que o processamento de informação com base em algoritmos, tal como a computação, não está vinculado ao uso do computador. A máquina ganha o seu valor por facilitar o processamento de informação, na sua grande maioria, com uma capacidade superior e nunca alcançável pelo ser humano, incluindo aquele que criou a máquina.

Neste sentido o uso de processos, ferramentas e metodologias com base em AAD (*Algorithm Aided Design*), passam por aumentar o espectro de possibilidades para resolução dos problemas inerentes ao projeto de arquitetura. Engloba tanto as questões de caráter subjetivo, por exemplo inerentes à conceção da forma, mas também a resolução de problemas técnicos com caráter objetivo. De modo geral, as áreas artísticas têm tendência a marcar uma posição céptica em relação aos processos de natureza mecânica que envolvem a racionalização da arte, e a ideia de uma criação não humana.

Most of the theories related to form pertain mainly to subjective interpretation and perception. In contrast, algorithmic logic involves a deterministic approach to form and its shaping forces; it suggests rationality, consistency, coherency, organisation and systemisation. (Terzidis, 2003/2011, p.94)

Como expressado anteriormente, uma das premissas do algoritmo, passa pelo desenvolvimento de processos com diferentes níveis de abstração, que permitem otimizar a eficiência dos processos para a aplicação generalizada (Woodbury, 2011). No entanto, na arquitetura o uso dos mesmos não é valorizado pela otimização e automatização dos processos e capacidade de imitar o arquiteto, as componentes subjetivas e criativas têm um grande peso (Menges & Ahlquist, 2011). Neste sentido, a AAD permite a concepção de processos de trabalho para fins exploratórios, ou seja, a geração de soluções lógicas, fundamentadas, eficazes e precisas influenciadas pela qualidade criativa do operador ou usuário.

O arquiteto não está treinado para pensar em processos que expressem à máquina a intenção e não um objeto, ou seja, a dificuldade de trabalhar sobre a abstração do objeto final (Woodbury, 2011), no passo a passo que leva a construção de um sistema de soluções, ou como Alexander (1968/2011) expressa um sistema que gera sistemas. Inclusive autores como Robert Woodbury (2010), Thomas Fisher e Christiane M.Herr (2001), Mateus, Roseta e Monteiro (2013) demonstram muito bem a importância e a

diferença entre a apresentação de uma solução para o problema em questão e a capacidade de saber resolver problemas. O primeiro é focado na conclusão da tarefa em questão e na apresentação de uma solução. O segundo aplica-se ao arquiteto apto a desenvolver processos para a resolução de problemas, em que a solução é o produto da racionalização de processos lógicos (logic-based). O pensamento algorítmico permite sintetizar os procedimentos necessários, passo a passo, que permitem a resolução do problema em partes, que combinadas formam uma solução com propriedades holísticas. O foco não passa por obter uma solução específica e única, mas sim obter um processo adaptável a vários problemas e que aborda diversas soluções, valorizando os processos de trabalho exploratório provenientes do pensamento computacional, com o uso do computador como uma ferramenta de apoio ao arquiteto.

Similarly in design, shape grammars, mathematical models, topological properties, genetic systems, mappings and morphisms are algorithmic processes aimed at exploring uncommon, unpredictable and uncharted formal properties and behaviours. (Terzidis, 2003/2011, p.95)

Os conceitos elaborados pelos estudiosos da segunda metade do século XX, bem como a vertente interdisciplinar da computação, permitiram o desenvolvimento de algoritmos com propósito próprio, ou seja, a criação de diversos sistemas que aplicam processos computacionais nas metodologias do projeto arquitetônico. Estes conhecimentos expressados nos anos sessenta por Gordon Pask, Cedric Price, John Frazer, Christopher Alexander e Ludwig von Bertalanffy, por sua vez não devem ser interpretados como explicações ou “tutoriais” de como usar as ferramentas algorítmicas, mas sim como bases para entendimento dos processos de lógica na resolução e interpretação dos problemas na arquitetura com o uso de metodologias AAD. Os sistemas gerados são baseados em algoritmos que integram tanto conceitos naturais como artificiais. Em termos de aplicação a metodologias de projeto e design, os sistemas paramétricos, generativos e evolutivos são os que têm marcado uma maior presença e demonstrado maior relevância nas disciplinas artísticas. Cada um com o seu próprio valor e propósito de aplicação.

3.2 | Sistema Algorítmico

O homem depara-se constantemente com novas adversidades provenientes do sistema em que está inserido. Uma cidade, um edifício ou até mesmo a sociedade são sistemas que confrontam a mudança constante e alterações do seu ambiente, cada um é o todo e a parte de um sistema que deve ser percebido como dinâmico. O confronto com novos problemas e a obtenção de soluções, faz parte da própria natureza humana, que quando confrontado com as suas próprias limitações cria ferramentas para colmatar tais fraquezas. Tal como observado anteriormente, as tecnologias de informação e os processos computacionais são o produto do seu tempo, que por sua vez revelam-se com capacidades de processamento de informação muito superiores ao ser humano. Os sistemas concebidos com base em algoritmos, são processos computacionais cada um com a sua lógica própria na resolução de problemas. Estipulam procedimentos com funções pré-definidas que abordam o problema de maneira diferente. A abstração por sua vez permite o desenvolvimento de sistemas que abrangem uma maior área de aplicação.

Os processos computacionais permitiram o desenvolvimento de metodologias com aplicações diretas na arquitetura e no modo como o projeto é abordado. O computador e os *softwares* desenvolvidos para uso do arquiteto, permitem implementar os sistemas algorítmicos, por sua vez, os próprios *softwares* são concebidos com base nesses mesmos sistemas. O arquiteto atualmente tem à sua disponibilidade uma ferramenta com capacidade para produzir, criar e explorar soluções criadas com sistemas algorítmicos. Os sistemas mais abordados por estes *softwares* passam pelo uso de sistemas paramétricos, generativos e evolutivos, que por vezes são implementados em conjunto, quando relevante para a resolução do problema. A conjugação de processos computacionais, por sua vez, permite viabilizar novas utilizações práticas do sistema, como por exemplo o uso de um *software* que permite desenvolver os sistemas paramétricos, que integra ferramentas com base em sistemas generativos para gerar “infinitas” soluções.

Sistemas Paramétricos

Um sistema paramétrico é um processo computacional, baseado em parâmetros e restrições (*constraints*) predefinidas no sistema para gerar variações na solução base. Um processo muito presente em metodologias dedicadas ao desenvolvimento do projeto e design, permitindo a simplificação e otimização do processo de produção da solução, pela automatização e mecanização do mesmo, sem que seja necessário modelar ou programar de raiz. Os usuários executam as alterações em parâmetros específicos, com visualização em tempo real do impacto dessas modificações no resultado (*output*). A funcionalidade do sistema depende de informações explícitas fornecidas pelo operador e segue um conjunto predeterminado de regras, desenvolvidas com base em procedimentos algorítmicos, que permite gerar as diversas variações da solução inicial (Nagaraj & Werth, 2020). No entanto deve-se ter em conta, que este tipo de sistema é por si só limitado no fornecimento de soluções alternativas, sendo apenas possível gerar variantes da solução base que é o produto dos parâmetros e restrições (*constraints*).

Os sistemas paramétricos permitem resolver o problema através de vários processos diferentes, caracterizando-se pela abordagem na resolução do problema por via de restrições (*constraints*) e parâmetros (Woodbury, 2010, p.12). No entanto estes processos para serem corretamente usados, é necessário que as restrições sejam expressas antes de serem resolvidas. O número de restrições evidenciadas é proporcional à complexidade do problema, ou seja um projeto ou parte de um projeto

poderá albergar a definição de centenas de restrições de modo a obter um sistema que gera a solução base e as suas variantes (Mateus, Roseta e Monteiro, 2013). Hoffmann e Joan-Arinyo (2005) salientam três tipos de sistemas com abordagens diferentes baseada em gráficos (*Graph-based*), baseada em lógica (*Logic-Based*) e algébricas (*Algebraic*) (Woodbury, 2010, p.12). Um sistema paramétrico confere ao usuário o controle sobre a solução, pelo ajuste da mesma segundo as alterações proporcionadas pelo usuário, uma relação facilmente associada com a teoria da cibernética. Onde é possível afirmar que o comportamento do sistema, ou seja, o resultado (*output*) que este fornece ao usuário, é o produto do ajuste por *feedback* fornecido pelo operador que controla o sistema (Cui & Tang, 2017).

Existem diversos *softwares* CAD que integram ferramentas com base neste processo computacional, como por exemplo o Grasshopper, uma extensão do Rhinoceros, que permite construir um circuito paramétrico de raiz através de programação visual (*visual-scripting*), ou os softwares BIM como o Revit e o ArchiCAD, que por base já têm os elementos predispostos ao usuário, que executa a alteração dos parâmetros e informações iniciais (inputs), para melhor acomodar o resultado às suas necessidades.

Os sistemas paramétricos são utilizados em diversos campos como na engenharia, arquitetura e design de produtos. O uso dos mesmos torna-se benéfico em circunstâncias que exijam procedimentos com design ou projeto interativos, possibilitando a investigação de variantes. Por sua vez, os sistemas paramétricos na sua aplicação ao design e ao projeto, constantemente beneficiam da integração de ferramentas com base em sistemas generativos e evolutivos, para que possam abordar uma maior diversidade de problemas de projeto (Cui & Tang, 2017).

Sistemas Generativos

Um sistema generativo facilita a criação e a investigação de várias soluções pela exploração de alternativas. A exploração por sua vez, é executada por via da utilização de algoritmos que produzem a transformação do resultado automaticamente. Semelhante ao sistema paramétrico, as soluções apresentadas são geradas a partir de uma base, embora o sistema generativo permita a obtenção de soluções mais criativas, com características que se distanciam da base inicial. Este processo computacional procura obter o maior número possível de alternativas, sem foco na procura de uma resposta específica, indo para lá das informações fornecidas inicialmente, afirmando-se como um método de exploração de ideias e soluções a base de algoritmos (Cui & Tang, 2017). Estes tipos de sistemas possibilitam o desenvolvimento de metodologias com utilização prática no mundo atual, permitindo a criação de soluções dinâmicas com capacidade de interação com o contexto. São ferramentas desenvolvidas a partir *Cellular Automata*, *L-Systems*, *Genetic Algorithms* e *Shape Grammar*, traduzindo-se em ferramentas com base em conceitos matemáticos aplicados nas metodologias projetuais (Henriques et al., 2019).

Um sistema generativo é uma configuração com base em definições abstratas para a obtenção de alternativas únicas em relação à solução base, que permite a implementação e desenvolvimento de metodologias de trabalho de caráter digital ou analógico. Apesar de ser um processo inicialmente desenvolvido na obtenção de soluções em quantidade e não qualidade, o uso de algoritmos para filtrar a informação, permite por sua vez a avaliação da mesma, de modo a identificar soluções com maior potencial (Cui & Tang, 2017). A determinação do processo algoritmo ou metodologia a ser implementado ao sistema generativo provém das necessidades do problema em questão. Cabe neste caso ao arquiteto, identificar os objetivos e limitações do sistema, para que a exploração das soluções traga um maior número de benefícios ao desenvolvimento do projeto. Existem diversos processos que permitem o aprimoramento da informação, baseados em algoritmos específicos para a função, ao qual é possível

salientar o uso de algoritmos de natureza proveniente de AI, machine learning, sistemas evolutivos, rule-based e processos de projeto manual ou analógicos (Nagaraj & Werth, 2020).

Os sistemas generativos podem ser integrados e manipulados por sua vez através de outros sistemas. As metodologias de projeto que integram processos computacionais generativos, também tendem a abordar os sistemas paramétricos para produção do sistema generativo, fornecendo um maior controle sobre a solução gerada. Um exemplo real, é os softwares paramétricos como o Rhinoceros que, a partir da programação visual proporcionada pelo Grasshopper, permitem a construção de sistemas paramétricos com propriedades generativas. O desenvolvimento destes tipos de sistemas permite a ampliação do número possível de resultados com possibilidade de gerar uma série de alternativas à solução base provenientes do sistema. Uma geração que é proporcionada automaticamente a partir de algoritmos generativos, dispendo ao arquiteto uma variedade de soluções por filtrar. Neste sentido o arquiteto poderá usar algoritmos evolutivos para realizar a seleção das soluções automatizadas, sendo que este é apenas um de muitos processos possíveis para aprimoramento dos resultados. A validade e utilidade do sistema depende apenas da sua capacidade de cumprir os requisitos necessários no processo de *problem solving*.

Existem mais exemplos de *softwares* com base em sistemas generativos como é o caso do ChatGPT, que gera respostas segundo o *prompt* fornecido; a mesma lógica aplica-se as IA de geração de imagem, como o software DALL-E, Midjourney e Prome AI que produzem imagens como resposta aos *prompts* e parâmetros introduzidos no sistema. Este tipo de ferramentas, por enquanto, ainda precisam do feedback do usuário para executar e validar os seus resultados. Deste modo podemos afirmar que o usuário atualmente é quem têm o controle sobre o resultado neste tipo de sistemas generativos.

Sistemas Evolutivos

Um sistema evolutivo refere-se a um sistema que incorpora algoritmos evolutivos como método para gerar e evoluir as soluções ou projetos. Os algoritmos evolutivos representam os processos computacionais que se inspiram no fenômeno da seleção natural. O propósito destes algoritmos é aprimorar o resultado, ou seja, a obtenção de soluções otimizadas. Os resultados (*output*) são refinados a partir de processos interativos com base em algoritmos genéticos que têm por base os conceitos provenientes da natureza, como por exemplo a mutação genética, o cruzamento genético e os fenômenos de seleção natural. Os sistemas evolutivos podem ser vistos como uma variante dos sistemas generativos, mas com foco no desenvolvimento de soluções otimizadas e específicas, ou seja, a procura da solução que cumpre os requisitos de qualidade em contraste com a procura do maior número de alternativas provenientes de uma solução base (Martino & Celani, 2014).

Os sistemas evolutivos são processos computacionais que permitem obter e identificar as soluções ideais, ou seja, que melhor se adequa ao problema em questão. A solução é o produto gerado de requisitos especificados pelo usuário ou criador do sistema. Este tipo de sistema, para além de ser usado para automatização e otimização dos processos computacionais, também tem a capacidade de tomada de decisões, caso tal lhe seja permitido executar. O sistema, após criado e definido, irá fornecer a melhor solução possível de acordo com os requisitos; o usuário do sistema apesar de ter controle sobre o processo de refinamento, não tem controle sobre o resultado apresentado, que por sua vez será o produto fundamentado pelo sistema segundo uma lógica própria com base no algoritmo genético fornecido. A solução é objetiva e não subjetiva, pois foi a sobrevivente entre todas as possibilidades.

No entanto, este tipo de sistemas generativos e evolutivos podem apresentar soluções ou

resultados inesperados, a este tipo de produtos do sistema denominam-se de emergentes. A emergência refere-se ao fenômeno onde padrões, estruturas ou comportamentos mais complexos, surgem das interações entre os componentes ou entidades mais simples que integram o sistema. O sistema visto como um todo (*whole system*), demonstra-se como maior que a soma das suas partes, ou seja, o sistema passa a exibir propriedades ou comportamentos que não podem ser diretamente atribuídos ou previstos apenas a partir das suas componentes individuais. A emergência é frequentemente identificada em sistemas complexos, como sistemas biológicos, sistemas sociais e até mesmo em sistemas com metodologias aplicadas ao projeto generativo, onde as interações e relações entre os componentes dão origem a resultados novos e inesperados (Martino & Celani, 2014).

Existem vários tipos de sistemas evolutivos, cada um com o seu nível de complexidade na identificação e seleção das soluções. Por sua vez este tipo de sistema, por serem baseados em conceitos naturais permitem a exploração fundamentada de soluções com um toque natural. A arquitetura que sempre se inspirou da natureza pode facilmente computar soluções que abordam os conceitos biológicos no desenvolvimento da forma e espaço, como por exemplo morfogênese, bio-inteligente, bio-inspirado e bio-mimética (Menges & Ahlquist, 2011).

		Tipo de sistema		
		Sistema Paramétrico	Sistema Gerativo	Sistema Evolutivo
Propriedades	Processo algoritmo	Parâmetros baseados em regras e restrições.	Geração automática de soluções por IA ou sistemas gerativos.	Algoritmos baseados em processos naturais, de evolução genético.
	Objetivos de problem solving	Modificar o resultado segundo requisitos específicos, automatização e otimização do processo.	Exploração de diferentes soluções; Automatização do processo.	Geração de soluções otimizadas por identificação de emergências por processos de seleção evolutivos.
	Exemplo	Elementos como paredes e janelas em softwares BIM, que não permitem processos exploratório.	Midjourney para geração de imagens; Uso do Galapagos para gerar infinitas soluções; Random generator.	Machine learning: Uso de galapagos para maximizar ou minimizar um resultado, ou seja otimização do resultado final.

Tabela 1 | Tabela comparativa de sistemas abordados (Tabela do autor)

Existem diversos softwares com aplicação dos processos evolutivos, tal como é verificado nos sistemas generativos e nos sistemas paramétricos. A aplicação a metodologias de projeto e design deve ser vista como uma escolha estratégica. Os sistemas bem como os processos computacionais facilmente são usados como ferramentas de apoio na resolução do problema, logo a escolha da ferramenta deve ser feita de modo a cumprir os requisitos necessários para a resolução do problema. A implementação destes processos computacionais nos softwares CAD permitiu a obtenção de novas ferramentas para proveito do arquiteto, para além de impulsionarem o desenvolvimento de novas metodologias de trabalho técnico, exploratório e criativo.

3.3 | Modelação Paramétrica

Parametrics is more about an attitude of mind than any particular software application. It has its roots in mechanical design, as such, for architects it is borrowed thought and technology. It is a way of explore relationships. (Whitehead apud Woodbury, 2010, p.1)

Os processos inerentes à computação, foram a chave para o desenvolvimento de software e dispositivos capazes de processar informação, atribuindo ao computador um estatuto de permanência em qualquer setor laboral. Na arquitetura o computador é usado como uma ferramenta para desenvolvimento do projeto, pelo uso de *softwares* com carácter gráfico. Apesar de imprescindível em qualquer atelier de arquitetura este apresenta diferentes níveis de utilidade e relevância para a concretização do projeto final. O impacto mais relevante ou que merece uma maior atenção por parte do arquiteto, é sobre os *softwares* que permitem o desenvolvimento de metodologias de trabalho, que vão para lá das bases de computadorização. Permitindo ao arquiteto desenvolver modelos exploratórios, com aplicabilidade aos problemas inerentes do projeto de arquitetura, valorizando o pensamento computacional. Estes *softwares* são ferramentas de produção gráfica, do qual o arquiteto se apropria para facilitar o seu trabalho.

A produção é feita pela interação visual do arquiteto com o desenho, modelo ou imagem, que atualmente permitem a execução das ações que vão para lá, do simples “imitar o desenho manual” (emulador). Os processos computacionais abordados anteriormente estão na génese da modelação paramétrica. A base da modelação paramétrica passa pela formalização do modelo a partir do fornecimento de informações iniciais (*inputs*), que irão interagir com as regras pré-estabelecidos, ou seja os parâmetros, que produzem o resultado (*output*) (Mateus, Roseta e Monteiro, 2013). A utilização de sistemas paramétricos traduz-se em intenções e não no objeto final, sendo de modo geral usada em *softwares* para a produção de variantes de tipos específicos ou famílias de produtos, com a vantagem da atualização automática dos resultados (Herr & Fisher, 2001). Os *softwares* BIM como o ArchiCad e o Revit, são bastante conhecidos por facilitarem o trabalho do arquiteto na produção gráfica de modelos 3D e documentação bidimensional a partir da modelação paramétrica, embora a principal vantagem não resida nesse aspeto. A arquitetura é produzida pelo uso de elementos tipo integrados no *software* e acedidos pela interface de utilizador. No entanto este processo de trabalho não passa da computadorização, pois o arquiteto está previamente vinculado aos parâmetros existentes e apenas fornece os *inputs* exigidos, sem ter capacidade para a exploração livre de soluções, que se afastem dos elementos base pré-existentes no *software*. É importante referir que esta propriedade da modelação paramétrica, ou seja, a criação de variações a partir da mesma base, apesar de empregue nos softwares BIM, num modo geral para otimizar e automatizar o processo e a produção de trabalho, a mesma, nos softwares como o Grasshopper, Dynamo é por sua vez, o que permite o uso de processos computacionais como metodologias exploratórias a partir de sistemas paramétricos.

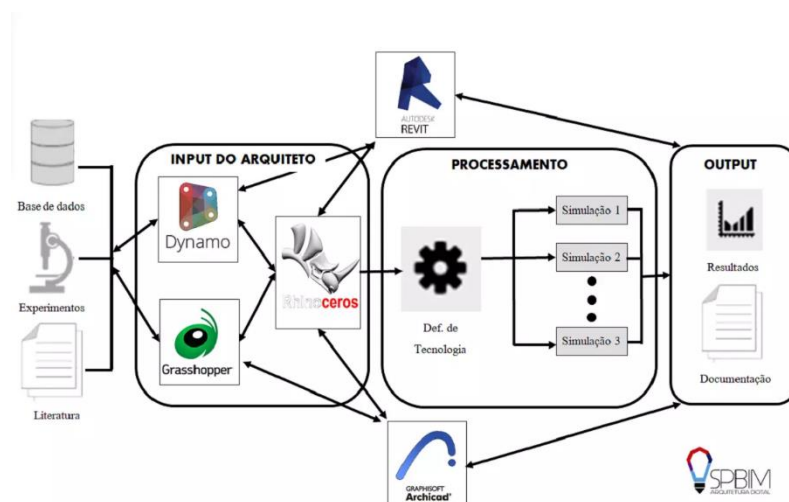


Fig. 18 | Processo de análise algorítmica (spbim.com.br)

Apesar da aplicabilidade a diversas áreas de estudo a parametrização, tal como o algoritmo, encontra a sua origem na matemática como “set of equations that express a set of quantities as explicit functions of a number of independent variables, known as “parameters.”(Stover e Weisstein, 2023).

Mathematical calculations are routinely made by the system: putting new points on the active plane, placing a line tangent to a circle through a point and specifying a point as the centroid of a polygon each model basic mathematical inferences. (Woodbury, 2010, p.33)

A matemática é onnipresente nos diversos softwares gráficos usados em arquitetura. Os CAD independentemente da sua complexidade, integram os conceitos matemáticos que permitem o processamento do “desenho” visual (Woodbury, 2010, p.11). O arquiteto não necessita de ser um matemático para o uso dos mesmos, mas para o uso das ferramentas focado em procedimentos de modelação paramétrica, este precisa de entender que os conceitos abordados na matemática e na geometria, são muitas vezes a resposta para a produção da solução. O foco não persiste em resolver e calcular as complexas fórmulas inerentes aos problemas matemáticos, mas por desenvolver um pensamento racional, que por processos de lógica permite a geração do objeto pretendido (Mateus, Roseta e Monteiro, 2013). O objetivo do arquiteto não é fundamentar as formas dos elementos arquitetônicos que propõe através de proposições matemáticas. A fundamentação da forma é apenas o benefício proporcionado pelo uso da matemática, em particular a geometria, para extrapolar com precisão o projeto gerado da mente para o papel, ecrã ou modelo físico (Woodbury, 2010, p.33).

The term “architect” means master-builder, while “geometer” can be translated as Earth-measure. This master-builder-geometer had a comprehensive knowledge of his trade, from contemporary design rules of mathematics and geometry to a profound understanding of its expression through material and structure, (...) (Ceccato, 2010, p.9)

Cristiano Ceccato (2010) faz referência aos arquitetos e estudiosos do passado como produtos do seu tempo, em que arquitetura e geometria coincidiam com a prática, marcando a sua presença durante o desenvolvimento da arte e da técnica. Os conceitos formulados inicialmente por Platão, Euclides, Pitágoras, Arquimedes e mais tarde desenvolvidos por Leonardo da Vinci, Descartes e Piero della Francesca, foram o fio condutor que permitiu uma aplicação direta ao projeto de arquitetura, atingindo o seu exponencial durante o Renascimento (p.9). A geometria facilmente associada à arquitetura Gótica, manifesta-se nas formas geradas e nas estruturas complexas das catedrais, sendo fruto dos processos racionais implementados por arquitetos de um tempo, em que estes eram obrigados a deter tais conhecimentos matemáticos (Woodbury, 2010, p.33).

Gothic buildings can be understood and were evidently designed as complex sequences of geometric construction proceeding from a few key dimensions. Traditional Persian Rasmi domes result from projecting a drawing onto a predetermined dome geometry. Palladio (1742; 1965) expounded on and (sometimes) used propositional systems in his building plans and elevations. (Woodbury, 2010, p.33)

Apesar da presença da matemática e da geometria na arquitetura não ser surpresa para nenhum arquiteto, o domínio dos seus conhecimentos tende a cair perante a vontade artística. As novas ferramentas computacionais exigem do arquiteto, conhecimentos mínimos, principalmente da geometria, para a execução do trabalho e desenvolvimento das soluções.

Mark Burry trabalhou como um dos principais arquitetos responsáveis no desenvolvimento e conclusão do projeto da Sagrada Família de Gaudí desde 1979. O seu contínuo trabalho durante 30 anos, tornou-se uma referência para a utilização de processos computacionais e das ferramentas digitais na arquitetura. As metodologias desenvolvidas passaram pela colaboração entre arquitetos, matemáticos e programadores no desenvolvimento de software e processos computacionais, de modo a decifrar a lógica por detrás do trabalho inacabado de Antoni Gaudí (Fitzpatrick, 2011). As análises aprofundadas sobre os projetos e modelos desenvolvidos por Gaudí (fig.19), revelaram o uso de conhecimentos concretos de processos computacionais com geração geométrica. Características facilmente associadas aos atuais procedimentos de modelação paramétrica digitais, mas executados por métodos analógicos (Bury, 2011, p.108).



Fig. 19 | Modelação digital de elemento arquitetónico da sagrada família (www.semanticscholar.org)



Fig. 20 | Maqueta pendular da Sagrada família de A.Gaudi (www.flickr.com)

The hanging model itself was a network of catenary systems. (...) Gaudi was working in a hands-on manner with his model-makers composing walls, columns, domes, towers and vaulted ceilings predominantly by intersecting hyperboloids of revolution. These intersected conglomerates were decoratively embellished by articulating the surface unions with triangular planes or hyperbolic paraboloids, the directrices of the hyperbolic paraboloids being designed to coincide with selected generatrices from the hyperboloids. (Burry, 2011, p.108)

Devido à complexidade natural das formas e estruturas geradas por sistemas analógicos, por Antoni Gaudí, Mark Burry foi obrigado a desenvolver metodologias e processos computacionais capazes de gerar com precisão as formas de Gaudí. Baseando muitas vezes o seu trabalho em metodologias de engenharia inversa, algoritmos genéticos e otimização de respostas (best-fit optimization). Tudo isto só foi possível pelo trabalho colaborativo com matemáticos e programadores, que foi essencial, para o desenvolvimento de métodos geométricos e procedimentos que ligam a forma computacional com a forma material (Burry, 2011).

To use mathematics is to begin with established mathematical facts and to rely on it to make a construction or, even more loosely, as a metaphor for a design move. To do mathematics is to derive theorems (new mathematical facts) by inference from prior known statements. The difference is in both intent and practice. (Woodbury, 2010, p.33)

Atualmente existe uma clara diferenciação entre o arquiteto que usa a matemática e o

engenheiro ou cientista que faz matemática. O arquiteto ao projetar arquitetura, idealiza uma forma segundo um propósito, uma função, que por vezes carece da racionalização objetiva e fundamentada procurada pelos engenheiros (Burry, 2011, p.118). A modelação paramétrica e os sistemas algorítmicos permitem uma aproximação facilitada da arquitetura racional, constituída por formas que são ao mesmo tempo a expressão objetiva da matemática e a criatividade subjetiva da arte. As metodologias de trabalho do arquiteto ganham um carácter estratégico na resolução dos problemas inerentes ao projeto de arquitetura (Woodbury, 2010, p.34-35).

Os softwares de visualização gráfica facilitam o uso, muitas vezes inconsciente, da matemática por parte do arquiteto, mesmo que este formalize o seu modelo a partir de elementos visuais, as suas ideias facilmente são desconstruídas em fórmulas matemáticas e conceitos geométricos aplicados à computação da forma. Existem diversos projectos que abordam ativamente geometrias geradas pelo computador, ao desenvolverem soluções arquitetónicas com apropriações matemáticas, por exemplo o uso de diagramas de Voronoi (fig.21) aplicados no design da fachada de um edifício. Uma abordagem à arquitetura que pode ser verificada no The Fry Building no Reino Unido (fig.22) e na Galeria Warmińska na Polónia (fig.25).

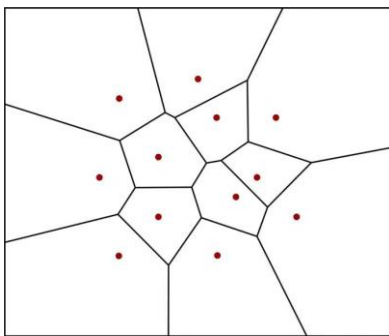


Fig. 21 | Diagrama de Voronoi
(Imagem do autor)



Fig. 22 | The Fry Building (awrn.co.uk)
(www.semanticscholar.org)



Fig. 23 | Galeria Warminska (framestudio.eu)

Deste modo a matemática, com particular ênfase na geometria, permite a compreensão da forma ou objeto final, de modo a produzir os ditos modelos tridimensionais, sendo que estes são, por sua vez, gerados por sistemas algorítmicos. O computador produz sempre o que lhe é pedido, cabe ao arquiteto saber utilizar o computador para fins exploratórios, em vez do uso exclusivamente para tratamento de imagem e produção do projeto já projetado. O foco deve passar sobre um pensamento computacional desenvolvido por processos de computação em vez de computadorização.

Deste modo a valorização, está mais uma vez na intenção e não no objeto final, ou seja, o arquiteto precisa de se abstrair do objeto final, para que possa desenvolver o processo pelas partes que o constituem. Em termos práticos, é a simplificação do objeto pela sua geometria base. Um cilindro é constituído por duas bases circulares paralelas, e por uma face que une as duas bases. Um possível processo de produção do modelo digital deste sólido geométrico, poderia partir de um círculo preenchido e de seguida extrudido (fig.26). Esta metodologia de divisão do problema em sub-partes mais pequenas e resolvidas individualmente, pode ser diretamente relacionada com o trabalho de Christopher Alexander sobre processos computacionais aplicados ao projeto. Uma metodologia que facilita a produção de modelos com maior complexidade geométrica, mantendo as suas propriedades holísticas (fig.27).

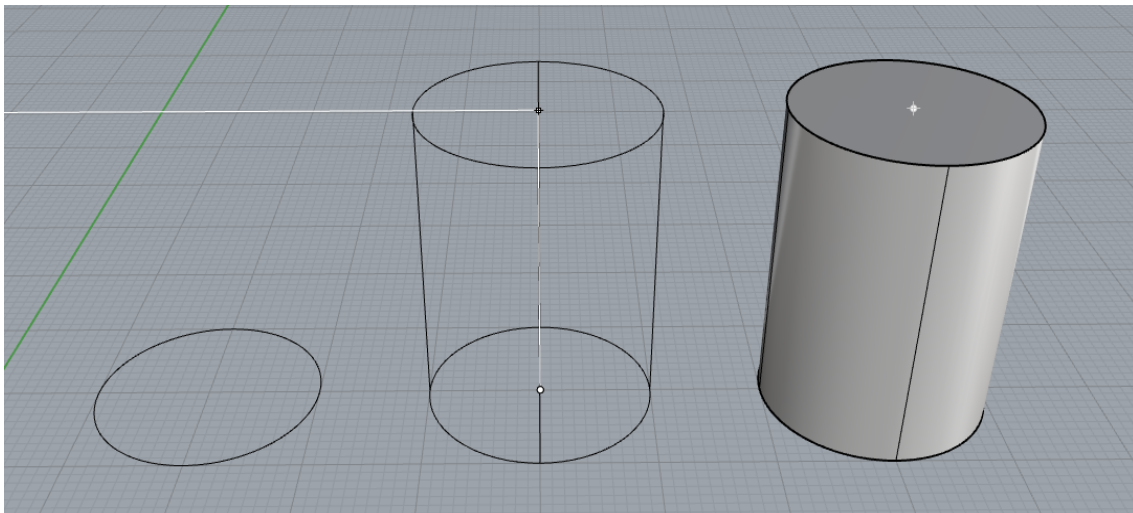


Fig. 24 | Etapas de modelação digital convencional de um cilindro (Imagem de autor)



Fig. 25 | Modelação paramétrica do projeto NEF Kagithane 03 (www.youtube.com)

A modelação paramétrica digital, passa por comunicar ao computador as peças necessárias, ou seja, os símbolos que permitem, que este produza a solução pretendida. As fórmulas e conceitos inerentes à matemática e geometria, são traduzidos em comandos através das linguagens de programação, permitindo a construção gráfica e visual da solução pretendida (Mode Lab, 2015).

O arquiteto que por base não têm conhecimentos de programação, e exhibe dificuldade em formular os necessários processos algorítmicos, vê a sua vida facilitada por softwares que permitem o uso de ferramentas de programação visual (visual scripting) (Woodbury, 2010, p.12). Uma variante da programação, que permite aos usuários programar o código (*code*), pelo uso de componentes (*nodes*) reutilizáveis que vão interagir e cooperar entre si no circuito. Estes componentes (*nodes*) são blocos de *script* que se manifestam no ecrã como elementos visuais ou gráficos que simbolizam ações, formas geométricas e fórmulas matemáticas, facilitando assim a compreensão e a criação do programa, por parte de usuários com conhecimentos e experiência limitada na área da programação (Nierstrasz et al., 2001).

As ferramentas de programação visual têm sido especialmente usadas em softwares com propósitos exploratórios, ao fornecer ao usuário a capacidade de personalizar os resultados, segundo as suas necessidades, para além de permitir a automatização de tarefas e processos, sem a utilização de programação tradicional, como a extensão Grasshopper do Rhino, o Dynamo da AutoDesk, o Node Editor para gerar ferramentas, texturas e animações personalizadas no Blender e ComfyUI para geração de imagens por Ai a partir do software Stable Difusion. Em todos os exemplos o usuário cria o programa para produção de um tipo de resultados, ao delinear as interações e relações entre os vários componentes selecionados a partir da interface visual.

A programação visual baseia-se nos sistemas à base de propagação (*propagation-based system*). Um processo computacional paramétrico simples e eficaz, aplicável a diversas metodologias de resolução de problemas (*problem-solving*). Conferindo ao programa, a atualização automática do sistema, pela interligação (*links*) dos componentes (*nodes*) que se comunicam entre si através das dependências estabelecidas (Woodbury, 2010, p.12).

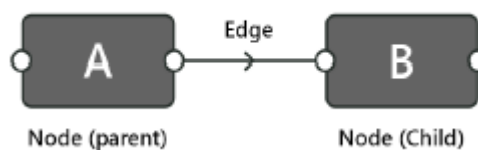


Fig. 26 | Lógica de ligação de Componentes em programação por nodes (Autor Desconhecido)

Este processo computacional é associado a um diagrama (*graph*) organizado para que o problema seja diretamente solucionado. Por sua vez, a solução vai ser o produto da ligação e relação entre os componentes (*nodes*), partindo da informação inicial conhecida (*inputs*), que se propaga pelo sistema para computar a informação desconhecida, ou seja a solução. Este tipo de trabalho deve ser tido em conta como um ato estratégico, pois o usuário têm de identificar as informações iniciais (*inputs*) necessárias para a produção do modelo. Por sua vez, o input é transmitido aos restantes componentes pelas ligações (*links*) que vão estabelecer as dependências e restrições do sistema na produção da solução (*output*), ou seja propagando o *input* para obter um *output* (Woodbury, 2010, p.12).

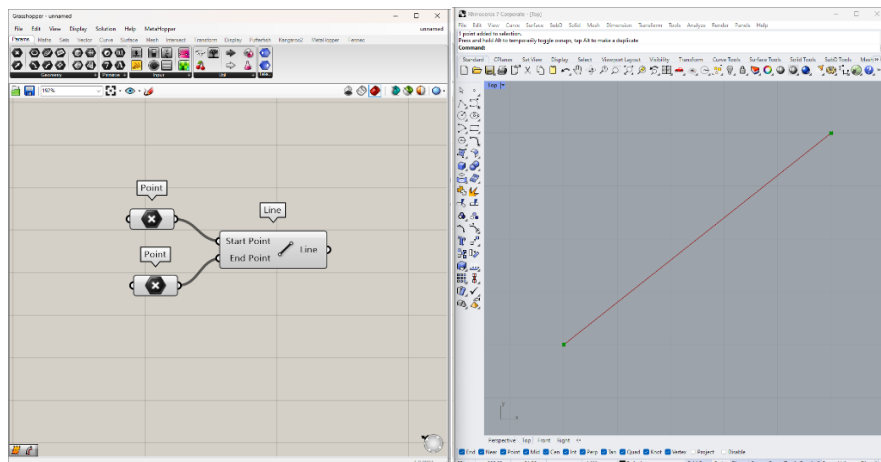


Fig. 27 | Construção de linha via Grasshopper (Imagem de Autor)

Em termos práticos cabe ao arquiteto ou usuário, definir um circuito lógico que permita gerar a solução, empregando as fórmulas e conceitos inerentes à matemática e geometria. Transformando os mesmos em processos algorítmicos, que permitem desenvolver o circuito lógico passo a passo (step by step), ou numa associação mais direta componente a componente. A produção de um modelo através de parâmetros com fins exploratórios é produto do pensamento computacional, que se abstrai do objeto final, permitindo visualizar o output como um sistema gerado por um sistema. Conforme a intenção e as necessidades do problema, os diferentes níveis de abstração no programa ditaram a adaptabilidade do sistema para utilização em diferentes situações, com foco na produção de variantes de uma solução base (sistemas paramétricos), ou de soluções alternativas geradas de um input (sistemas generativos, sistemas evolutivos) (Gordon Pask, 1969, p.495).

O arquiteto deve abordar a modelação paramétrica como um meio para produzir a sua arquitetura. Controlando a ferramenta para que corresponda às suas necessidades, executando os devidos ajustes no processo através do feedback que fornece à máquina, para que esta produza a arquitetura que vai de acordo com as expectativas do arquiteto.





4. INTELIGÊNCIA ARTIFICIAL (IA) APLICADA À ARQUITETURA



4.1 | AI - Artificial Intelligence

Globally we are going through a transition period affecting us ideologically, biologically, ecologically, technologically and spiritually. AI is a dominant theme in our headlines. Daily publications reinforce either utopian or dystopian perspectives on AI futures, as well as the perceived advantages or disadvantages AI applications will bring to humanity. (Farrow, 2019)

AI (en) ou IA (pt) é a sigla usada para identificar a Inteligência Artificial, uma disciplina da ciência dos computadores que visa replicar a inteligência humana por processos computacionais, capazes de executar tarefas que normalmente estão associadas às capacidades e comportamentos humanos e animais, como a percepção visual, reconhecimento da fala, tomada de decisão e resolução de problemas de forma autônoma (Skinner, 2012; Farrow, 2019). No entanto, o funcionamento da IA tem por base a utilização de algoritmos (generativos, evolutivos, baseados em lógica, etc) e as investigações que partem da inteligência artificial, podem ser divididas em seis sub-disciplinas, que incluem *machine learning*, robótica, visão computacional (*computer vision*), raciocínio automatizado (*automated reasoning*), percepção da máquina (*machine perception*) e representação de conhecimento (*knowledge representation*). Existem mais disciplinas derivadas do IA mas estas são as que têm uma maior influência na área e que por sua vez, ramificam-se em mais sub-disciplinas (Farrow, 2019). É a partir dos processos computacionais que os computadores adquirem o conhecimento necessário para analisar e interpretar os dados fornecidos, com capacidade de adaptação adequada a gerar previsões que permitam a tomada de decisão e execução de ações, com base nos inputs e informações computadas (Skinner, 2012).

A IA é um sistema cada vez mais recorrente no nosso quotidiano, que marca presença nas tecnologias há várias dezenas de anos, desde usos mais simples como diferenciar e filtrar emails para a pasta de spam, ou numa forma mais impressionante como os carros autónomos da Tesla e softwares como a Alexa da Amazon, com capacidade de controlar todos os dispositivos eletrónicos numa casa (Schroer, 2023). A IA nasceu da procura constante por parte do ser humano de desenvolver novas formas de processamento de informação, que possibilitam estender o conhecimento humano (Farrow, 2019).

Intelligent robots and artificial beings first appeared in ancient Greek myths. And Aristotle's development of syllogism and its use of deductive reasoning was a key moment in humanity's quest to understand its own intelligence. (Schroer, 2023)

Apesar dos primeiros indícios de pensamento de uma inteligência artificial remontarem à Grécia antiga, as origens da IA como conhecemos hoje começaram com o desenvolvimento dos computadores digitais durante a Segunda Guerra Mundial. Foi em 1942, que Isaac Asimov publicou "As Três Leis da Robótica" (fig. 28), uma ideia regularmente referenciada pela ficção científica, que estabelece os princípios para que a inteligência artificial nunca possa magoar o ser humano (Schroer, 2023). Com o avançar dos estudos durante o pós-guerra, Alan Turing e John Von Neumann nos anos cinquenta do séc. XX, apesar de não terem criado o termo Inteligência Artificial, são considerados os fundadores da

tecnologia por detrás da IA. A arquitetura dos nossos computadores modernos foi devidamente formalizada pelos dois investigadores, estabelecendo assim o seu estatuto de máquina universal com capacidade de executar instruções quando devidamente programadas para tal (COE.INT, s.d.).

No entanto, Alan Turing propôs um teste para identificar se uma máquina é inteligente, ficando conhecido como o Turing Test após publicar a investigação *“Computing Machinery and Intelligence”*. Deste modo o matemático voltava, pela segunda vez a revolucionar o mundo das tecnologias⁵, uma época marcada com a frase *“Can machines think?”* (Schroer, 2023) Turing delineou um fenómeno que denominou de “jogo de imitação” (*“imitation game”*). Neste jogo, espera-se que um participante humano distinga com clareza, no contexto de uma conversa por telétipo, se o interlocutor com quem comunica é um ser humano ou uma entidade artificial. Apesar da natureza controversa deste artigo, que para muitos especialistas não cumpre os requisitos para identificação de uma IA, o mesmo é frequentemente citado como a génese de investigações que estabelecem os pilares sobre a demarcação entre o homem e a máquina (COE.INT, s.d.).

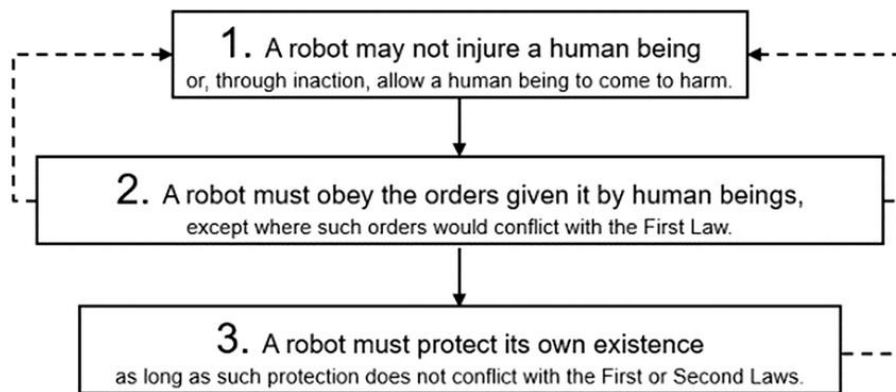


Fig. 28 | Três Leis da robótica de Isaac Asimov (www.researchgate.net)

Na mesma década, a Inteligência Artificial foi reconhecida como uma área de estudo em Dartmouth Summer Research Project on Artificial Intelligence no ano de 1956 e conotada por John McCarthy, que liderou a conferência que é hoje considerada como o local de nascimento da IA (Skinner, 2012). No entanto, a definição de IA tem variado com o tempo, consoante a perspectiva do autor.

Marvin Minsky após trabalhar com John McCarthy em 1956 define IA como *“the construction of computer programs that engage in tasks that are currently more satisfactorily performed by human beings because they require high-level mental processes such as: perceptual learning, memory organization and critical reasoning.”* (COE.INT, s.d.).

Patrick Winston, professor no M.I.T, publicou em 1977 o discurso que declara aos seus estudantes a sua visão da IA, num campo que considera em constante mudança e evolução com a entrada e desenvolvimento de novos estudos, abordagens e aplicações de IA. Winston caracteriza a IA como:

⁵ Sendo o primeiro o projeto que liderou na criação da máquina *The Bombe* (A Bomba), que descodificou a Enigma a máquina de encriptação Nazi que mudou o rumo da Segunda Guerra Mundial.

“Artificial intelligence is the study of ideas which enable computers to do things that make people seem intelligent” (Fry, 1980, p.61).

Stuart Russel e Peter Norvig (1995) abordam a IA no seu conceituado livro *“Artificial Intelligence: A Modern Approach”*, como um agente inteligente na máquina - *“the study of agents that receive precepts from the environment and perform actions.”* (Schroer, 2023).

Apesar da dificuldade aparente em obter uma única definição do que é a Inteligência Artificial, estudiosos como Wally Rhines (1985), preferem uma visão mais abstrata da disciplina predizendo um maior número de possibilidades. Rhines sugere - *“think of AI as the ‘branch’ of computing... (and) it may be wise at this point not to define it, because to define is to limit its possibility”* (apud Farrow, 2019). No entanto, apesar da vasta capacidade e variedade de programas de inteligência artificial dificultarem a definição da disciplina, Arend Hintze salienta quatro tipos de IA com base no seu funcionamento, permitindo distinguir os diferentes tipos de IA: Máquinas reativas (*Reactive machines*); Memória limitada (*Limited memory*); Teoria da mente (*Theory of mind*); Autoconsciente (*Self awareness*).

Máquinas reativas (*Reactive machines*) - representam a inteligência artificial com capacidade de perceber e responder ao ambiente diretamente à sua frente. No entanto, este tipo de IA constitui a configuração mais básica, não possuindo qualquer capacidade de armazenamento de memória, deste modo não existe forma de aprender com as experiências do passado, ou seja, sem capacidades evolutivas na tomada de decisões em tempo real. Estas máquinas possuem uma perspetiva limitada e são projetadas para realizar tarefas específicas. As máquinas reativas apresentam resultados consistentes, por reagirem sempre da mesma maneira aos estímulos para o qual foram configuradas, aumentando assim a sua credibilidade e fiabilidade. Um exemplo de uma máquina reativa é o Deep Blue da IBM (fig.27), um supercomputador conhecido pela sua capacidade de jogar xadrez. O Deep Blue era capaz de reconhecer as peças de xadrez e executar movimentos lógicos baseados nas regras pré-estabelecidas de um jogo de xadrez. Outro exemplo de uma máquina reativa, é o AlphaGo do Google (fig28), que utiliza a sua própria rede neural para avaliar o estado atual do jogo para executar as escolhas mais apropriadas. No entanto, o AlphaGo não considera as possíveis jogadas futuras durante o processo de tomada de decisão (Schroer, 2023; APD - Portugal, 2023; Hintze, 2016).



Fig. 29 | Deep Blue (phamoxtech.com)



Fig. 30 | Alpha Go (abertoatedemadrugada.com)

Limited memory (Memória limitada) - no contexto da Inteligência Artificial, enquadra-se na categoria dos sistemas de IA que possuem a capacidade de reter dados e previsões anteriores, ou seja, com capacidade de armazenamento de memória interna. Este tipo de sistemas de IA analisa as experiências passadas armazenadas na memória interna, em busca de indícios sobre os possíveis acontecimentos futuros, para formular as decisões com base nas experiências anteriores. Um sistema de IA com memória limitada, pode ser obtido por via de *machine learning* de um modelo com novos dados ou pela construção de um ambiente de IA em que os modelos possam ser treinados e atualizados de forma autónoma. Este procedimento abrange múltiplas etapas, que integram o estabelecimento de dados de treino (*establish training data*), formação de um modelo que incorpora algoritmos de *machine learning*, a verificação das capacidades preditivas do modelo, análise reactiva a feedback proveniente da informação fornecida manualmente (humana) ou do ambiente com que interage. A IA com memória limitada supera as máquinas reativas em termos de complexidade e oferece maiores possibilidades no que diz respeito ao exame e utilização de novos dados, pela sua capacidade evolutiva perante o problema (Schroer, 2023; APD - Portugal, 2023; Hintze, 2016).

Theory of mind (Teoria da mente) - um conceito que no campo da IA refere-se à capacidade das máquinas de IA de apreenderem e compreenderem as cogitações, sentimentos e procedimentos na tomada de decisão humana, animais e de outras máquinas. Isto exige que as máquinas de IA tenham a capacidade de apreender e processar a noção de “mente” e as volatilidades das emoções na tomada de decisão, em tempo real. No entanto, atualmente não existe tal aptidão tecnológica e científica, que possibilite atingir este nível de inteligência artificial. Continua a ser um conceito abstrato que necessita de mais avanços tecnológicos da IA. A base da teoria da mente parte do reconhecimento de outras entidades vivas que possuem pensamentos e emoções que influenciam a sua conduta, e as máquinas de IA precisam de ser capazes de utilizar esta informação para tomar as decisões de forma autónoma (Schroer, 2023; APD - Portugal, 2023; Hintze, 2016).

Self awareness (Autoconsciência) - no domínio da inteligência artificial diz respeito à capacidade das máquinas de IA de compreenderem a sua própria existência, bem como de discernir a presença e o estado emocional de outras entidades. Atualmente, a autoconsciência no domínio da IA continua a ser uma construção teórica, uma vez que ainda não atingimos a capacidade tecnológica e científica necessária para atingir tal nível de sofisticação da IA. A conquista da autoconsciência na IA exigiria a replicação da consciência humana por parte das máquinas, permitindo-lhes processar e compreender instantaneamente conceitos psicológicos. No entanto, é essencial reconhecer que a autoconsciência em IA é uma aspiração para um futuro, sendo constantemente abordado pela ficção-científica sem qualquer fundamento material e com aplicação prática testada (Schroer, 2023; APD - Portugal, 2023; Hintze, 2016).

O modo como a máquina funciona é uma das formas possíveis de avaliar o IA (Hintze, 2016), no entanto tendo em conta que os princípios base desta tecnologia passam pela imitação do homem, não é de nenhum modo surpreendente que este, seja catalogado e avaliado sobre esses mesmos princípios. Deste modo a IA é recorrentemente comparada e questionada pela sua semelhança ao homem, ou seja, são comparadas as capacidades humanas com as capacidades da IA, um método de avaliação que se traduz em três categorias: ANI (*Artificial narrow intelligence*), AGI (*Artificial General Intelligence*) e ASI (*Artificial super intelligence*).

ANI (Artificial narrow intelligence) - Uma variante específica da inteligência artificial com foco em executar tarefas específicas e únicas com a maior proficiência possível, podendo ultrapassar ou igualar os resultados com a qualidade e rapidez desempenhada pelo ser humano (Schroer, 2023). É frequentemente identificada em sistemas com capacidade de jogar um jogo com regras fixas como o

xadrez, em ferramentas introduzidas em software como o Gmail para identificar spam (Rius, 2023, p.9). Apesar de ser uma IA que poderá ultrapassar o homem na execução de uma tarefa específica, atualmente todos os sistemas de IA conhecidos são sistemas que se enquadram na categoria ANI por apenas serem aptos a responder a uma fração dos problemas das comunidades humanas. Deste modo por mais avançado que sejam os atuais sistemas de IA, nenhum tem a capacidade de executar tarefas com uma total diversidade e versatilidade, ou seja, uma IA construída para gerar imagens complexas de uma habitação poderá não ter a capacidade para desenhar um projeto que tem em conta todas as nuances que dizem respeito às necessidades e desejos humanos (Rius, 2023, p.9).

AGI (Artificial General Intelligence) - Constitui o avanço dos sistemas de inteligência artificial que têm a capacidade de compreender, adquirir conhecimento e empregá-lo nas mais variadas tarefas e domínios. A AGI é frequentemente comparada à ANI, que como verificada anteriormente consiste numa IA para executar tarefas ou funções específicas (Latif et al., 2023). O objetivo da AGI é imitar as capacidades intelectuais e cognitivas do ser humano. Permitindo assim, que uma máquina consiga pensar sobre um raciocínio lógico e resolver problemas de uma forma que se assemelha ao homem, podendo evoluir e aprender por si mesma com base nas suas experiências de “vida” (Rius, 2023, p.9). Este tipo de inteligência artificial é explorada em ficção-científica como nos filmes “Blade Runner” e “2001: Space Odyssey” (Rius, 2023, p.9), apesar de apresentar obstáculos reais que restringem a existência desta tecnologia, por exigir um progresso na área de machine learning, processamento de linguagem natural e computação cognitiva (Latif et al., 2023).

ASI (Artificial super intelligence) - É uma inteligência artificial identificada como o apogeu máximo desta tecnologia, superando brutalmente as capacidades humanas em todos os campos com a devida relevância. Alguns estudiosos argumentam que uma inteligência artificial que atinja o nível de AGI irá por sua vez atingir o nível de ASI, pois continuará a evoluir e a aprimorar as suas capacidades com o decorrer do tempo de uma forma autónoma, e sempre a um ritmo cada vez mais elevado. O impacto deste tipo de IA atualmente não passa de teorias por parte dos estudiosos em que alguns demonstram o seu receio sobre a mesma, no entanto é visto por muitos como a última invenção do homem, por tornar o intelecto humano obsoleto (Rius, 2023, 9).

... the first ultra intelligent machine is the last invention that man need ever make, provided that the machine is docile enough to tell us to keep her under control. (Maini and Sabiri apud, Rius, 2023, p.9)

Apesar de ser possível avaliar a IA consoante a sua capacidade em comparação à capacidade humana, alguns estudiosos da área realçam a questão de se a AI possui mente própria ou apenas imita a mente, levando à distinção entre uma IA forte (*strong AI*) e uma IA fraca (*weak AI*) (Fjelland, 2020, p.2). Uma máquina que é apta a possuir consciência, que de uma forma comum é considerada como tendo a capacidade de experienciar, sentir e estar ciente do ambiente que a rodeia é avaliado como sendo uma IA forte, no entanto qualquer IA, que se distancia dessa premissa é dada como uma IA fraca, ou seja, que apenas imita uma mente (Fjelland, 2020, p.2-3; Rius, 2023, p.10). A pertinência desta reflexão é constantemente posta em causa, tendo em conta que uma IA para passar no teste de Turing apenas precisa de ser uma IA fraca, sendo que apenas os outputs/respostas fornecidos pela máquina bastam para

que esta passe no teste. Os estudiosos que refutam esta avaliação afirmam que, concentrar os estudos nesta questão é tão relevante para a área de investigação como distinguir se um submarino “nada” ou se um avião “voa” (Rius, 2023, p.10).

No entanto um facto mantém-se, a inteligência artificial é o produto do desenvolvimento tecnológico proporcionado pelo ser humano, após séculos de trabalho, investigação e aprendizagem que permitiram a existência das atuais tecnologias (COE.INT, s.d.). A IA é interpretada como o estudo interdisciplinar de como as máquinas podem ser feitas para agir e pensar de um modo considerado como inteligente (Rius, 2023, p.3). Deste modo a máquina assemelha-se ao ser humano, em que este não nasce ensinado, é necessário introduzir informação pré-adquirida ou proveniente da experiência. Desde ANI ou AGI, *machine learning* apresenta-se como um subcampo da IA, mas que se tornou uma componente imprescindível no desenvolvimento desta tecnologia que permite diferenciar a qualidade e o propósito dos diferentes sistemas de IA. Esta necessidade de ter a informação para operar pode ser verificada na máquina que surgiu do projeto The Bombe (A Bomba) de Alan Turing durante a segunda guerra mundial. Uma invenção possibilitou fundar alguns dos primeiros princípios para o funcionamento de um sistema AI, que apesar de não ser um sistema AI, foi a primeira máquina capaz de automatizar o processo de descriptação de um código complexo. No entanto, a máquina por si só não consegue produzir uma resposta ao problema, ela necessita de uma base de dados (*data-base*). Para fácil compreensão da importância da informação no funcionamento da máquina podemos remeter ao filme “O jogo da imitação” (“*The imitation game*”), que relata o processo e os acontecimentos por detrás do projeto The Bombe. Apesar da simplificação do filme, o personagem principal (Alan Turing) encontra-se com dificuldades em pôr a máquina a trabalhar de modo a obter uma resposta ao problema, a descriptação do código Nazi, ao qual Alan Turing soluciona ao fornecer à Bomba os *prompts* necessários para que esta compreenda o que lhe é apresentado. No filme esse *prompt* passou apenas por identificar três palavras que apareciam sempre na mensagem militar, que era interceptada às seis da manhã, onde naquela época na Alemanha as mensagens terminavam com Heil Hitler, e nessa mesma hora era enviada uma previsão meteorológica que continha a palavra tempo (Tyldum, *The Imitation Game*, 2014). Apesar da realidade ser mais extensa, a ideia por detrás do funcionamento está bem representada. Só assim é que a máquina conseguia associar os símbolos apresentados com os caracteres chave, decifrando os caracteres desconhecidos por tentativa e erro ao verificar padrões no código Nazi. A descriptação da máquina Enigma apesar de ser executada pela The Bombe uma máquina, é verificada como um feito da capacidade humana, ao identificar as fraquezas do código Nazi e fornecer a informação necessária que permitiu a obtenção do output correto através do uso da máquina (Cox, 2018).

Nowadays the term AI is becoming synonymous with machine learning (ML). A machine’s ability to learn from examples and improve with experience - as opposed to relying on predefined logic rules or brute force - has only recently started to captivate industry outsiders. (Rius, 2023, p.11)

Num mundo cada vez mais baseado em informação com características próprias e proveniente das mais variadas bases de dados, passa a existir um maior foco em capacitar a IA para processar essas mesmas informações de uma forma autónoma e com a habilidade de evoluir com o tempo (Sarker, 2021, pp. 1-2). Deste modo, apesar de *machine learning* ser uma subdisciplina da IA tem vindo a desenvolver-se exponencialmente, havendo uma procura cada vez maior por entender e desenvolver esta temática

(Sarker, 2021, p.2). Este campo de estudo permitiu capacitar a máquina para analisar e computar a informação de uma maneira inteligente, possibilitando que esta possa também aprender e evoluir com as suas experiências. *Machine Learning* envolve a identificação de padrões num *dataset* (conjunto de dados) treinado, que permite obter previsões úteis e formular nova informação (Rius, 2023, p.11). Os desenvolvedores desta tecnologia baseiam-se no uso de algoritmos, para que cada sistema corresponda o melhor possível com a aplicação pretendida. Cada algoritmo é único por si só, inclusive dois algoritmos com a mesma categoria poderão fornecer diferentes respostas para o mesmo problema. Apesar da sua especificidade é possível categorizar os algoritmos usados em machine learning em quatro categorias: supervisionado, não supervisionado, semi-supervisionado e com reforço de aprendizagem (Sarker, 2021, p.2).

Supervisionado - É um tipo de *machine learning*, em que o computador é equipado com um conjunto predeterminado de *inputs* e os *outputs* correspondentes. Por exemplo, é possível fornecer um *dataset* de *inputs* que informa se o output correspondente é classificado como “alto” ou “baixo” (Rius, 2023, p.12). O objetivo da ML supervisionada é treinar a máquina para prever com precisão a saída de novos dados não identificados anteriormente, um sistema que envolve frequentemente a utilização de algoritmos de classificação ou regressão (Sarker, 2021, pp.3-4). Nos algoritmos de classificação, a saída é classificada em classes distintas. Por exemplo, o algoritmo pode categorizar um indivíduo como “digno de crédito” ou “não digno de crédito” com base em características específicas do valor de entrada. Por outro lado, os algoritmos de regressão procuram um output que forneça um valor previsto dentro de um domínio específico numérico contínuo dentro de um determinado intervalo. Por exemplo, identificar o rendimento expectável a partir de uma série de inputs com a respetiva informação (Rius, 2023, p.12).

Não supervisionado - Uma categoria de algoritmo de ML que permite examinar datasets não identificados, desprovido de qualquer tipo de intervenção humana. Isto implica a aquisição de conhecimento pela identificação de padrões e estruturas com base em procedimentos centrados em informação pré-estabelecida, sem qualquer conhecimento prévio sobre o output ou intervenção humana (Sarker, 2021, p.4). Os dois métodos de ML não supervisionado mais comuns são o *clustering* (agrupar) e o *association* (associação). Um processo por *clustering* envolve a divisão de elementos em grupos pelas características que têm em comum. No entanto, um sistema de ML com base em algoritmos de associação, vai procurar relações em comum entre os diferentes elementos (Rius, 2023, p.12).

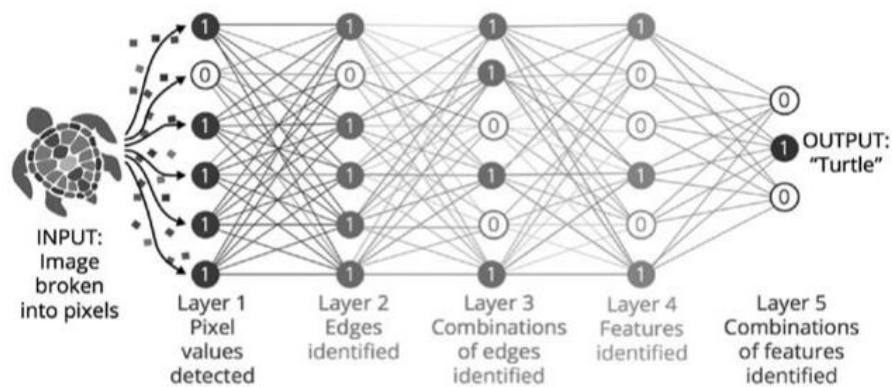
Semi-supervisionado – É a combinação de técnicas supervisionadas e não supervisionadas para lidar com dados previamente identificados e não identificados, tornando este tipo de algoritmo útil em cenários em que os dados previamente identificados são mais escassos que os não identificados. Esta abordagem é verificada nas mais variadas aplicações, como por exemplo tradução automática, deteção de fraudes, rotulagem de dados e categorização de texto (Sarker, 2021, p.4).

Reforço de aprendizagem – É um tipo de *machine learning* que adota um cenário, em que o software procura otimizar o resultado através de recompensas cumulativas e minimização de riscos, ao realizar ações num determinado ambiente. Este método de aprendizagem pode ser entendido por uma criança que brinca, sem qualquer tipo de instrutor explícito, mas que possui uma conexão sensorial com o ambiente que a envolve. Esta ligação é um meio para que a criança obtenha informações pertinentes de causa e efeito, enfrentando as consequências das suas ações e verificando os passos necessários para atingir os seus objetivos (Rius, 2023, p.12). Serve como um instrumento potente para facilmente educar os modelos de IA, que por sua vez, permite automatizar ou melhorar a eficácia de sistemas complexos como se verifica na área da robótica, em veículos com condução autónoma, produção e logística em cadeias de abastecimento. No entanto, estes algoritmos não são aconselhados para resolver problemas

básicos e de resolução óbvia (Sarker, 2021, p.4). ML por algoritmos de reforço distingue-se do método supervisionado, pois os pares *input/output* não necessitam de ser explícitos para o sistema, ou seja, o foco persiste no desempenho que evolui pela tentativa e erro, balanceando entre a exploração excessiva da informação pré-estabelecida e a procura de novas informações (Rius, 2023, p.13).

Como verificado anteriormente *machine learning* é uma subdisciplina da IA, que engloba o uso de algoritmos para melhorar tanto a experiência de uso como o desempenho do sistema de IA. No entanto, o sistema Deep Learning tem ganhado cada vez mais força e devem ser entendidos como um subcampo de *Machine Learning*, com foco em algoritmos com aprofundamento (i.e., multi-layer) de redes neurais, construídos à semelhança do cérebro humano (Rius, 2023, p.3). *Deep learning* oferece uma estrutura computacional que integra múltiplas camadas de processamento, incluindo uma camada para input, oculta e de output, para que possa adquirir conhecimento a partir dos dados. O principal benefício proveniente do *deep learning* em comparação com as técnicas convencionais de ML reside no seu desempenho aprimorado aplicável em vários cenários, especialmente ao lidar com conjuntos de dados extensos (Sarker, 2021, p.13). Alfonso Rius (2023) oferece uma explicação sobre a relação entre o termo *deep* e uma rede neural artificial:

The term “deep” (as opposed to shallow) points to the fact that the artificial neural network (ANN) has more than one hidden layer. For each layer, there are a number of neurons attempting to decode and propagate information, with each layer processing increasingly complex details. (Rius, 2023, p.13)



Source: House of Lords Select Committee on Artificial Intelligence, 'AI in the UK: Ready, Willing and Able?' (HL 2017-19, 100-I) 15.

Fig. 31 | Classe de algoritmo: Deep Learning (De Molina Rius, 2023, p.12)

Deep learning - É a metodologia de *machine learning* mais avançada da atualidade, que quando aliada a outros tipos de algoritmos consegue estender as suas possibilidades de aplicação bem como as áreas de utilização. Um exemplo de uma IA que beneficia de *deep learning* e algoritmos de reforço é o Alpha GO.

AlphaGo processed 15,000 games played by human experts and was trained with a DNN to find patterns in those games. (...) This involved the use of reinforcement learning to reward the agent for taking moves that increased its chances of succeeding. Many spectators were bewildered by AlphaGo's 'ability to grasp the full nuance and complexity of this ancient Chinese war strategy game, with its 10170 possible board positions' (Rius, 2023, p.14)

O mundo do futuro será construído com o apoio da IA, que irá ser formado pelos algoritmos de *machine learning* e processos computacionais produzidos pelos seus programadores. Esta tecnologia já faz parte da história da humanidade e deve ser encarada de mente aberta para melhor tirar o proveito da mesma.

4.2 | IA como ferramenta de produção de imagens

Independentemente de conotações ideológicas, escolhas políticas e socioeconómicas, as ferramentas digitais tornaram-se indispensáveis, e chegaram para ficar, de modo irreversível. (...) O processo em curso não é apenas o uso de uma nova ferramenta, mas é inseparável de um quadro cultural muito mais amplo. (Neves et al., 2023, p.4)

A arquitetura é constantemente abalada com as inovações tecnológicas, que são recorrentemente adaptadas de outras indústrias e áreas de estudo, proporcionando diversas alterações estruturais, almejando o encurtamento do tempo de produção, melhorar a eficácia e possibilitar a redução dos custos inerentes à produção (Barker, 2024). As tecnologias de inteligência artificial (IA) na área da arquitetura têm provocado o receio dos profissionais que encaram estes avanços, como uma substituição da mão ou mente criativa humana (Barker, 2024; Schumacher, 2023). Diversos são os projetos que só foram conseguidos com o uso das mais variadas ferramentas digitais, onde cada vez mais a valorização está no uso consciente de um pensamento computacional, em vez da simples produção de objetos (Neves et al., 2023, p.4). No entanto, esta é uma disciplina que vive diretamente associada à imagem, ou seja, aqueles que a praticam criam, projetam e comunicam a partir do uso de imagens. É importante salientar que a palavra imagem não deve ser associada diretamente a um desenho, manifestando-se de diferentes formas, sendo valorizada como uma produção com interação visual (Woodbury, 2010, p.11; Schumacher, 2023).

... I would like to remind us that a picture can speak a thousand words. From these coherent images we can create a well-articulated spatial composition, and also get input about the plastic articulation and material expression of the building. (Schumacher, 2023)

Tendo em conta a importância da imagem, não é surpresa que as inovações tecnológicas com aplicabilidade a esta indústria onde o arquiteto é um dos principais atores, culmine na procura de melhorar e inovar a produção e qualidade da imagem. Atualmente as metodologias relacionadas com a produção de imagem são tão extensas quanto a variedade de ferramentas disponíveis, que devido à sua versatilidade de aplicação, acabaram por ser transportadas do seu domínio original, para uma aplicação prática e teórica na arquitectura (Team Karwan, 2024). Como verificado anteriormente, os procedimentos digitais ganharam maior relevância nos ateliers de arquitetura a partir do anos noventa, tendo em conta o atual espectro de possibilidades, é impressionante como em apenas trinta anos “o digital tornou-se no método técnico hegemónico para o projeto.” (Neves et al., 2023, p.10). Para além dos programas de desenho e modelação interativa que encontramos nos software CAD e suas evoluções (ex: BIM), é factual que o uso predominante do digital na arquitetura passe pela digitalização das ideias preconcebidas, ou seja, desenhar o que já foi projetado (Franzer & Rastogi, 1995, p. 8), processo também designado por computadorização. Esta abordagem sobre as ferramentas digitais é perfeitamente válida, porém, verifica-se como aquém das expectativas utópicas proporcionadas pelos estudiosos dos anos sessenta (Neves et al., 2023, p.10). Um aspecto curioso tendo em conta que os objetivos iniciais das novas tecnologias já se

verificam como alcançados, ou seja, o uso efetivo das mesmas proporciona a automatização e otimização dos processos de trabalho, permitindo regularmente a redução de custos inerentes às diversas fases de um projeto de arquitetura. Este facto deveria motivar o arquiteto a procurar um uso criativo das mesmas, pondo em prática uma metodologia que beneficia do pensamento computacional nas diversas fases do projeto de arquitetura, ao seleccionar a ferramenta adequada às especificidades de cada problema, neste caso o uso de diferentes tipos de imagem com características próprias. Por exemplo, um projecto que vá iniciar obra necessita de ter formalizada as plantas, os cortes, os alçados, entre outros tipos de documentação, onde de uma forma geral nesta fase de projecto a imagem tem um carácter técnico, informativo e preciso. Em comparação com as fases iniciais em que predomina a troca de ideias com o cliente, onde dependendo da situação, o uso do chamado *render* terá maior peso e valor durante essa mesma comunicação. Assim a renderização arquitetónica afirma-se como o processo de criação de imagens bidimensionais expressando uma tridimensionalidade de um projeto de arquitetura, de modo a ilustrar uma aproximação realista ao produto final, antes deste mesmo ser construído (Autodesk, s.d.).



Fig. 32 | Exemplo de Render 1 (www.cuubstudio.com)



Fig. 33 | Exemplo de Render 2 (www.realspace3d.com)

A compreensão da proposta por parte do cliente ou investidor não deve ser negligenciada. Deste modo, este tipo de imagens tende a facilitar a “venda” do projeto ao cliente, simplificando a compreensão das ideias propostas pelo arquiteto (RealSpace, 2023). De facto, verificamos que nos últimos trinta anos os ateliers de arquitetura passaram a procurar os especialistas nas emergentes tecnologias digitais de produção de imagens, desde peritos em modelação digital e animação 3D, bem como especialistas em tratamento digital de imagem 2D (Neves et al., 2023, p.10). Estes profissionais inicialmente surgem das indústrias de vídeo e gravação, como é o caso da indústria televisiva e do cinema, bem como da indústria dos videojogos. Estas indústrias receberam um grande investimento inicial devido ao seu potencial económico, permitindo o rápido desenvolvimento destas tecnologias. Atualmente, e com maior relevância a partir da segunda década do século XXI, o currículo do arquiteto passou a integrar tais competências de renderização (Rahim, 2023). Deste modo a produção de imagens para marketing e desenvolvimento do projeto, trouxe as suas vantagens e desvantagens para os ateliers de arquitetura. A principal vantagem é o acesso cada vez mais regular a imagens que se aproximam com a futura realidade do projeto idealizado pela mente do arquiteto facilitando a venda do mesmo (RealSpace, 2023), mas por sua vez, a necessária alocação de recursos para produção destes modelos e imagens pode-se tornar bastante dispendiosa em tempo e dinheiro, em que até as alterações mais “simples” podem obrigar a reformulação parcial ou total do modelo que lhes deu origem. Não podemos também esquecer que é preciso ter acesso aos equipamentos e softwares capazes de processar tais tarefas, culminando em custos

extra que vão para além do profissional que executa o trabalho (Team Kaarwan, 2024.). Consequentemente, esta valorização abrangente de imagens que se aproximam, idealmente, da futura realidade, obriga à procura de distinção entre trabalhos, proporcionada pela emergente competição que facilmente se publicita (ex: redes sociais) num mundo de fácil acesso à informação (RealSpace, 2023, Schumacher, 2023). No entanto, arquitetos como Patrick Schumacher (2023) do atelier ZHA, assumem a emergente competição como um catalisador para a produção de melhor arquitetura, ao obrigar os ateliers a mudarem as suas práticas e visões que praticaram durante anos.

Foi então que o início de 2021 trouxe a popularização dos sistemas de IA e a exploração por parte do público comum, com isto diversas áreas foram então confrontadas com novas possibilidades de metodologias de trabalho (Neves et al., 2023, p.10). Um dos sistemas de IA que ganhou relevância nesse mesmo ano foi o ChatGPT da Open AI. A seguinte informação sobre este sistema foi fornecida pelo próprio ChatGPT 3.5 como resposta ao *prompt* fornecido pelo autor -"Faz um texto entre 5 a 10 linhas onde dizes quando foi lançada a primeira versão do chatgpt o gpt1, em que consiste o chatgpt e no final explicas que não é um sistema *open source* mas que tem o "ChatGPT Licensing Program" (2023).

A primeira versão do GPT (*Generative Pre-trained Transformer*), conhecida como GPT-1, foi lançada pela OpenAI em junho de 2018. Este modelo de linguagem pré-treinado representou um marco importante na geração de texto automatizada. O ChatGPT, por outro lado, é uma implementação específica do GPT voltada para conversas e diálogos naturais.

O ChatGPT é projetado para compreender e gerar texto de maneira contextual, permitindo interações mais humanas com máquinas. É importante observar que o ChatGPT não é um sistema de código aberto, mas a OpenAI oferece o "ChatGPT Licensing Program," que possibilita o acesso mediante pagamento, tornando essa tecnologia disponível para uma gama mais ampla de usuários e organizações, embora ainda esteja sujeita a licenças e termos específicos da OpenAI. (OpenAI, ChatGPT 3.5, 2023)

O ChatGPT é uma potente ferramenta de *Natural Language Processing* (NLP - Processador de Linguagem Natural) (Semrush Team, 2023), uma categoria de IA e ML que permite que um computador possa gerar, compreender e responder à linguagem humana (Oracle Cloud Infrastructure (OCI), 2021). Esta capacidade de processamento só é executada por *Large Language Model* (LLM - Modelo de Linguagem de Grande Dimensão), que de um ponto de vista simplificado, traduz-se num programa de computador de grande escala especializado em compreender e gerar respostas com linguagem natural, ou semelhante ao diálogo humano (Semrush Team, 2023). O ChatGPT, como qualquer outra IA, passou por um processo de treino (ML) onde foram utilizados métodos únicos de *Deep Learning*, que permitiram dotar o sistema com capacidades de interpretação de informação, que se assemelham ao processamento cognitivo humano (Abideen, 2023). Felizmente a OpenAI, empresa desenvolvedora do ChatGPT, publicou toda a investigação referente ao processo de ML da sua IA. A documentação revelou que o treino deste sistema de IA passou, de uma forma resumida, pelo fornecimento de 45 terabytes de conteúdo textual compactado para que pudesse assim ter informações referentes à linguagem, gramática e relações contextuais, permitindo a avaliação e reconhecimento de padrões para gerar uma resposta (*output*) (Semrush Team, 2023).

Com o avanço dos sistemas GPT não tardou a que surgissem soluções com fim em criar imagens a partir de *prompts*. Deste modo, em janeiro de 2021 a OpenAI lançou oficialmente o DALL-E, esta IA com acesso público generalizado foi o primeiro modelo Text-to-Image da empresa. O DALL-E é a evolução do conceito apresentado pela OpenAI em junho de 2020 que propunha a utilização de tecnologia Deep Learning apoiada pelo modelo GPT para compreensão da linguagem natural proporcionada pelo usuário em forma de *prompt*, para geração de novas imagens (Kerner, 2023). Esta IA foi pioneira e impulsionadora de novos sistemas e modelos estáveis que viriam a surgir, com base na geração de imagens supervisionada a partir de *prompts*.

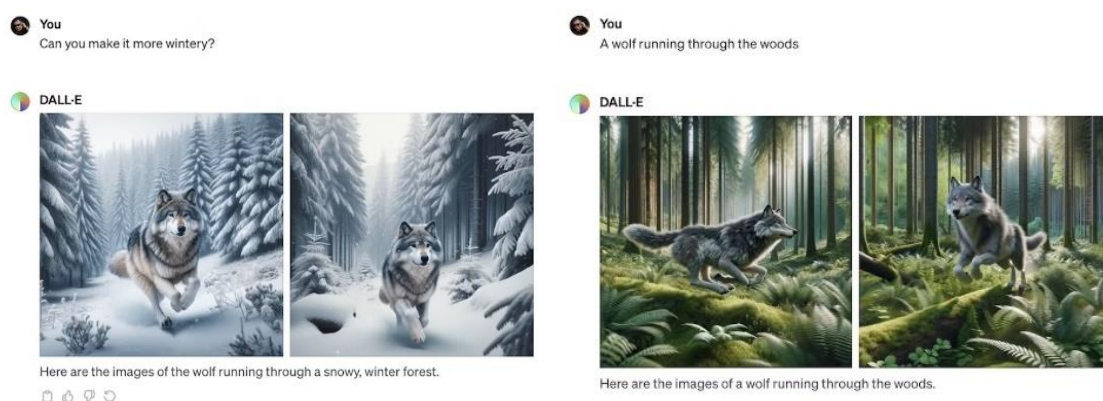


Fig. 34 | Resultados de DALL-E (zapier.com)

Deste modo existem alguns modelos que sobressaíram no mercado como o Midjourney lançado em março de 2022 (Mosele, s.d.) tornando-se um open beta em julho do mesmo ano, com acesso generalizado ao público a partir de uma subscrição, que possibilita o dito acesso ao sistema a partir de um servidor de Discord (Wikipedia, 2024). Esta IA tornou-se extremamente popular pela qualidade e nitidez das imagens produzidas, que se aproximavam de fotografias. Para além de terem treinado a IA para reconhecer os mais diversos conceitos e estilos artísticos, também permitiram a utilização e escolha de diferentes modelos de geração de imagens.

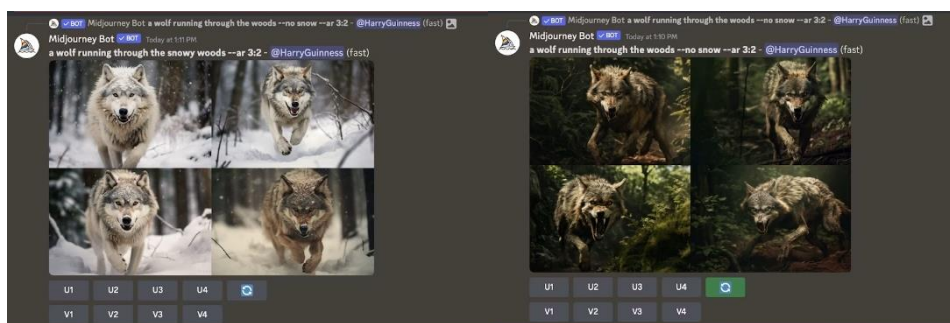


Fig. 35 | Resultados de Midjourney (zapier.com)

Um outro exemplo relevante é o Stable Diffusion desenvolvido pela Stability AI que pertence, tal como os exemplos anteriores, à classe dos Deep Learning Diffusion Models, que utiliza a matemática presente em conceitos de difusão da física para construir as imagens (Golden).

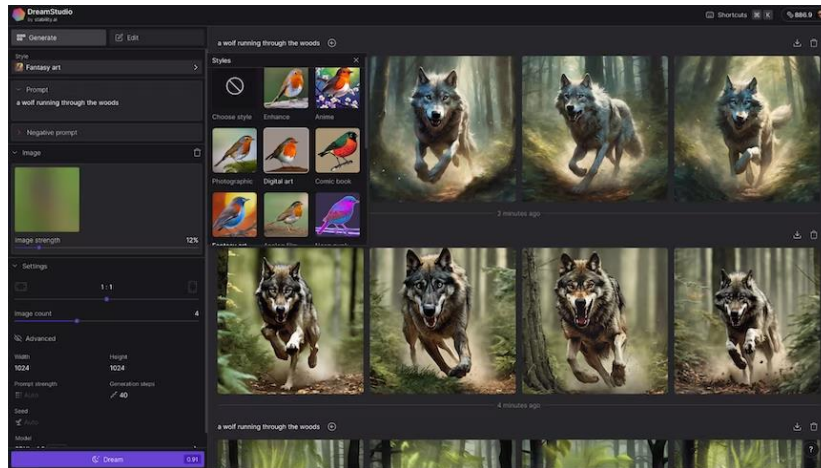


Fig. 36 | Resultados de Stable Diffusion (zapier.com)

Na sua génese, este modelo de ML, passa pela desconstrução gradual da estruturação dos dados, neste caso de uma imagem, através de procedimentos de difusão progressiva. De seguida, o modelo deverá aprender a reverter o processo de difusão progressiva, com o intuito de restaurar a estrutura dos dados. Este processo passa por treinar o modelo para eliminar o ruído Gaussiano da imagem desfocada, aplicando o processo de reversão da difusão, também conhecido como *sampling* (amostragem) (Yan, 2022).

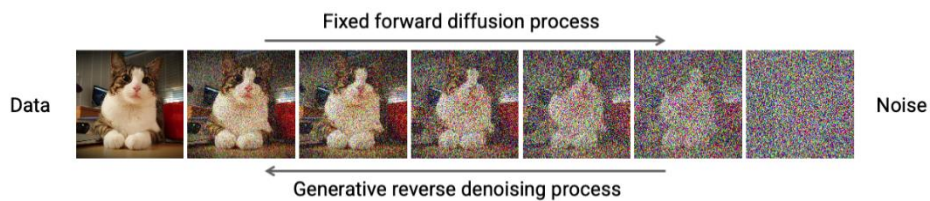


Fig. 37 | Processo de difusão vs Redução de ruído (www.linkedin.com)

Este processo permite a construção da imagem a partir do ruído, ou seja, visualmente uma imagem sem qualquer tipo de definição e nitidez, sendo progressivamente melhorada através de interações de refinamento (*steps*), de modo a apresentar ao usuário uma imagem nítida que se aproxima do *output* pretendido (Golden AI, 2024). A sequência de interações ou *steps* corresponde ao cálculo realizado pelo algoritmo sobre o coeficiente de difusão, que têm em conta as propriedades locais da imagem (gradientes e limites ou bordas) (Yan, 2022).

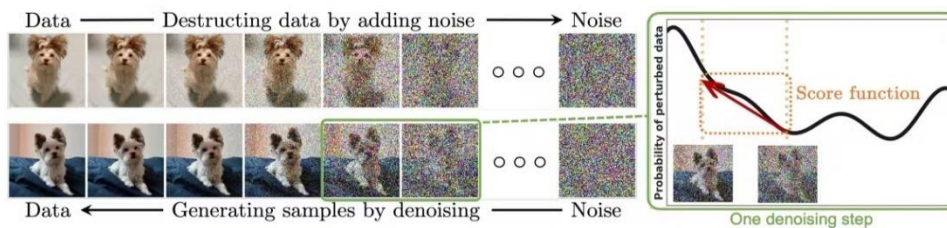


Fig. 38 | Modelo de Difusão (www.researchgate.net)

Este coeficiente é indispensável para o correto funcionamento do modelo, por determinar a intensidade e a direção que irá tomar o processo de difusão, possibilitando um ajuste dinâmico por parte do algoritmo que suaviza as transições entre as diferentes partes da imagem (Golden AI, 2024; Yan, 2022). Deste modo, é através de processos de difusão que os pixels são redistribuídos com base na informação local ou base. Onde o algoritmo diminui o ruído ao difundir os valores do pixel, nas zonas mais nítidas da imagem, para produzir texturas uniformes ao mesmo tempo que preserva a transição entre os diferentes limites dos elementos presentes na imagem. Um procedimento seletivo que permite preservar o detalhe e nitidez da imagem, para além de prevenir que o resultado esteja desfocado e enevoado com possibilidade de perda dos elementos importantes (Golden AI, 2024).

A geração de imagens a partir de um modelo de ML por difusão assemelha-se ao pontilhismo, uma técnica de pintura derivada do movimento impressionista, em que composição da imagem deriva de um processo de justaposição de pontos e manchas formando os aglomerados que produzem os elementos do desenho ou pintura (Dicio - Dicionário Online de Português, s.d.; Contribuidores Wikipédia, 2023b).



Fig. 39 | Pintura por técnica de pontilhismo (www.peonyandparakeet.com)

Os AI de geração de imagem para além da componente gerativa do sistema também integram uma componente restritiva, ou seja o texto que condiciona, informa e direciona a IA para que esta atinja o *output* pretendido. Esta condicionante é proporcionada pelo uso do CLIP (*Contrastive Language-Image Pre-training*), que sobrepõe o texto e a imagem a partir de uma layer de projeção (Yan, 2022). O CLIP foi igualmente desenvolvido pela OpenAI para demonstrar que o DALL-E é capaz de gerar o *output* pretendido. Para que isto fosse possível o modelo foi treinado com um total de 400 milhões de imagens previamente legendadas ou identificadas, permitindo que a OpenAI pudesse analisar e identificar a legenda ou *prompt* que melhor se adequa ao *output*, ou seja à imagem gerada (Kerner, 2023). Deste modo o modelo de ML aprende os conceitos visuais em forma de texto, por via de linguagem natural supervisionada e executando o chamado zero-shot (Yan, 2022). Zero-shot é a capacidade de a IA executar tarefas a partir dos conhecimentos prévios, relacionamento de conceitos e identificação de padrões (Kerner, 2023).

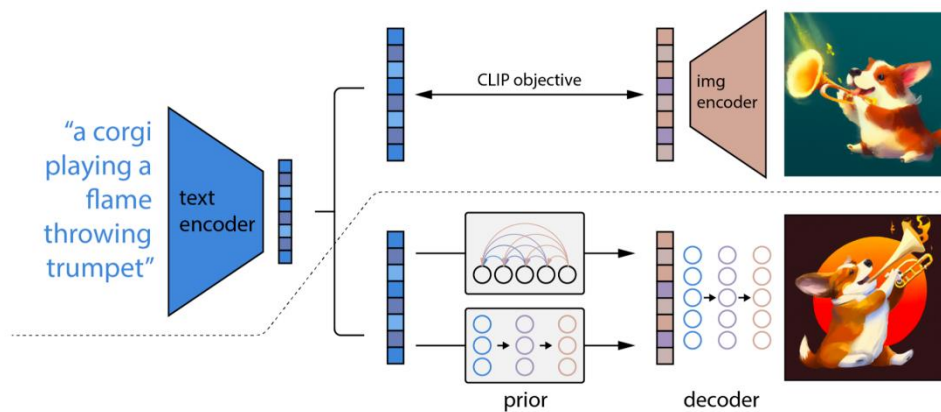


Fig. 40 | Lógica do modelo CLIP (<https://lilianweng.github.io>)

O modelo usado para treinar a IA irá condicionar e definir as capacidades do mesmo, no entanto esta emergente tecnologia não cria, gera ou produz imagens do zero, necessita sempre de ter acesso à informação para gerar o *output* com base nessa mesma informação. Este facto leva diversos estudiosos e de certo modo o público em geral a questionar a originalidade das imagens produzidas pela IA e por sua vez a quem deve ser dado o crédito como o autor das mesmas.

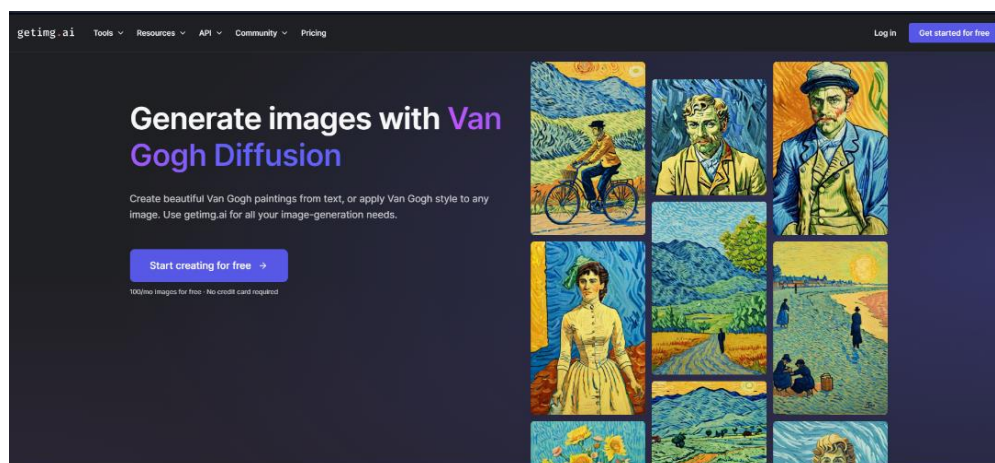


Fig. 41 | IA para geração de imagens estilo "Van Gogh" (getimg.ai)



Fig. 42 | Notícia da CNN sobre a polemica em volta da IA (edition.cnn.com)

De qualquer modo, no contexto da aplicação prática destas tecnologias no âmbito da arquitetura, estas ferramentas permitem gerar em tempo real as ideias do arquiteto, traduzindo-se na produção rápida de imagens que, para além disso, permitem ao arquiteto “falhar rápido”⁶ (Chadha, s.d.). Em adição também irá facilitar, quando necessário, a comunicação entre o arquiteto e o cliente, onde o uso destas ferramentas poderá desempenhar diferentes papéis, desde uma aproximação rápida a uma intenção do projeto, ou simplesmente uma forma de visualizar as variações e agilizar o processo de seleção durante a troca de ideias, como por exemplo na escolha de acabamentos, estilos arquitetónicos e designs de projeto (RealSpace, 2023; Team Kaarwan, 2024.; Wadhwa, 2023b). Uma mudança de paradigma em relação às tecnologias de IA está atualmente a ser verificada na produção de imagens e possibilidades projetuais, mantendo a presença do arquiteto na criação e filtragem dos resultados (Chadha, s.d.; Wadhwa, 2023b). Deste modo, a imagem em arquitetura reafirma-se como um bem essencial, que faz parte do processo mental do arquiteto no desenvolvimento do projeto, que se manifesta igualmente, como uma ferramenta imprescindível de comunicação entre o arquiteto e todos os agentes intervenientes até à conclusão do projeto (RealSpace, 2023; Team Kaarwan, 2024.; Wadhwa, 2023b).

⁶ "Falhar rápido" é uma metodologia de trabalho focada no aperfeiçoamento por feedback, ou seja, a partir da produção rápida de protótipos ou possíveis soluções, e posteriormente fazer uma avaliação crítica dos resultados irá permitir agilizar e acelerar o processo de *problem-solving*.

4.3 | IA como ferramenta de produção de projetos

“ARCHITECTURES é uma plataforma em nuvem para projeto arquitetônico assistido por inteligência artificial.” (Architectures, s.d, <https://architectures.com/que-es-architectures/>)

A ARCHITECTURES é uma plataforma com base em IA que, por meio de um sistema BIM, disponibiliza um modo de automatizar a resolução arquitetônica de programas de habitação. Com possibilidade de trabalho colaborativo em plataforma e *cloud*, transforma-se numa ferramenta de uso útil para toda a indústria AEC (Arquitetura, Engenharia e Construção). Esta ferramenta permite facilitar e automatizar o longo processo de desenho e organização de um projeto, pelo uso de um sistema paramétrico, avaliado em tempo real pela assistência de IA.

A vantagem deste tipo de sistema, está na sua reduzida linha de aprendizagem, onde os requisitos necessários para a utilização deste programa passam pelo conhecimento técnico em arquitetura e a compreensão, que não é tarefa de somenos, do que é um projeto de arquitetura. Destaca-se de outras aplicações com base em sistemas de modelação generativa, onde pelo contrário, para a realização de certas tarefas é necessário conhecimentos e a compreensão de conceitos matemáticos para a criação do sistema generativo (por exemplo o grasshopper). O programa atualmente está preparado para a criação de projetos de habitação plurifamiliares, e que no futuro terá a capacidade de auxiliar o arquiteto na realização de qualquer projeto para usos públicos (hotéis, centros de saúde, hospitais) ou privado (habitação unifamiliar). As regulamentações e as devidas normas para a validação de um projeto também são garantidas pelo sistema de IA que fornece apenas soluções para que garantem que estas sejam cumpridas, bem como uma avaliação energética do edifício.

Apesar de parecer estranho um *software* preparado para a realização de um edifício habitacional plurifamiliar não ser compatível para a realização de um edifício habitacional unifamiliar ou um hotel, a razão por detrás é bastante simples. O *software* funciona com base em geração paramétrica tal como outros programas com base nesse sistema (Revit, Archicad) ele necessita de ter esses parâmetros incorporados na sua base de dados para que estejam disponíveis na interface. Só assim é possível avaliar e efetuar o “pedido”. Ou seja, para a realização de um edifício habitacional unifamiliar é necessário que os parâmetros que definem esse tipo de edifício estejam descritos de forma que este possa ser gerado. Com isto é possível compreender que esta ferramenta apesar de revolucionar e de deter a capacidade de realizar um projeto em um “click” de distância, ela necessita sempre que alguém faça esse “click” e entenda o resultado obtido.

Esta ferramenta atualmente têm a capacidade de produzir em tempo real os diversos desenhos técnicos que fazem parte de um projeto (plantas, cortes, alçados), bem como as diversas componentes que definem o programa de um edifício habitacional (lojas, apartamentos, garagens, embasamentos) fornecendo também o mapa estrutural e o cálculo de custos para a realização da construção e estimativas de tempo, permitindo assim ao projetista informar o cliente e reduzir a margem de erro na orçamentação. Este tipo de ferramenta e a sua utilidade para o mundo da construção têm sido consequentemente criticadas e posta em causa, as problemáticas da mesma por vezes não se baseiam na sua viabilidade ou o questionar da capacidade de resposta ao problema, mas sim na ameaça de tornar obsoleta a necessidade da profissão de um arquiteto (Susskind, 2015).

4.4 | Autor? PC vs Operador

"All you need to paint is a few tools, a little instruction, and a vision in your mind." (Bob Ross, 1983 -1994)

A citação acima transcrita foi afirmada por Bob Ross nos anos 80, onde o autor apresentava um programa televisivo onde ensinava técnicas de pintura a quem assistia. Onde por contexto, torna-se evidente que as ferramentas a que Bob Ross se refere são por exemplo os pincéis que conjugados com técnicas de pintura, ou seja, as instruções que permitem ao artista produzir a imagem que transmite a vontade ou visão do artista. No entanto, devido ao seu alto nível de abstração, a mesma frase serve para descrever a base metodológica inerente ao uso dos novos sistemas generativos de IA, mais concretamente os usados para geração de imagens, em que o resultado gerado é proveniente das instruções que o usuário fornece ao software. Uma compatibilidade que aproxima o processo analógico para produção de imagens com o processo digital, visto que partilham uma lógica base em comum onde a IA é a ferramenta em que o artista utiliza e fornece as instruções adequadas, na forma de *prompts*, de modo a produzir uma imagem gerada por algoritmos.

A (r)evolução tecnológica tem abalado as mais diversas áreas de estudo, onde as consequências deste impacto das IA são ainda desconhecidas e provenientes de especulação baseada em diferentes fundamentos. No entanto, apesar de alguma compatibilidade lógica, no mundo artístico a principal questão que é levantada sobre estas ferramentas é o impacto que terão na criatividade humana, ou seja, eliminam a necessidade dos processos exploratórios de uma ideia, proveniente da automatização de resultados.

A criatividade humana é posta em causa a cada avanço tecnológico aplicável ou de possível uso no mundo artístico. Uma questão que prevalece, mas que depende do ponto de vista de quem o argumenta, onde os diferentes princípios metodológicos inerentes ao processo de criação demonstram a desumanização total da arte ou a cooperação perfeita entre o homem e a máquina. No entanto, o valor destas novas tecnologias é dado pelos processos com que estas ferramentas são abordadas, ou seja, o uso simples de produção ou para a procura exploratória, um facto que persiste e se aplica tanto às novas tecnologias de IA como ao velho (mas útil) CAD. Independentemente da abordagem escolhida pelo usuário ambas são válidas, mas o nível de criatividade ou benefício criativo que o usuário retira da utilização da ferramenta é diferente. É neste sentido que Tim Fu argumenta que a questão reside no "*how much you leave the task for the machine*", promovendo a distinção entre *low effort prompting* (*prompting* de baixa dedicação) e *high effort prompting* (*prompting* de alta dedicação). O arquiteto designer do atelier ZHA, salienta o *low effort prompting* como um processo que passa pela ideação vaga do que se pretende, passando para o preenchimento da informação textual necessária para gerar resultados e simplesmente aceitar o obtido sem exploração. A IA dá uma sensação de aumento da criatividade e produtividade (Schumacher, 2023), ao permitir a obtenção de resultados sem grande esforço, mas facilmente desvalorizados quando comparados com os processos de trabalho aplicados na emergente *Art of prompting*. Uma abordagem que se enquadra nos *high effort prompting*, onde o indivíduo (arquiteto) ou organização (atelier), aloca tempo e recursos para realmente entender a lógica por detrás do algoritmo. Sobressaindo a importância por detrás de cada decisão que possibilita o controle da máquina, com o intuito de direcionar a mesma para obter os resultados pretendidos. Onde a exploração do resultado

permite a permanência do controle criativo, num processo que começa por introduzir as propriedades base que acompanhadas de regras generativas permite a obtenção de nova informação derivada de um sistema dinâmico, ou seja um processo com a aplicação de metodologia computacional (Ahlquist & Menges, 2011a; Pask, 1969/2011; Alexander, 1968/2011). Em suma, o criticismo em volta do tema da criatividade não deve ser manifestado para questionar o valor criativo do resultado gerado, mas sim sobre o processo e experiência que levou a obtenção do resultado.

Contudo, uma das particularidades do desenvolvimento tecnológico é o aumento da velocidade de produção e o aumento da capacidade de produção, uma ideia corroborada por Schumacher numa entrevista feita por Wadhwa (2023b). Uma propriedade verificada também nas ferramentas com tecnologia de IA, que ajuda para que seja transmitida uma ideia errada, de que o processo lento e analógico é mais criativo que o processo rápido e digital. Uma pressuposição que desvaloriza as novas metodologias de trabalho aplicáveis às áreas, que ironicamente, valorizam a exploração de novas possibilidades. A área artística que inclui as disciplinas mais técnicas de concepção e desenvolvimento de projeto como é o caso da arquitetura e do design nas suas diversas variantes, acabam por constantemente recorrer ao desenho, *sketch* ou esboço como processo para a exploração criativa e apresentação de ideias, principalmente em fases iniciais do projeto. Uma metodologia de trabalho geralmente focada em processos analógicos que ocupam um lugar sagrado no portfólio destas disciplinas. Onde o autor do esboço executa uma série de interações influenciadas pelos precedentes e inspirações, por exemplo pelo uso do lápis que se movimenta no papel, na busca do resultado ideal entre as diversas variantes que cria sequencialmente. Uma abordagem recorrente e convencional, que tem o seu valor e utilidade mais que comprovado na área artística. Em contrapartida temos uma abordagem que, pelo uso da ferramenta generativa digital, permite ao usuário, a partir de um esboço base, construir uma narrativa que é produzida pela interação consecutiva com a máquina para refinamento do resultado. Um processo que permite gerar diversas variantes, mas focado no controle sobre as mesmas, ou seja, uma produção sobre a vontade humana.

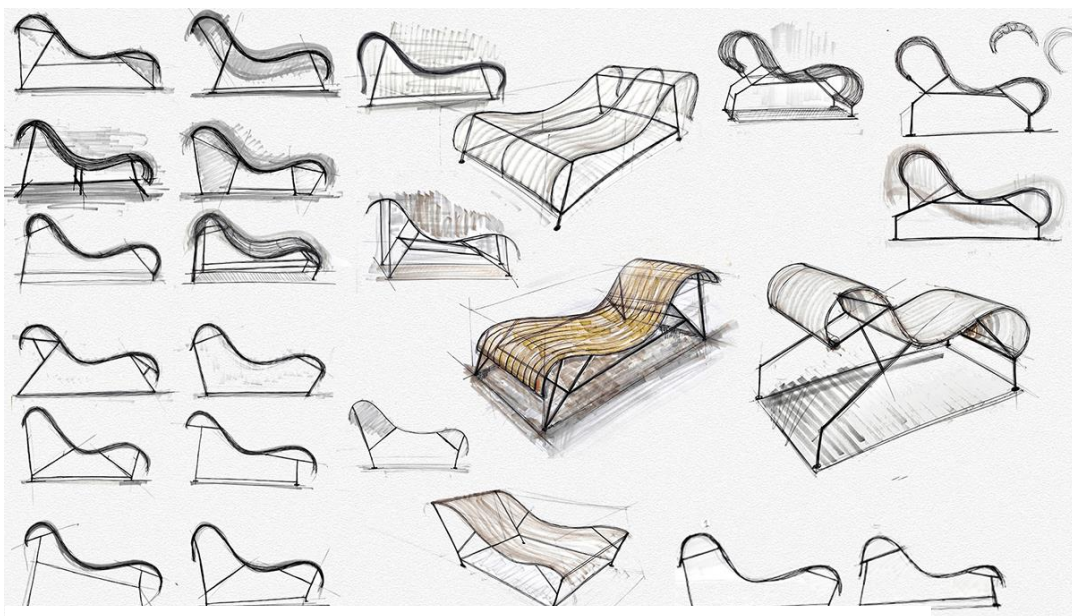


Fig. 43 | Exemplo de processo de esboço manual (www.behance.net)



Fig. 44 | Processo de sketch pelo uso de IA, feito por Tim Fu (Wadhwa, 2023a)

A comparação entre o esboço executado por processos analógicos com o esboço executado por processo digital, permite evidenciar facilmente características em comum. Por exemplo, ambos os processos partem de uma ideia base ou vontade do artista, que segue uma forma ou um sentido que a cada subsequente interação, permite uma aproximação clara do produto ou resultado final. No entanto uma comparação direta entre as diversas interações sequenciais, torna-se complicado definir qual dos dois processos é o mais criativo, devido à semelhança da lógica base por detrás do processo. No entanto, o input humano que controla e influencia o resultado, presente em ambas as abordagens, estimula e é porventura estimulado de forma diferente. Patrick Schumacher descreve este trabalho exploratório ou processo criativo como um diálogo, em que a inteligência artificial se demonstra como uma ferramenta mais responsiva e estimulante, que uma folha de papel vazia e uma série de livros com imagens inspiradoras.

Uma tecnologia que ainda é uma matéria experimental e em testagem nas diversas áreas, de modo a averiguar o seu potencial prático de utilização. No entanto, e mantendo o foco sobre a área artística em 2010, Hugh Whitehead havia escrito *“at the highest level of fluency we may yet see a generation emerge who can ‘sketch with code’”*. Expectativas expressadas no prefácio do livro de Woodbury (2010) sobre design paramétrico, que vão de encontro com a atual posição das IA. Que tal como a modelação paramétrica, proporcionam a liberdade criativa através da exploração dinâmica de soluções e variantes, com a possibilidade de transpor a metodologia base dos processos analógicos para o digital, tirando partido das propriedades dinâmicas inerentes às novas tecnologias. Estas novas possibilidades de trabalho que permitem aproximar a analogia *“sketch with code”* da realidade, como demonstrado pelo exemplo anterior (fig.42), onde Tim Fu utiliza a tecnologia de IA para esboçar a ideia, e aproximar sequencialmente a imagem com a realidade. Este é apenas um exemplo de uma série de novas oportunidades de processos de trabalho. No entanto, estas ferramentas acabam por colocar em questão diversos conceitos e fundamentos enraizados tanto na profissão como nas instituições de ensino das disciplinas artísticas. Onde ideais como *“the student who sketches well tends to do well in the studio;”* arriscam a se tornarem obsoletos por valorizarem a capacidade de desenvolver e manifestar a ideia através de um meio específico e tradicional, em vez de valorizarem a criatividade no processo e a qualidade da solução.

Deste modo, e mantendo o foco nos exemplos anteriores, o *sketch* digital não se enquadra no pré-conceito puro de um esboço produzido à mão levantada que demonstra as capacidades de desenho do artista. Inclusive, apesar do exemplo de *sketch* digital ter sido gerado a partir de um esboço desenhado manualmente e por alguém com qualidades de desenho provenientes do seu *background* na área, o mesmo processo poderia ter sido usado por um indivíduo sem qualquer tipo de habilidade para o desenho, produzindo um resultado semelhante e com a mesma qualidade. Nesse sentido, a IA expande a criatividade humana ao romper com os limites, que restringem a participação por parte de indivíduos, que neste caso não possuem habilidade para produzir imagens que comuniquem a solução com clareza. Uma condição que proporciona uma mudança de paradigma, por de certo modo, fornecer o acesso generalizado de aptidões artísticas a quem antes não as tinha. Um tema recorrentemente manifestado pelos profissionais da área artística que receiam a desvalorização do seu trabalho, provocada pela banalização das suas capacidades de produção.

Contudo devemos ter em conta que a tecnologia atual apenas possibilita o desenvolvimento de sistemas ANI, que apesar de limitados a executar tarefas específicas demonstram capacidades que igualam ou até mesmo superam a do homem, como é o caso das generative-AI que produzem resultados que superficialmente parecem finalizados. Esta tecnologia tem tido impacto sobre a geração de imagens em arquitetura, que para a pessoa comum produz resultados satisfatórios, pondo em causa a necessidade de um arquiteto para a produção de um projeto. Uma situação que causa o receio dos profissionais da área. No entanto, a profissão não se reduz apenas à mera produção de imagens criativas. O arquiteto tem responsabilidades, às quais a IA, por ter capacidades cognitivas limitadas, não consegue responder. Deste modo, o profissional da área tem de saber lidar e responder aos diversos agentes externos e internos que condicionam e influenciam o processo de conceção do projeto. Uma das muitas questões que irá diferenciar o uso destas ferramentas por um profissional que não se deixa iludir pelas imagens aparentemente finalizadas e bonitas, mas que por sua vez beneficia do estímulo proporcionado pelas mesmas. Permitindo ao arquiteto empregar os conhecimentos adquiridos através das experiências pessoais e estudo de precedentes, como critérios de avaliação e controle de resultados. Pois a imagem, conceito ou ideia por si só não têm qualquer utilidade prática se não for possível materializar as geometrias geradas pela inteligência artificial.

A IA é uma tecnologia que não passa despercebida e apesar de fomentar o receio de muitos profissionais, é reconhecida por diversos artistas pelo seu potencial de utilização e obtenção de resultados. Apesar da emergente competição que proporciona, muitos avaliam a tecnologia como um impulsionador da imaginação humana, que possibilitará a criação de novos movimentos artísticos *“whether for the pursuit of meaningful human-AI collaborations or to prove that man-made art is better than AI-generated art”* (Refik Anadol citado por Cianciolo & Chia, 2023). Deste modo, a motivação e o interesse sobre esta nova tecnologia têm possibilitado o desenvolvimento de novas narrativas, como é o caso da exposição *“Glaciar Dreams”*. Onde Refik Anadol tira partido dos algoritmos de projeção, utilizados nos IA gerativos, para gerar novos instantes a partir dos existentes, que refletem com precisão as características da informação inicial sem a repetir. Onde cada imagem gerada é única. Anadol neste projeto valorizou inclusive, a origem da informação utilizada pela IA, tendo sido proporcionada pelo próprio, ou seja, as imagens foram todas geradas exclusivamente com base na recolha fotográfica realizada pelo artista e a sua equipa. Deste modo, o autor para além de exibir o seu controlo sobre a tecnologia e o resultado, também se iliba de qualquer tipo de infringimento de direitos de autor, um tema polémico em discussão que rodeia os IA e ainda sem qualquer tipo de consentimento. Este é um exemplo que valoriza o controlo da tecnologia, tirando partido das suas vantagens e com consideração sobre as suas limitações e questões. Mantendo o arquiteto como o autor e criador do produto final por ser ele o

controlador do sistema gerativo. Um argumento que, mais uma vez, é remetido às investigações sobre a relevância da cibernética para a arquitetura de Gordon Pask (1969/2011). Onde o autor salienta o controlador como aquele que catalisa, suporta, fundamenta e toma as decisões que proporcionam o resultado.

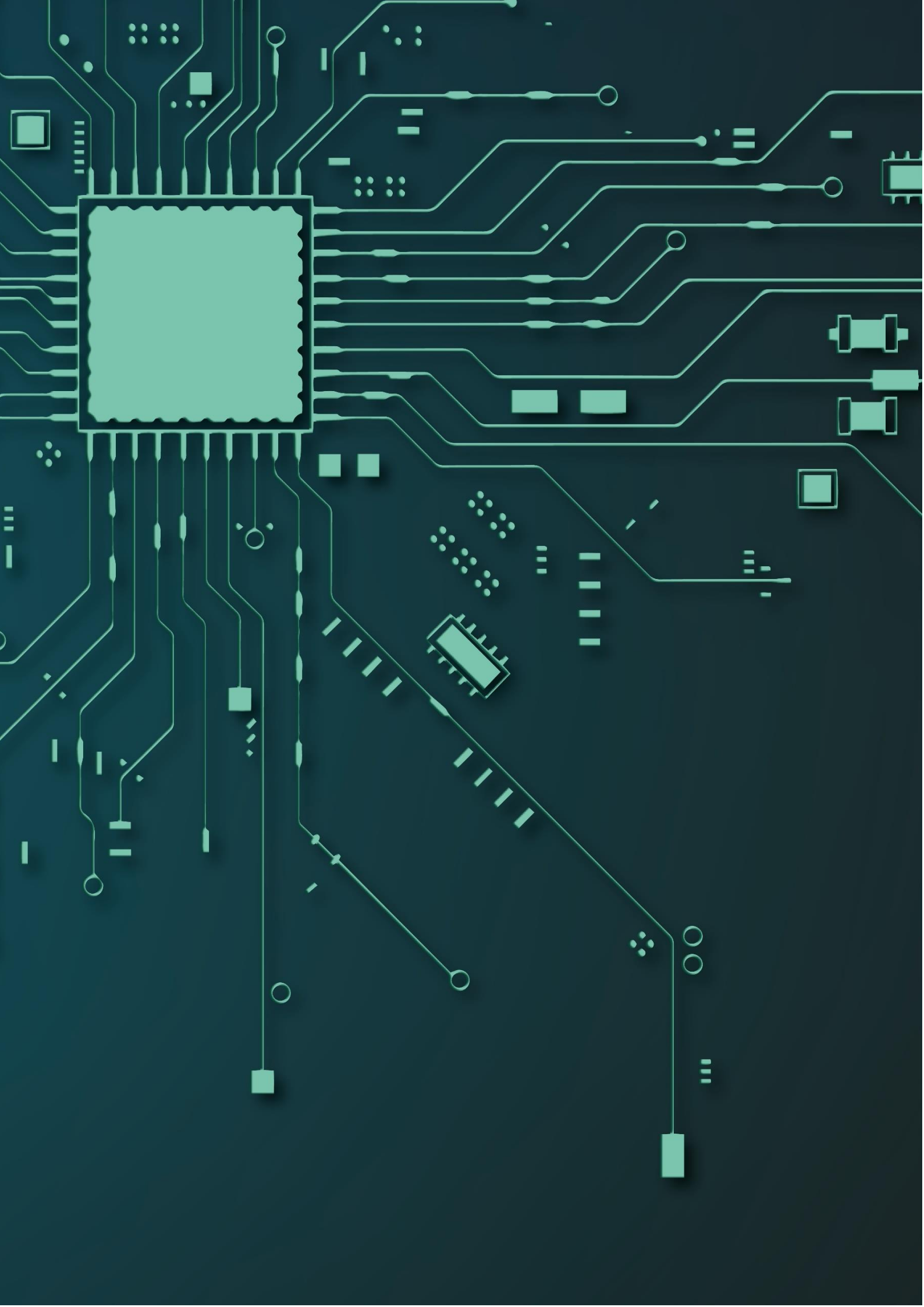
No entanto, esta questão que incide sobre quem é o autor se a máquina ou o homem, é por vezes exagerada e deve ser avaliada em contexto, pois não existem criações de puro gênio, ou seja, sem qualquer base em conhecimentos prévios. Deste modo, é defendido que a origem da informação que permite a produção do resultado pela inteligência artificial deve ser referenciada. Mas o modo como a tecnologia opera não é argumento para desvalorizar a originalidade ou inovação do resultado produzido.

“As we know the algorithm learns patterns and styles, and that's really not something you can copyright. Because imagine a world if you did. If Frank Gary's office can only do Frank Gary as design or our office at Zaha Hadid can only do Zaha Hadid style design, that would be kind of a very dystopian world...” (Wadhwa, 2023a)



Fig. 45 | Frames do trabalho de Refik Anadol (Cianciolo & Chia, 2023)

De qualquer modo, o trabalho artístico de Refik Anadol (fig.43) reflete o valor do saber controlar a tecnologia, tirando partido das vantagens que esta proporciona sem negligenciar as limitações e questões inerentes à mesma. Uma tecnologia que não passa nem passará despercebida e que merece a nossa mínima atenção. Pois apesar do impacto que terá na arquitetura, mas não só, ainda ser desconhecido e pura especulação, com certeza não será neutro. Atualmente a inteligência artificial apenas atua sobre tarefas específicas, repetitivas, entediantes, automatizáveis e dependentes de decisões humanas para uma utilização viável. Deste modo e com base em outros momentos da história da humanidade. A mudança faz parte da evolução humana e a tecnologia é regularmente o impulsionador dessa mesma mudança. No entanto, existe sempre uma fase de transição, ou seja, um tempo que leva à estabilização e uso natural da tecnologia. Um exemplo facilmente correlacionável é os CAD, que mudaram o paradigma base de desenho arquitetónico, reduzindo a folha de papel estendido no estirador, a um compacto equipamento apenas limitado por quem o usa.





5. O CASO DO ATELIER TT

5.1 | Fluxo de trabalho no Atelier

O atelier TT ou Tomás Taveira e Associados foi fundado em 1966 pelo arquiteto Tomás Taveira. Um homem que marcou presença na arquitetura portuguesa, reconhecido pela sua arquitetura pós-modernista e pela exploração da cor em diferentes elementos das suas obras. Uma característica facilmente identificada em projetos (fig.44) como a estação de metro das Orlas a sede do BPN e as torres das Amoreiras, identificada como a principal obra de referência do arquiteto e um dos marcos da cidade de Lisboa. Um atelier que apresenta um portfólio diversificado e extenso, produto dos vários anos no ativo.



Fig. 47 | Obras de referência do arquiteto Tomás Taveira: a) Torres das Amoreiras (imperdivel.pt); b) Sede do BNU (pbase.com); c) Estação de metro das Orlas. (www.semanticscholar.org)

Em 2022-2023 o atelier contava com uma equipa de seis colaboradores, para a produção de todo o material inerente aos diversos projetos em curso, sob a coordenação e ideação do arquiteto Tomás Taveira, que como o elemento responsável pelos projetos era quem tinha a decisão final. No entanto, o arquiteto, possibilitava e incentivava a exploração de ideias e soluções pelos diversos membros da equipa, em que dependendo da situação havia maior ou menor liberdade criativa para se distanciarem da ideia base do arquiteto Tomás Taveira. Um trabalho exploratório com base na comunicação entre o arquiteto e a sua equipa que traduziu, em grande maioria, na criação e produção de modelos digitais tridimensionais. Um processo com base no uso do computador e *softwares* CAD bastante valorizado pelo arquiteto TT, que sempre demonstrou apreço e interesse sobre as ferramentas digitais e na integração das mesmas no processo de trabalho do atelier. Neste caso, o modelo digital era valorizado por permitir o aproximar da ideia com a realidade, facilitando a avaliação visual das soluções propostas e a apresentação das mesmas aos clientes, para além da exploração de ideias e elementos arquitetónicos sem limites formais. Um processo de trabalho recorrentemente adotado no atelier, por corresponder com as necessidades de trabalho, que por sua vez, compensam a alocação dos recursos disponíveis, dentro das capacidades do atelier, para a produção dos modelos digitais.

O SketchUp era a ferramenta disponível para produzir os modelos tridimensionais. Este *software*

é recorrentemente utilizado em arquitetura, mas não só, por ser prático e de fácil utilização quando comparado com softwares com maior complexidade, como é o caso do Rhinoceros e do 3ds MAX. Inclusive, diversos estudantes de arquitetura acabam por iniciar a prática da modelação 3D nesta ferramenta, devido à semelhança da ferramenta com os softwares CAD, como o AutoCAD. No entanto, só o SketchUp não bastava para aproximar o modelo com a realidade, apesar do software disponibilizar ferramentas para renderização de imagem, esta não é a função principal para o qual foi criado. Deste modo, era utilizado o TwinMotion para fazer a renderização dos modelos criados em SketchUp. Mais uma vez, este é um software prático e de fácil utilização, utilizado em arquitetura, mas não só, para produção de renders e animações. Um *software* disponível para qualquer pessoa, com uma licença de uso grátis desde que não seja utilizado para fins comerciais, deste modo, não é surpresa que este seja utilizado por diversos jovens arquitetos, mas não só, na produção de imagens de síntese. Uma ferramenta que desempenhava duas funções no trabalho de atelier. A primeira era que pelas imagens criadas ajudava o cliente a visualizar o resultado final e a entender o trabalho em que está a investir, que por sua vez facilitava a venda do projeto, a partir de uma apresentação mais interessante e cativante, comparado com as plantas, cortes e alçados que, à maior parte dos leigos, pouco ou nada dizem. A segunda é que esta ferramenta permite estudar texturas, cores e o impacto dos pontos de luz artificiais e naturais, atuando como um simulador.

O trabalho exploratório e de carácter mais criativo, por mais valorizado que seja na profissão, não é o único campo no qual o trabalho de atelier atua. O arquiteto deve ser tanto um artista como um técnico, para poder elaborar e projetar os seus edifícios. Deste modo, no atelier TT eram utilizadas ferramentas CAD, neste caso o ZWCAD, para produzir os necessários desenhos técnicos, como plantas, cortes, alçados e pormenores construtivos dos elementos arquitetónicos idealizados. A equipa, para facilitar o seu trabalho colaborativo, aproveitou-se do método de hiperligação de ficheiros por XRef (referências externas), que permitiu aliviar o peso dos ficheiros e aumentar a produtividade. Implementando uma espécie de trabalho em plataforma digital, a partir do servidor privado do atelier ao qual os computadores estavam ligados. Uma mudança no modo de trabalho e organização do atelier, inspirada nas metodologias de trabalho colaborativo integradas por exemplo em *softwares* BIM, como o Revit e o ArchiCAD. Este novo sistema obrigou a reestruturação organizacional de trabalhos que já estavam em curso, e a aprendizagem de novas ferramentas presentes no *software*. No entanto, após a fase de transição verificou-se um aumento da produtividade por permitir o trabalho em simultâneo de vários colaboradores, que desenvolviam as suas tarefas em ficheiros secundários, organizados por camadas e diretamente conectados ao ficheiro principal ou ficheiro mãe (fig.45), que era gerido por um elemento da equipa.

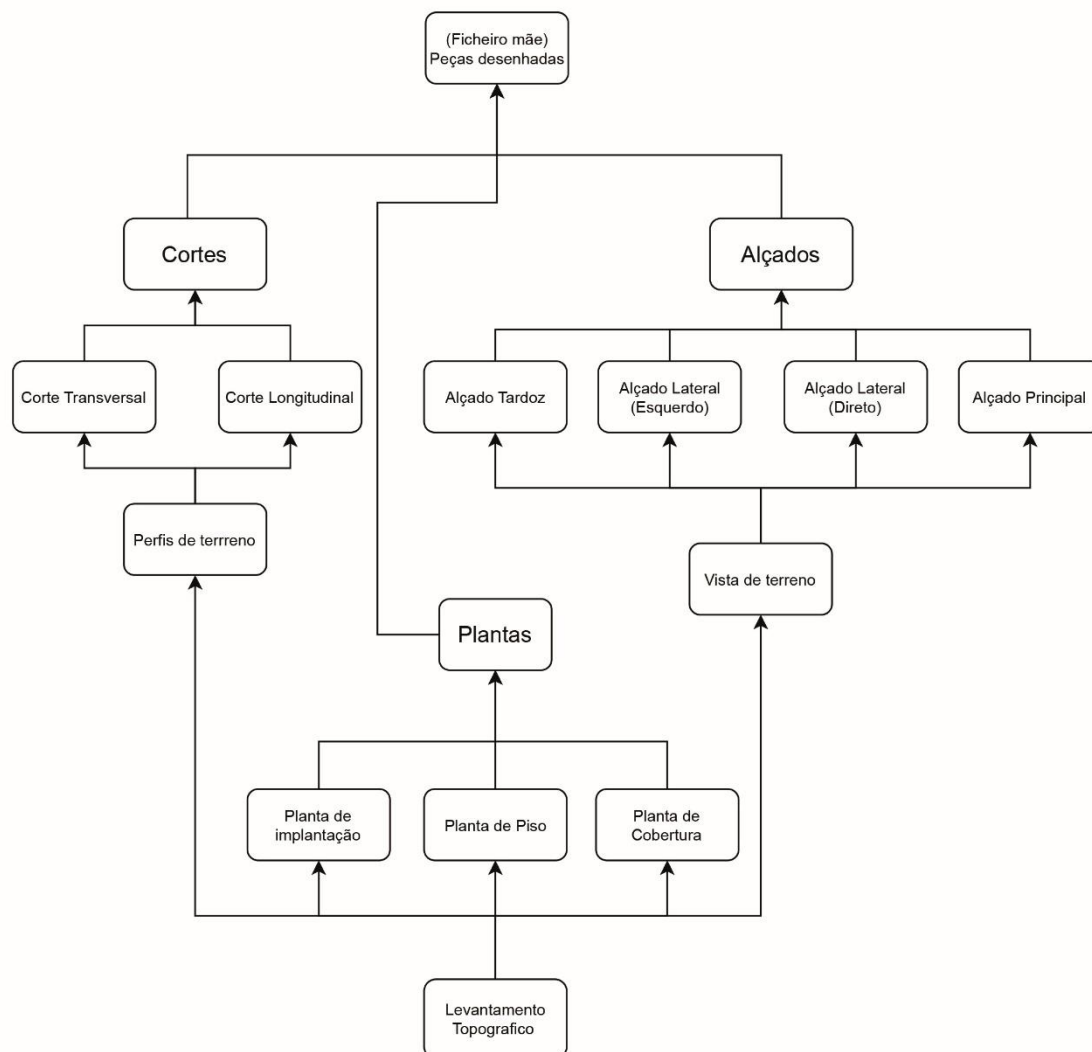


Fig. 48 | Organograma de lógica de trabalho com xRef implementada no atelier TT (Imagem do autor)

A necessidade obriga, por sua vez, ao desenvolvimento de novas abordagens, dentro dos recursos disponíveis para a resolução de um problema. A implementação do trabalho em “plataforma digital”, proporcionou uma nova metodologia de trabalho, que resolveu as dificuldades do sistema organizacional do atelier. Os recursos disponíveis são um factor muito importante no desenvolvimento tanto de um projeto como do atelier, um aspeto importante que devemos ter em conta sobre as ferramentas digitais. Onde o interesse ou a apreciação sobre as mesmas não é suficiente. Pois o trabalho é produzido por uma máquina que necessita de cumprir certos requisitos mínimos, para processar os complexos trabalhos sem qualquer tipo de paragem ou bloqueio do *software*. No entanto, devemos ter em conta o factor humano, ou seja, o utilizador ou operador da máquina que precisa de ter o conhecimento para a utilizar, que por sua vez pode obrigar a alocação de recursos para instrução dos trabalhadores. No atelier TT, os equipamentos disponíveis não se enquadram dentro dos requisitos necessários para produzir um trabalho otimizado e rápido. Tarefas como renderização de imagens eram limitadas pelos equipamentos disponíveis, ou seja, o computador não tinha capacidade para produzir

imagens com alta definição, para além dos constrangimentos causados pelo bloqueio de *software* e velocidade de processamento dos resultados pretendidos, que reduziram a produtividade do atelier. Uma realidade experienciada em primeira mão, que evidencia o peso que os recursos disponíveis têm no processo de trabalho.

O computador e os diversos softwares utilizados no atelier, permitiram o desenvolvimento de resultados, desenvolvidos inconscientemente por processos de projeto computacional, bem como por processos que não passaram da computadorização e pura representação digital de ideias previamente formalizadas. Mais uma vez, o contexto evidencia-se como um fator importante na avaliação das ferramentas digitais, ou seja, o ambiente no qual a ferramenta é inserida e o modo como é utilizada, irá ditar a relevância da mesma para a realização do trabalho de atelier. Deste modo, foram selecionados cinco casos de estudo, referentes a diferentes momentos de trabalho, experienciados em primeira mão no ambiente de atelier. Cada caso remete a um problema específico, que foi resolvido através de metodologias de trabalho mais convencionais. No capítulo seguinte experimentam-se e propõem-se processos alternativos para a resolução do mesmo tipo de problema. Cada caso contempla uma série de características próprias que proporcionaram as estratégias adotadas para a resolução dos problemas em questão, bem como as adversidades e limitações dos processos de trabalho executados. Apesar de delineados para a especificidade dos problemas em análise, facilmente verificamos pontos semelhantes entre os diversos processos, uma característica inerente dos métodos de trabalho pré-estabelecidos no fluxo de trabalho base do atelier.

5.2 | Estrutura de avaliação e enquadramento de casos de estudo

A estrutura de análise de cada caso em estudo segue a seguinte ordem:

- **Identificação do caso em estudo** - Designação dada ao caso em estudo que remete ao tipo de objeto produzido.
- **Introdução** - Um breve enquadramento do objeto em avaliação, onde é evidenciado o contexto de trabalho em que foi realizado e as intenções conceptuais da solução produzida, bem como o problema arquitetónico a resolver.
- **Caraterísticas formais pretendidas** - Avaliação das propriedades físicas do objeto, geralmente pela desconstrução da forma em elementos simplificados, acompanhada do tipo de exploração pretendido evidenciado pelas características priorizadas.
- **Processo de trabalho e resultados** - Descrição e explicação da lógica do processo de trabalho realizado em ambiente de atelier na resolução da problemática, demonstrando por imagens os resultados obtidos, sem focar na explicação exaustiva dos procedimentos executados, ou seja, o passo a passo do processo. No entanto, quando pertinente este tipo de explicação foi feita.
- **Dificuldades do processo** - Conclusão que salienta as dificuldades e limitações do processo colocando em evidência aspectos sobre os quais o processo de trabalho alternativo deve incidir.

Os casos em estudo fizeram parte de diferentes momentos da conceção dos seguintes projetos:

- **Projeto Maia Tech (Casos de estudo 1-3 e 5):** O Maia Tech foi uma proposta de projeto para a zona industrial da Maia, que promovia a construção de um edifício destinado a albergar comércio e/ou serviços. Um projeto focado em quebrar a monotonia estética diretamente ligada à função, típica das zonas industriais onde predominam edifícios intrinsecamente ligados à função. Este projeto distingue-se da envolvente pelas suas fachadas, onde predomina o uso de aço corten numa camada exterior que altera a sua configuração e padrão, consoante o espaço interior.
- **Projeto Escola de Dança (Casos de estudo 4-5):** Um projeto que visa a ampliação e reabilitação de um edifício existente para criar uma Escola de Dança. O conceito central deste projeto é traduzir a fluidez e a energia da dança na conceção dos espaços. Uma arte reconhecida pelo dinamismo e expressividade, que serve de principal inspiração na criação de um ambiente que fomenta a criatividade e o bem-estar dos seus utilizadores. Equilibrando a funcionalidade dos espaços destinados ao ensino e à prática da dança com uma estética atraente que rompa com a monotonia. Uma obra que parte da ampliação do edifício existente, valorizando-o pela reabilitação e restauro de espaços e elementos arquitetónicos de modo a acomodar o seu novo carácter de utilização, que integra o antigo com o novo.

5.3 | Casos de estudo

01 | Caso de estudo: Processos de produção de estudo prévio

Introdução

Numa fase inicial de conceção do projeto o edifício foi abordado de diferentes maneiras, havendo um processo exploratório sobre os possíveis volumes durante a produção do estudo prévio.

A primeira proposta consistia na criação de dois edifícios autónomos, cada um com configurações próprias que permitia a existência de diferentes tipos de espaços destinados ao coworking e espaços de lazer, como um estabelecimento de restauração e bebidas no rooftop e um lounge comum.

Na segunda proposta pôs-se em hipótese um projeto que abordava o programa pela fragmentação dos espaços em 5 edifícios individuais com diferentes configurações, como número de pisos, forma base, espaços exteriores, coberturas acessíveis e/ou varandas.

No entanto, optou-se por uma terceira hipótese solicitada pelo cliente devido à uma mudança no programa do projeto, onde era criada uma variante da primeira proposta, unindo os dois edifícios autónomos através de um elo de ligação, de modo a maximizar dentro dos parâmetros urbanísticos definidos pelo PDM.

Caraterísticas formais pretendidas

A segunda proposta de projeto ao abordar a problemática pela fragmentação do programa em edifícios individuais, obrigava a existência de espaços exteriores, alternados entre momentos de percurso e espaços verdes que harmonizam o afastamento entre edifícios. Estes momentos abertos derivam do posicionamento dos volumes irregulares dentro dos limites do lote, onde a forma dos espaços deriva do *offset* da planta de implantação dos edifícios. Os edifícios podem ser entendidos como sólidos geométricos que pela adição ou subtração de peças, produzem os volumes irregulares que possibilitam a existência de um projeto com diferentes vivências. Como a subtração do interior de um cubo permite a existência de um pátio, ou a sobreposição de uma cópia de um volume de proporções reduzidas sobre o volume original, produz uma varanda circundante no edifício. Toda uma série de possibilidades focada na diversidade de volumes.

A terceira hipótese e final compreende a adaptação de uma forma existente, de modo a corresponder com os valores otimizados pelo aumento das proporções originais. No entanto, essa adaptação por aumento de proporções não resolve o projeto, apenas permite a obtenção do volume que integrará o programa solicitado.

Processo de trabalho e resultados

Na primeira abordagem começou-se por estudar a malha urbana e definir a forma como o edifício se ia relacionar com a mesma. Foi então proposto dois volumes que se destacavam da envolvente, criados através de processos manuais de modelação 3D. Um trabalho realizado a partir de ajustes por

interação constante que atuavam como o feedback do controlador sobre o objeto a cada resultado. O projeto foi assim conceptualizado de fora para dentro e correspondendo ao programa pedido.

Ainda na fase de estudo prévio foi proposta uma fragmentação dos dois edifícios em cinco, que iriam albergar diferentes espaços com diferentes tipos de uso. Esta solução foi abordada apenas e só pelo uso do computador através de processos manuais de modelação tridimensional, que se assemelham aos processos de desenho convencional, onde a intuição neste caso do arquiteto Tomás Taveira em colaboração com o colaborador que executava a tarefa gráfica, gerava as formas pretendidas. Nesta abordagem foram testadas uma série de volumetrias com diferentes configurações e posições dentro da área de intervenção. Estes volumes foram surgindo ao longo do processo de exploração da ideia, a partir de sólidos geométricos convencionais alterados por processos de subtração e/ou adição das figuras originais, proporcionando o desenvolvimento de possíveis hipóteses de projeto. O resultado provém dos ajustes manuais do modelo criado, onde o processo exploratório é realizado a partir das recorrentes interações do sujeito com o objeto, na procura de uma solução ideal por tentativa e erro.

Apesar de, como referido anteriormente, a mudança do programa ter proporcionado a adaptação da primeira solução, para melhor corresponder com os requisitos de projeto do cliente. Onde o foco estava sobre a maximização da área de implantação do edifício. Para tal foi necessário escalar a forma anteriormente concebida, de modo a esgotar a área de implantação disponível. Um processo realizado em ZWCAD, onde por tentativa e erro escalou-se o perímetro da forma, de modo a otimizar o projeto. Uma tarefa simples, mas que para atingir o resultado pretendido, obriga à recorrente repetição do comando *scale* (escalar) e a consequente medição da área para confirmar a área de implantação obtida.

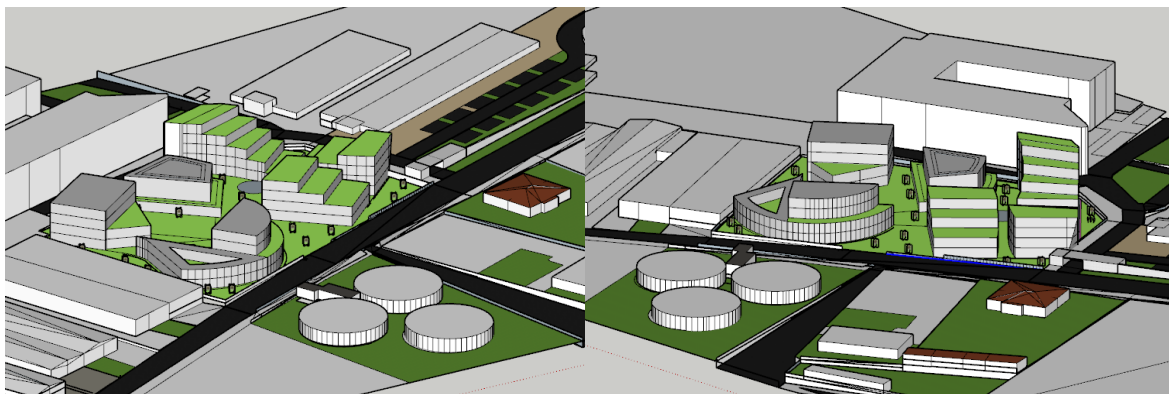


Fig. 49 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Modelo original em Sketch Up.

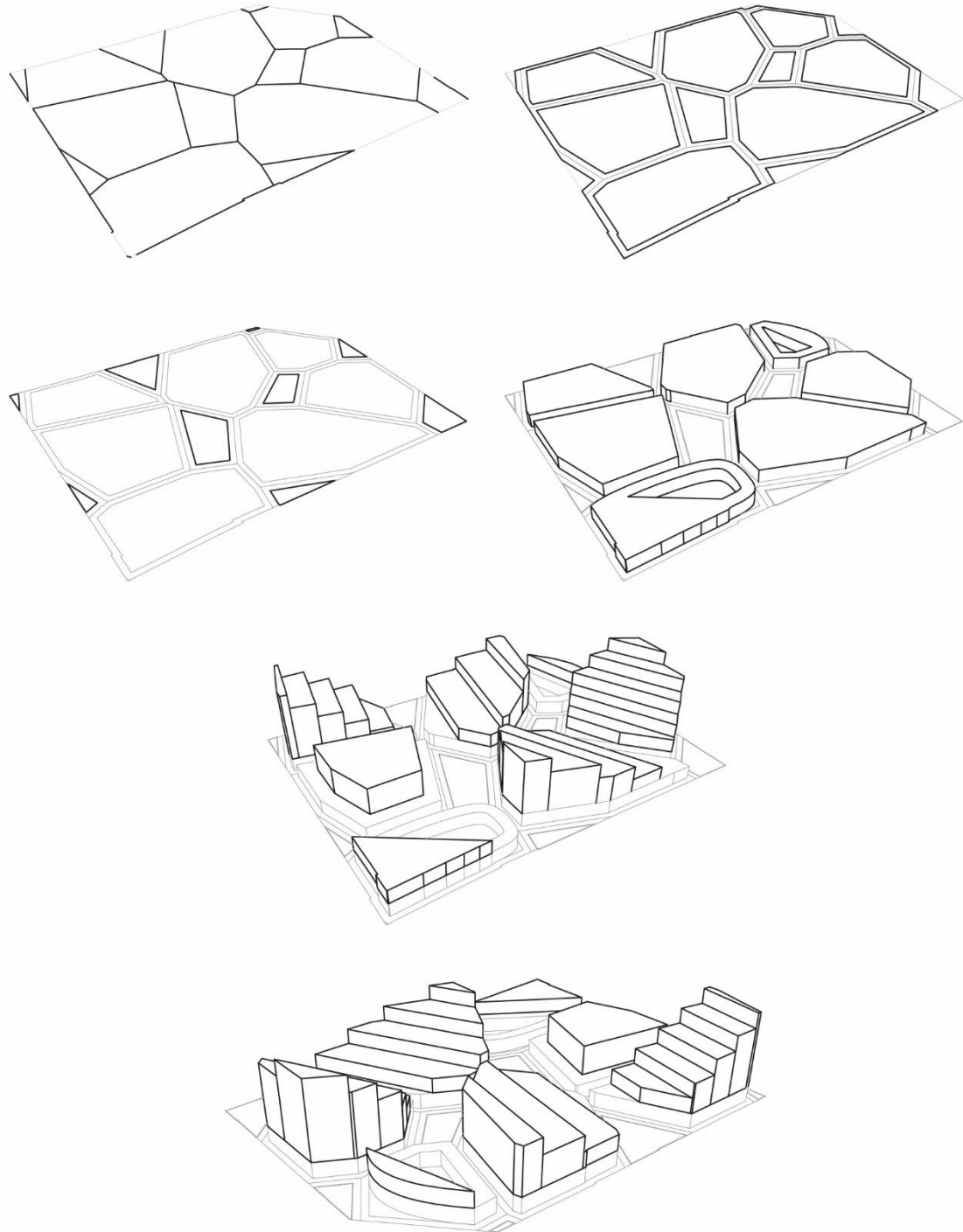


Fig. 50 | Processo de construção de caso em estudo (Imagem do autor)

Dificuldades e limitações do processo

O caso de estudo em análise apesar de abordar a mesma fase do projeto, salienta-se em dois momentos de trabalho. O primeiro momento parte da concepção de um projeto fragmentado em diversos edifícios, onde a exploração da ideia é executada por vários processos manuais com o intuito de obter uma hipótese de projeto, gerada pelo filtrar dos resultados obtidos com os devidos ajustes. Um processo de trabalho limitado e lento, que carece da versatilidade necessária para a produção rápida de variantes de soluções. No entanto, este tipo de processo exploratório onde o resultado provém do processo mental de quem o executa, é comum em fases iniciais de projeto, em que ainda não existe uma ideia bem consolidada.

O segundo momento é a otimização de uma solução segundo parâmetros específicos, que neste caso se refere ao aumento da planta de implantação da proposta até que esta atinja o valor pretendido. Um problema deveras simples de entender com uma resolução fácil, mas prejudicada por um processo lento e manual de ajuste por tentativa e erro, com confirmação dos resultados gerados. Um processo de trabalho repetitivo e entediante para atingir uma meta bastante objetiva e óbvia.

02 | Caso de estudo: Variante de fachada 1

Introdução

A fachada principal do Maia Tech que foi proposto ao cliente a fachada cortina em análise. Possibilitando a existência de varandas ou espaços exteriores semi protegidos, como conferia ao edifício um design distinto da envolvente. Deste modo, com a premissa de incorporar no projeto uma fachada cortina valorizada por soluções construtivas em metal. O arquiteto TT recorreu a designs de fachada que havia testado e experimentado em outros trabalhos anteriores a este, mas que infelizmente não tiveram a oportunidade de se materializarem em ideias finais. Foram assim postas em análise uma série de inspirações e soluções arquitetônicas, que dentro do entendimento do arquiteto TT, melhor se enquadram no programa do projeto do Maia Tech. A amostra de soluções passou por uma análise e discussão em equipa, de onde foram selecionadas as duas opções mais viáveis de serem incorporadas neste projeto.

A primeira opção testada foi uma estrutura tubular em metal, que se assemelhava a um diagrama de voronoi. Composta por células com propriedades diferentes entre si, variando em tamanho e forma, que era por sua vez alternadamente preenchidas com uma superfície opaca, criando uma diferenciação entre cheios e vazios. Uma solução que já havia sido modelada digitalmente em 3D no passado. Tendo bastado apenas, para uma primeira abordagem, a adaptação do modelo existente através de ajustes manuais, de modo a integrar o modelo digital existente do projeto do Maia Tech.

A segunda solução abordada, que acabou por ser a final, partiu da modelação manual em SketchUp, de um conceito baseado em esboços feitos pelo arquiteto TT. Este processo de transposição da ideia base em papel para o digital acabou por se tornar uma tarefa com grande liberdade criativa. Pois, devido a falta de precisão nos esboços, este não foi mais que uma inspiração para o colaborador encarregue da tarefa. Um processo de trabalho onde o resultado provém de quem o executa, e neste caso também dos ajustes do arquiteto TT, que influenciaram e de certo modo controlavam o resultado através de feedback. Em que a máquina não passa da ferramenta usada para “materializar” e explorar a ideia concebida pelo homem. O processo de trabalho executado apesar de originar da conceptualização em esboço, distancia-se do criticismo de Frazer e Rastogi (1998), sobre os CAD usados apenas para projetar o que já havia sido projetado. Ou seja, um trabalho valorizado por metodologias de projeto computacional em vez de computadorizado.

Características formais pretendidas

O objeto em estudo é constituído por uma composição de perfis metálicos, que podem ou não se cruzar, criando uma série de aberturas, que por sua vez, são alternadamente preenchidas por uma superfície opaca. Um desenho de fachada criado por estes dois elementos base sem qualquer tipo de posicionamento e dimensionamento pré-determinado. Possibilitando um design de fachada com inúmeras configurações dependentes apenas do cumprimento de uma série de regras formais, que condicionam e caracterizam o resultado produzido.

O objeto é assim posicionado à frente da laje do edifício, evidenciando os dois pisos através da moldura metálica que enquadra no seu interior a composição de perfis e chapas metálicas. Cada um destes perfis metálicos deve ser entendido como uma linha reta que inicia o seu percurso num ponto que

coincida com o eixo horizontal da esquadria, e termina num ponto que coincide com o eixo horizontal oposto da esquadria. Que pela conjugação de perfis irá formar a grelha, sem qualquer tipo de obrigatoriedade para um cruzamento constante dos perfis entre si. Inclusive a mesma regra aplica-se aos espaços aptos a serem preenchidos, através de superfícies opacas, que não estão vinculados apenas às aberturas criadas pelo cruzamento dos perfis entre si. No entanto as áreas preenchidas são sempre intercaladas, ou seja deve existir um espaço livre entre dois opacos que permita a entrada de luz natural.

Processo de trabalho e resultados

O objeto em análise como referido anteriormente foi a solução escolhida entre duas hipóteses de design para a fachada do edifício. A primeira solução já tinha um modelo digital 3D próprio utilizado previamente noutro projeto, sendo que, para esta análise foi apenas adaptado através de ajustes manuais. A segunda opção por ser modelada a partir de um processo manual e demorado com base em esboços, obrigou numa primeira fase, a que apenas se abordasse uma secção da fachada para fins avaliativos.

Após a avaliação das duas soluções optou-se pela segunda hipótese, passando a ser necessário um modelo tridimensional completo da fachada. O processo começou pela modelação da moldura metálica que limita a composição da fachada em cada piso. De seguida passou-se ao desenho sequencial dos perfis diagonais de forma arbitrária, de modo a gerar os espaços a serem preenchidos alternadamente por superfícies. Cada um destes elementos compositores do objeto é modelado por processos manuais, a partir da extrusão da base do tipo de perfil pretendido segundo a orientação do desenho reticulado da malha. O preenchimento dos espaços vazios gerados pela malha é igualmente produzido através de processos manuais de modelação 3D, que pelo desenho do perímetro do espaço entre perfis gera a superfície opaca. Esta metodologia de trabalho apesar de simples verificou-se como um processo repetitivo, entediante, e demorado obrigando a uma grande alocação de tempo e recursos, para a conclusão da tarefa. Levando à adoção de uma estratégia que alterou a premissa base do design da fachada. Em que colaborador encarregado da tarefa de modo a reduzir o tempo de trabalho, foi obrigado a produzir variantes a partir de módulos espalhados, ao invés de criar uma série de soluções nunca repetidas.

O processo de modelação do objeto pode ser facilmente correlacionado com o processo de desenho manual, onde o resultado produzido parte das interações sequenciais do sujeito com o suporte, provenientes do intuito e da vontade de quem o cria/desenha.

Numa fase posterior a solução proposta foi posta em causa pelo arquiteto. Sendo necessário definir um padrão único que se repetisse ao longo da fachada, de modo a agilizar o processo de fabricação e construção da solução, possibilitando o atenuamento dos custos desta abordagem. Para tal realizou-se um estudo de padrões, com uma amostra de vinte e quatro opções em desenho bidimensional ZWCAD. Para uma melhor análise dos diversos padrões criados, foi posta em hipótese a conversão de cada uma das soluções em modelo 3D e a sua adaptação ao modelo existente do edifício. Uma tarefa que se verificou inviável, tendo em conta o tempo que foi necessário abdicar para produzir a primeira solução de design da fachada. Deste modo, acabou por se ficar pela primeira abordagem gerada, decidindo-se que esta questão teria de ser adiada e repensada numa fase posterior do projeto.

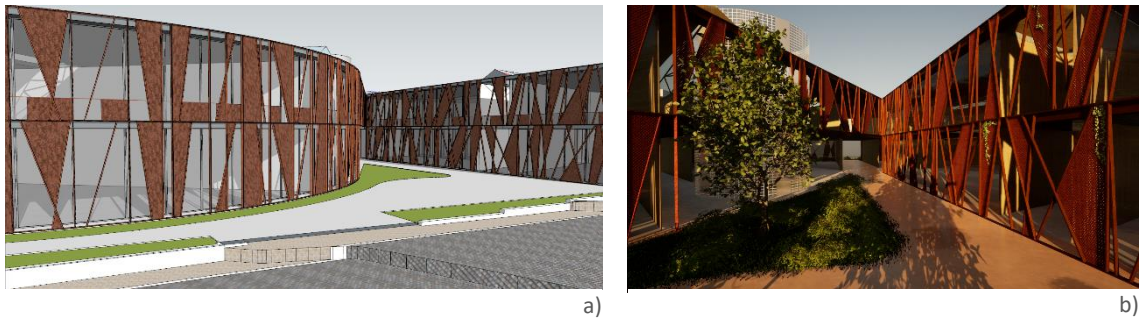


Fig. 51 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion

Dificuldades e limitações do processo

O método de trabalho que permitiu a criação do objeto em análise, passa pela repetição consecutiva do mesmo processo de modelação manual, de modo a obter uma composição constituída por uma série de elementos singulares, ainda que partilhem uma configuração base em comum (tipo de perfil). Um processo de trabalho desvalorizado pela repetição exaustiva, onde o tempo necessário para executar a tarefa está dependente da área que o objeto tem que cobrir. Podendo levar a tempos de produção bastante elevados, inviabilizando a exploração de variantes desta solução, ou o simples e muitas vezes necessário ajuste do modelo proporcionado por alterações ao projeto. Uma questão que em contexto de trabalho com metas pré-definidas, obriga a aceitação do primeiro resultado obtido por falta de tempo para produzir uma nova hipótese.

As problemáticas evidenciadas também podem ser verificadas em processos de desenho convencionais. Uma metodologia que, como referido anteriormente, se assemelha com o processo de trabalho digital em estudo. No entanto, esta parecença faz com que o resultado final possa tanto ser beneficiado como prejudicado por estes tipos de abordagem. Onde a solução está limitada pelas capacidades do sujeito que executa a tarefa, tendo em conta que este precisa de elaborar uma série de elementos compositivos diferentes dos anteriores.

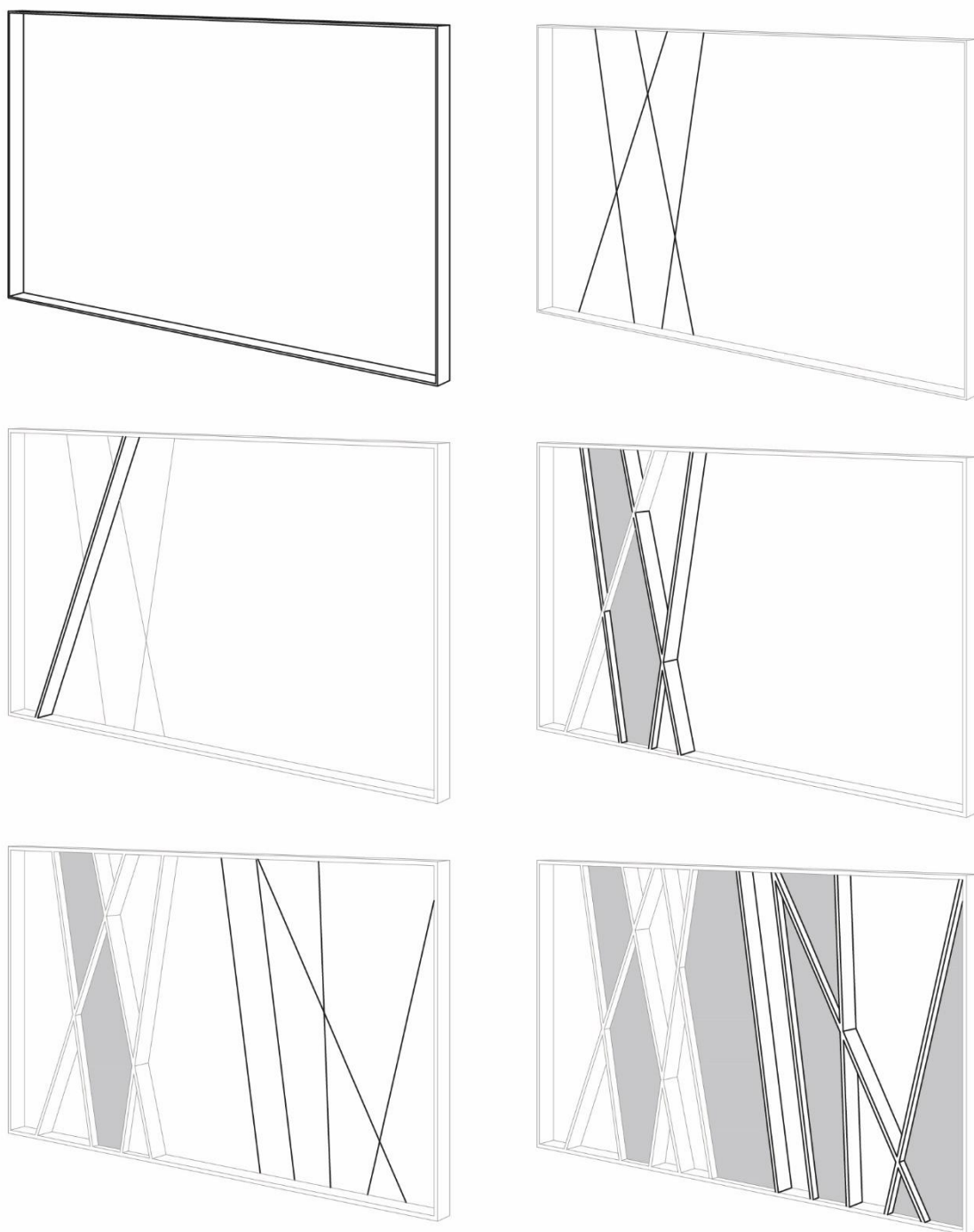


Fig. 52 | Processo de construção de caso em estudo (Imagem do autor)

03 | Caso de estudo: Hipótese de fachada 2

Introdução

A fachada em estudo foi a solução apresentada para resolver o problema inerente às fachadas cegas do edifício. Uma particularidade, que tendo em conta a premissa do projeto, a existência de uma ou mais fachadas completamente lisas e sem qualquer elaboração estética, não era tida como uma solução com interesse de aplicação para a projeto. No entanto, a solução segundo o arquiteto TT, não deveria passar pelo simples adornar ou interesse estético da fachada. Deste modo e com a imposta utilização do aço corten, idealizou-se um padrão com alto relevo, que se destaca da superfície da parede. Uma solução que posteriormente, teria de ser avaliada por especialistas/produtores da solução construtiva, de modo a averiguar o seu funcionamento. De qualquer modo, a solução final foi escolhida dentro de uma série de opções e variantes apresentadas para resolver o problema.

Características formais pretendidas

O objeto em estudo é facilmente analisado por um processo de desconstrução da forma complexa em elementos simplificados. Deste modo, verifica-se que o padrão que caracteriza a fachada é composto pela repetição sucessiva de um módulo, composto pela triangulação de duas superfícies. O emparelhado deste módulo com cópias do mesmo, produz uma faixa em zig zag, estendida pelo comprimento total da parede. A composição criada é repetida e sobreposta com a anterior, intercalando as saliências para produzir um padrão que alterna entre cheios e vazios (positivos e negativos). Esta é a lógica base da fachada, que se repete de modo a cobrir a totalidade da superfície da parede.

Processo de trabalho e resultados

O processo criativo foi realizado por via de técnicas de modelação 3D, através do software SketchUp, por permitir a execução da tarefa sobre um processo iterativo sem limites formais, por intermédio de ajustes por feedback sobre o resultado criado, embora manuais. O objeto final é criado pelo aglomerado de um módulo que obedece a regras específicas de concepção, onde as dimensões deste elemento base estão diretamente correlacionadas com o comprimento e altura da superfície onde assenta, neste caso a fachada do edifício. A modelação deste elemento base - que como referido anteriormente equipara-se às faces externas de um prisma triangular, com uma das faces assentes na superfície da parede - partiu do desenho bidimensional das linhas retas que correspondem à triangulação das duas superfícies (placas de aço corten) com a superfície de assentamento (parede). Onde a distância entre os pontos que coincidem com a superfície da parede, correspondem à divisão do comprimento da parede em x partes iguais. A mesma lógica é utilizada para definir a distância de extrusão das linhas retas, que corresponde à altura do módulo que compõe a fachada.

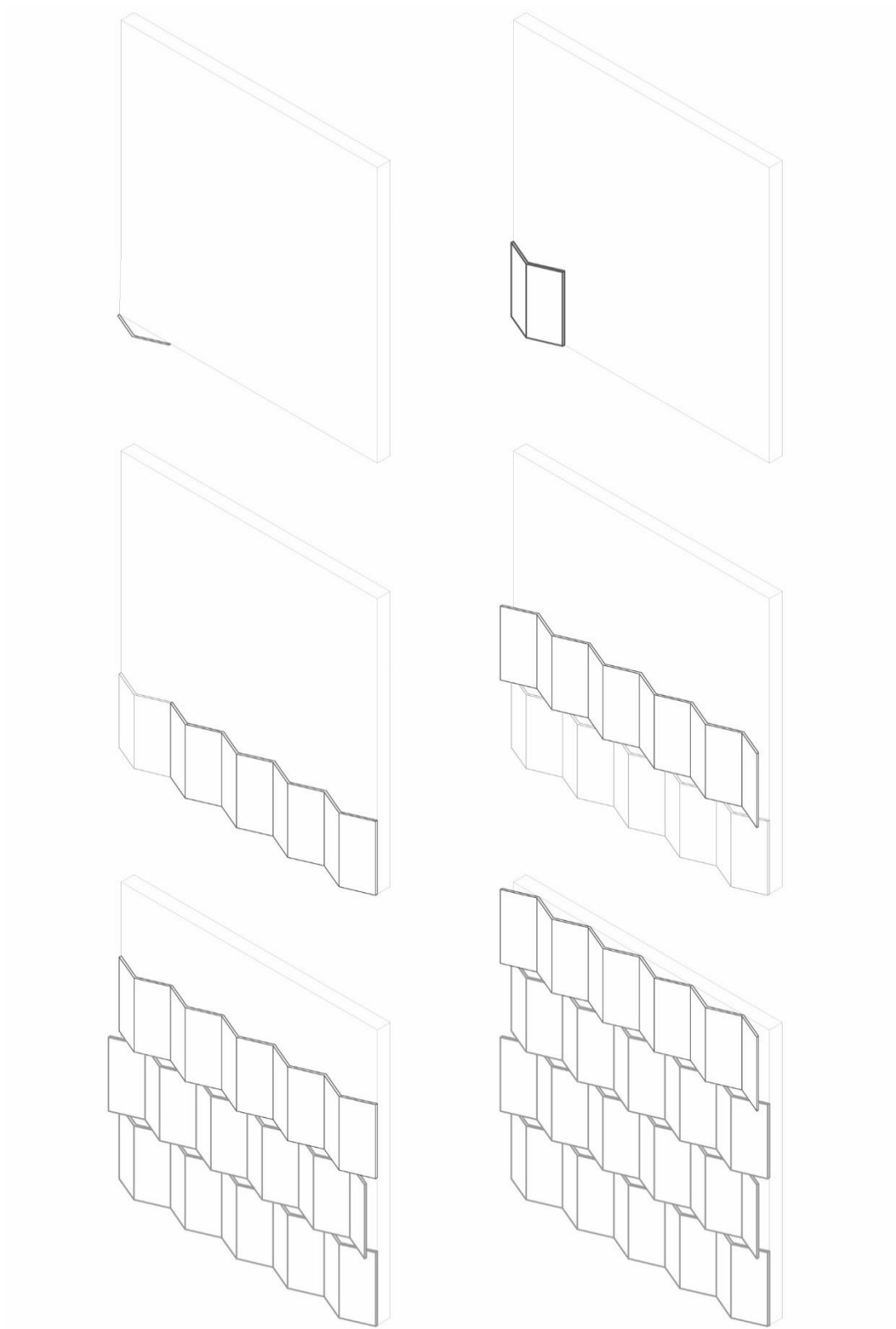


Fig. 53 | Processo de construção de caso em estudo (Imagem do autor)

Uma das particularidades desta composição é o relevo que deriva da triangulação das superfícies, onde consoante o ângulo entre as superfícies ou a distância de afastamento da reta de interseção, produz uma textura com maior ou menor relevo. O emparelhamento sequencial de cópias do módulo original pela totalidade do comprimento da parede, permite a obtenção de uma faixa em zig zag, que quando espelhada e posicionada sob a primeira, cria o intercalamento das saliências, que produz a alternância entre cheios e vazios (positivos e negativos). A solução de fachada fica então concluída, com a repetição da sequência de faixas (positiva e negativa), a partir de cópias das originais.

Numa primeira abordagem a triangulação das linhas retas foi feita a partir de um ângulo de 60º, aproximando-se das características de um triângulo equilátero. No entanto, devido às propriedades da forma geométrica o objeto obtido, acabava por se distanciar demasiado da superfície da parede, produzindo uma saliência demasiado pronunciada. A insatisfação em relação ao resultado obtido levou à exploração e testagem de variantes com diferentes amplitudes angulares. A solução ideal foi conseguida a partir de um processo de tentativa e erro, onde o autor desta dissertação participou na equipa encarregue desta tarefa, que foi obrigada a refazer “manualmente” o modelo para cada nova proposta de design. Adicionalmente, a existência de superfície de assentamento (paredes) com diferentes configurações, implicava o ajuste/adaptação do modelo produzido quando possível. Pois dependendo da semelhança entre superfícies, poderia ser necessário remodelar o objeto na sua totalidade.

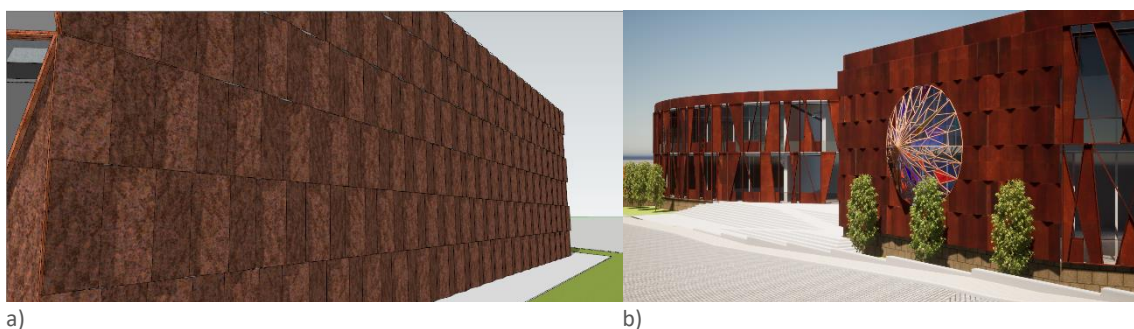


Fig. 54 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion

Dificuldades e limitações do processo

Apesar do processo de modelação digital deste objeto ser relativamente básico e fácil. O resultado obtido tende a ser fixo, por não ter qualquer tipo de capacidade de adaptação perante as alterações provenientes do processo exploratório. Um processo de execução repetitiva e entediante, onde o tempo de produção flutua consoante os momentos de exceção, levando a um consumo maior do tempo disponível. Deste modo, o processo de trabalho executado carece da versatilidade necessária para facilmente responder às questões do trabalho, onde perante alterações no projeto, obriga à execução manual de ajustes/adaptações, ou até mesmo a remodelação total da solução.

04 | Caso de estudo: Teto Acústico Escola de Dança

Introdução

Tendo em conta o carácter de utilização e a premissa de projeto, tentou-se desenvolver espaços funcionais ao mesmo tempo que valorizavam a essência do lugar. Deste modo, desenvolveu-se uma solução que pretende resolver os problemas do som como eco e ruídos indesejados em espaços amplos, através de painéis acústicos que revestem o teto das salas de aula. Uma solução convencional, caracterizada pelo posicionamento de módulos com diferentes configurações. Um aspecto tido em conta e do qual se tirou partido, para transmitir o dinamismo inerente à dança, ao desenvolver uma proposta que aborda o movimento através da sequência de lâminas que revestem o teto.

Características formais pretendidas

O objeto em estudo é constituído por um conjunto de painéis de madeira paralelos entre si, criando um efeito laminar para o teto acústico. O perfilamento destes painéis parte da vontade de criar um movimento dinâmico no teto acústico, apenas possibilitada pela sequência específica de painéis que em conjunto gera a forma pretendida. Cada painel é uma peça contínua limitada pelo perímetro da sala, posicionados sequencialmente mantendo sempre a mesma distância entre painéis. A forma é gerada por uma linha imaginária coincidente com os vértices das saliências dos vários painéis criando assim a ideia de movimento.

Processo de trabalho e resultados

O objeto em estudo conceptualizado partiu da criação de um teto acústico que criasse um ambiente dinâmico. Para tal foi desenvolvido um modelo tridimensional que permitisse ter uma ideia do espaço gerado, baseado na interseção da linha definidora do traçado do movimento dos painéis com os planos paralelos entre si. Inicialmente foram criadas linhas de controlo da forma, perpendiculares à direção pretendida para os painéis acústicos. Para gerar o movimento que resulta da composição foi necessário segmentar as linhas de controlo da forma, por pontos ao qual são atribuídas novas coordenadas a cada um deles, gerando uma linha segmentada com diferentes direções ao longo da sua extensão. Cada um desses pontos foi movimentado de forma arbitrária consoante o intuito de quem executa a tarefa. A configuração dos painéis acústicos é definida pela interseção dos planos com a linha de controlo da forma, com o objetivo de obter os pontos de controlo que configuram a silhueta do elemento. Este processo é então repetido para cada painel presente na composição até obter o resultado pretendido.

No entanto, a solução ideal foi obtida por um processo de tentativa e erro, sendo necessário criar três amostras de painéis em diferentes momentos da composição, de modo a obter uma ideia do possível resultado final. Quando a interpretação da possível solução final não correspondia às expectativas formais pretendidas, obrigava à retificação parcial ou total da linha de controlo e conseqüentemente o redesenhar da silhueta de todos os painéis. Este processo de filtragem de resultados pelo imaginar do produto final, e ajuste por tentativa e erro revelou-se como um processo repetitivo, lento e entediante.

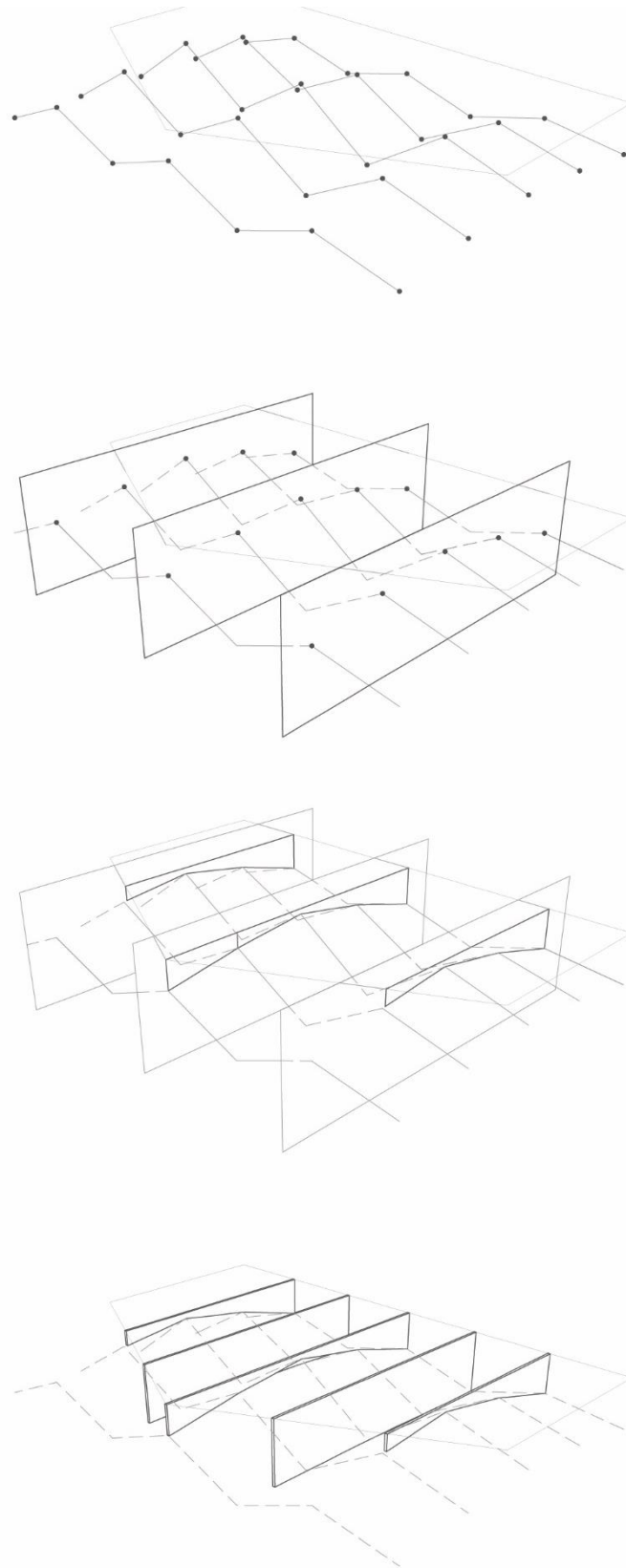


Fig. 55 | Processo de construção de caso em estudo (Imagem do autor)

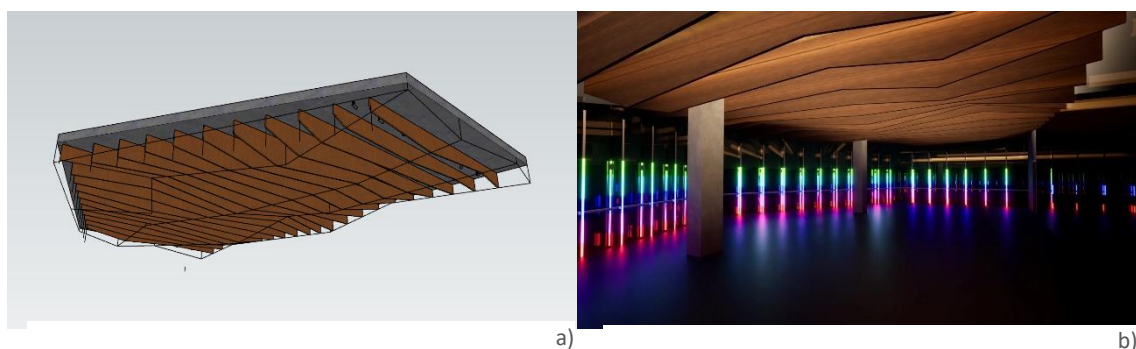


Fig. 56 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): a) Modelo original em Sketch Up; b) Imagem renderizada e trabalhada em Twinmotion)

Dificuldades e limitações do processo

O processo para obter o objeto em estudo parte da exploração via processos manuais de modelação 3D, onde o resultado é proveniente do intuito de quem o executa. No entanto, este método de trabalho peca por não permitir uma fácil visualização, ou entendimento sobre o resultado final durante a sua conceção. Tendo em conta que o processo inicia e está dependente de uma série de linhas criadas no espaço, torna-se difícil obter um vislumbre sobre o resultado final. Onde apenas a obtenção da silhueta dos painéis permite verificar a aparência da solução final. Sem esquecer que quando este não corresponde com as expectativas, obriga a repetição do processo de modelação e a execução dos devidos ajustes nas *guidelines*.

05 | Caso de estudo: Renders

Introdução

Um dos principais focos do atelier consistia na elaboração de imagens virtuais de aproximação da realidade, convencionalmente denominadas de renders. Um método que não só auxilia o arquiteto na elaboração do projeto, mas também facilita a compreensão do resultado final por parte do cliente. Tendo em conta que uma boa imagem agiliza a venda do projeto, por ser uma demonstração mais envolvente e atraente, em contraste com as representações tradicionais dos desenhos técnicos. Este tipo de imagens, pelas razões enunciadas anteriormente, era recorrentemente requisitado pelos clientes do atelier. Obrigando a um fluxo constante de produção destas imagens face às diferentes fases de conceção do projeto.

Caraterísticas formais pretendidas

Os renders criados eram produzidos segundo 3 parâmetros base - Ambiente, Textura e Luz - de modo a obter as imagens virtuais que representavam o trabalho do atelier.

Ambiente - É um parâmetro que vai para além da modelação do modelo 3D representativo do projeto proposto, havendo um maior foco sobre o ambiente simulado, ou seja, o modo como o espaço a ser vivido na realidade é imaginado e deve ser entendido. Um aspecto que limita de certo modo a criatividade do trabalho, para manter a coesão entre a imagem e o que esta representa do projeto, pois apesar de uma imagem valer mais que mil palavras, não deve nem pode transmitir as palavras erradas.

Textura - Talvez o parâmetro mais importante de renderização, por permitir a materialização do modelo digital base, que quando bem refinado produz a imagem mais ou menos realista que se aproxima com a realidade. Um parâmetro subjetivo e manipulado sobre o intuito de quem executa a tarefa, a partir do ajuste da cor, refletividade do material, dimensionamento do *UV map*, que ajusta o resultado para que corresponda às propriedades do ambiente definido. A imagem final, se analisada pela separação das texturas por secções individuais. Facilmente é verificado o carácter representativo das texturas, ou seja, individualmente carecem de realismo, visível na repetição de padrões ou divergência de refletividade entre certos materiais. Um aspeto que passa despercebido na conjugação da composição que gera o realismo da imagem final.

Luz - Um parâmetro relevante na produção de imagens virtuais, recorrentemente manipulado e estudado pelos arquitetos na conceção do projeto. Que na renderização tanto passa pela simulação da luz no espaço representado, a partir do posicionamento de pontos de luz virtuais e manipulação da luz natural, promovendo o ambiente idealizado, como o ajuste da câmara que fotografa o modelo virtual. Onde a harmonização entre as propriedades da câmara com o ambiente criado, permitem a obtenção de uma imagem nítida e detalhada.

Processo de trabalho e resultados

O facto do trabalho de atelier incorporar no seu processo de conceção do projeto metodologias com base no desenvolvimento de modelos digitais, facilitava a produção das requisitadas imagens virtuais de aproximação com a realidade. Em que estas imagens convencionalmente, são geradas a partir do

refinar dos modelos 3D em softwares idealizados para essa mesma tarefa.

No atelier Tomás Taveira os modelos digitais eram hiperligados com o *software* Twinmotion, onde era gerada uma simulação do ambiente no qual o projeto seria integrado, como por exemplo o ajuste da posição do sol de modo a criar uma aproximação da influência da luz natural com o edifício proposto. A produção de imagens é uma tarefa que beneficia do refinar do modelo digital, onde o aperfeiçoamento das propriedades que geram as texturas, dos elementos arquitetónicos, irão permitir a produção de uma imagem realista. Por vezes as propriedades aplicadas às texturizadas usadas, não correspondiam com a realidade, de modo a produzir imagens que realçam o ambiente do projeto proposto. Esta manipulação da realidade diretamente no modelo, foi uma estratégia adotada não só para melhorar a qualidade da imagem, mas também para reduzir a necessidade de trabalho de pós-produção, ou seja, ajustes e edições da imagem de modo a corrigir imperfeições. Todo o trabalho inicial era executado por processos manuais, no entanto quando já existia um ficheiro Twinmotion consolidado e as alterações no projeto não eram significativas, ou mantinham as mesmas texturas base, bastava o refresh do modelo 3D para atualizar o trabalho. Uma capacidade do software que face a pequenas alterações permitia reduzir o tempo necessário para gerar novas imagens. No entanto, o facto da primeira abordagem, ou o desenvolvimento de alterações significativas no projeto, por ser produzida através de processos manuais, fazia com que a execução desta tarefa fosse demorada. Um problema que, por vezes, devido à dimensão do modelo, era agravado pela falta de capacidade do computador para processar a geração de imagens e o correto funcionamento do software. Em que este tipo de tarefa para ser executada de modo a obter o melhor resultado, necessita de um equipamento que cumpra certos requisitos de hardware que vão para além dos mínimos.

A produção de imagens no atelier, em certo aspeto era facilitada pelas metodologias de trabalho e o constante desenvolvimento de modelos digitais durante o processo de conceção do projeto

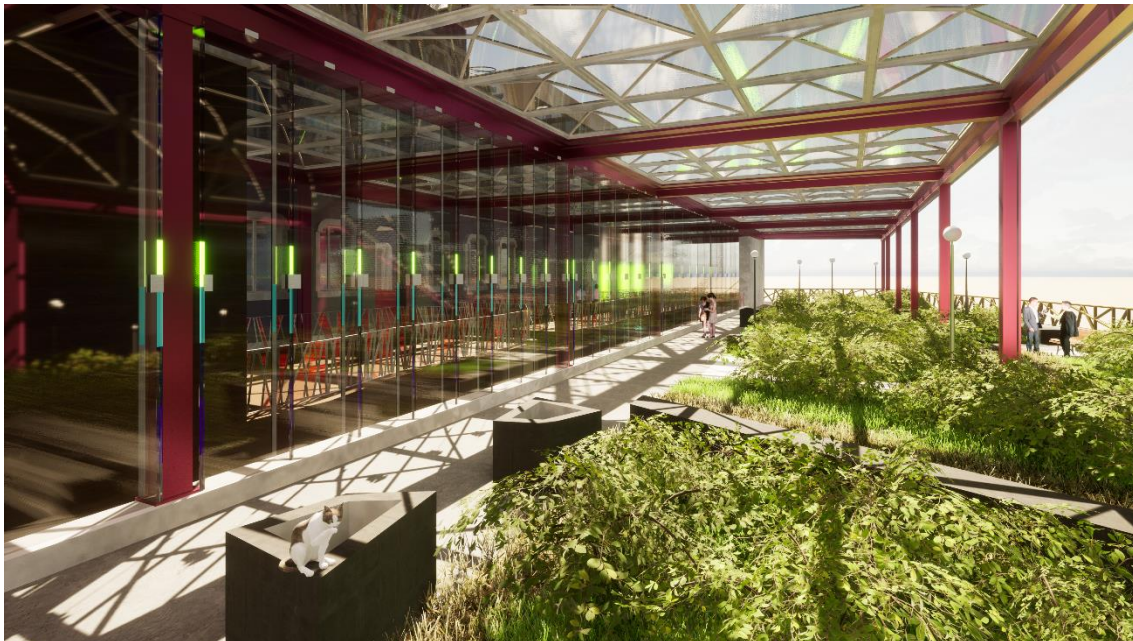


Fig. 57 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Imagem renderizada e trabalhada em Twinmotion (www.semanticscholar.org)



Fig. 58 | Resultados de caso em estudo (Imagens com direitos autorais de atelier TT): Imagem renderizada e trabalhada em Twinmotion

Dificuldades do processo

A produção destas imagens digitais era um trabalho que consumia uma parte significativa do tempo disponível para o desenvolvimento de material destinado à apresentação do projeto. A elaboração de imagens tipo render no atelier, em certo aspeto, é atenuada pelas metodologias de trabalho que integravam o constante desenvolvimento de modelos digitais durante o processo de conceção do projeto, eliminando a questão de produzir um modelo digital apenas destinado a gerar as imagens requisitadas pelo cliente. Uma situação que em fases iniciais de projeto, como o desenvolvimento de estudos prévios em que o projeto ainda não se encontra devidamente consolidado, verifica-se como um processo que implica um compromisso dispendioso, mas necessário dentro do contexto de trabalho do atelier TT.

5.4 | Síntese

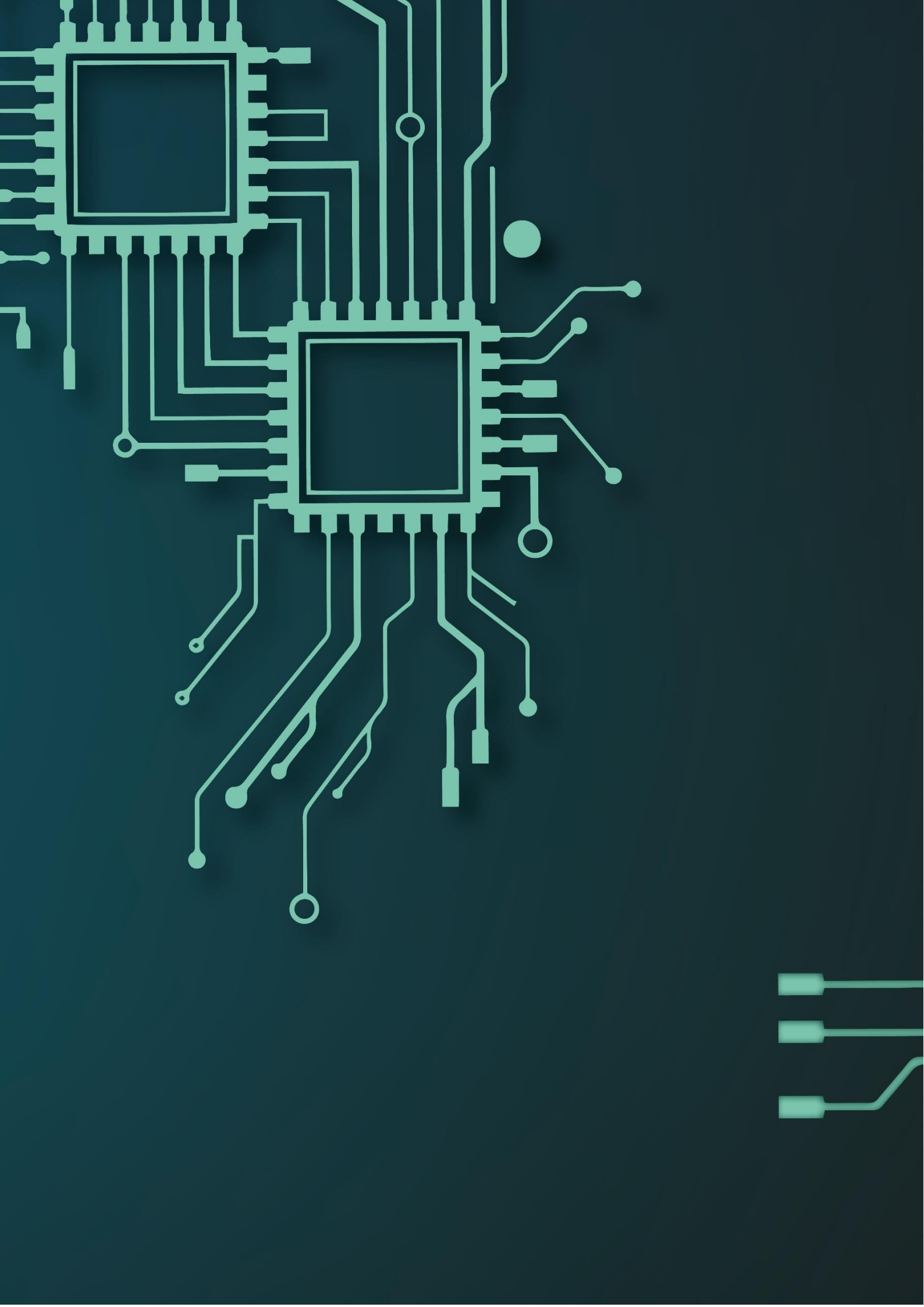
O atelier Tomás Taveira e Associados dedicava-se a uma grande variedade de projetos nas mais diversas áreas de aplicação. Deste modo, as estratégias de trabalho ou o modo como cada projeto deveria ser abordado, eram ponderadas para a especificidade e problemática de cada caso, onde o procedimento era delineado pelo arquiteto Tomás Taveira que geria a equipa do atelier. Apesar das estratégias serem determinado caso a caso, é normal que existam processos de trabalho em comum entre projetos e problemáticas diferentes. Um aspeto que origina da avaliação das experiências e precedentes de trabalho positivos, que leva à repetição dos processos adaptados para responder às problemáticas em questão. No entanto, os casos de estudo selecionado são algumas das situações de trabalho de atelier experienciadas em primeira pessoa, onde se verificaram dificuldades e problemas de execução nos processos de trabalho, que podem ser evitados pela implementação de métodos alternativos. É interessante notar que, embora cada caso responda a um problema específico, existem dificuldades e limitações comuns devido à semelhança na lógica básica do processo de resolução de problemas.

Uma das abordagens mais praticada no atelier, passava pela modelação digital de modelos tridimensionais, principalmente como uma ferramenta para fins exploratórios e de visualização da ideia. Maioritariamente executada por processos manuais de modelação 3D, baseados na interação constante do usuário que executa uma série de procedimentos para produzir um objeto. Um processo de trabalho que se verifica regularmente como exaustivo, repetitivo e entediante, culminado na obtenção de uma solução que de modo geral tende a ser estática e final. Um aspeto dos métodos de modelação 3D convencionais, verificável nos casos de estudo 1-4, que carece da adaptabilidade para que, com um simples ajuste, criar uma série de variantes da solução. Inclusive dependendo do objeto e da situação esta adaptação da solução poderia ser inviável, obrigando a criação do objeto a partir do zero. Questões que em ambiente de trabalho podem-se verificar como altamente penalizadoras, pois dependendo das circunstâncias o facto de os fluxos de trabalho serem lentos e intensivos em processos de produção manual dificulta a exploração de alternativas de resolução devido à limitação do tempo disponível para a conclusão da tarefa.

O tempo é uma força externa com que qualquer profissão necessita de lidar, que influencia a organização do trabalho e a priorização das tarefas. Uma questão refletida em todos os casos de estudo, nem que seja na mera evidência dos tempos de execução das tarefas. Nos casos de estudo 1 e 5 (Processos de produção de estudo prévio; Renders) é evidenciado um compromisso perante a produção de material para fins exploratórios e/ou visualização de ideias ainda não consolidadas, uma abordagem que no contexto de trabalho do atelier Tomás Taveira era necessária, mas convinha ser gerida tendo em conta a natureza descartável ou transitória do resultado. No entanto, este é um dilema que está fundamentado na necessidade, uma vez que a tomada de decisão exigia a avaliação e produção de hipóteses de solução em contraste com variantes de solução, segundo o intuito do sujeito que filtra e ajusta os resultados por feedback, até obter a solução ideal produzida por processos exploratórios de tentativa e erro. Onde a imagem ou aproximação com a realidade, quando têm um carácter mais representativo em vez de técnico, permite o entendimento facilitado da ideia ou conceito de projeto.

Em suma as dificuldades evidenciadas passam pela repetitividade de procedimentos executados por processos manuais, de modo a produzir objetos (modelação 3D) ou imagens (render) sem qualquer tipo de adaptabilidade ou ajuste rápido, perante alterações no projeto ou aplicação em partes do projeto com propriedades diferentes. Adicionalmente impõe-se a questão do tempo, que limita a viabilidade dos processos na produção de variantes de soluções e/ou diferentes hipóteses de solução, caracterizada na

representação do objeto proposto para fins avaliativos e de filtragem de resultados.



6. DA TEORIA PARA A PRÁTICA: APLICAÇÃO DA MODELAÇÃO PARAMÉTRICA E DA IA



6.1 | Dificuldades VS Oportunidades

A arquitetura é uma disciplina antiga que remota aos primórdios da humanidade. Ao longo do tempo, a disciplina evoluiu e adaptou-se às mudanças da sociedade, refletindo as necessidades, aspirações e avanços tecnológicos, evidenciados na história da arquitetura. Desde as cavernas pré-históricas até as catedrais góticas, cada edifício é o testemunho da capacidade humana para inovar e transformar os espaços a serem vividos. A arquitetura é uma disciplina moldada e dependente dos profissionais que a praticam, influenciados pelos conceitos, prioridades e acontecimentos dos tempos em que projetam as suas obras. Tentando harmonizar a arte com a técnica, numa profissão que exige tanto conhecimento técnico quanto a sensibilidade estética e criativa na concepção de um projeto. As exigências do mundo contemporâneo apenas aumentam as dificuldades desta profissão, já reconhecida pela sua complexidade.

O tempo é um problema com o qual todas as profissões têm que lidar, um agente externo que influencia a execução das tarefas e a capacidade de produção, face às exigentes metas impostas. Este é um problema ao qual o ser humano responde pelo desenvolvimento de métodos e tecnologia que otimizam e automatizam os processos de trabalho. Neste sentido, o trabalho desenvolvido pretende apresentar processos alternativos com base no uso do computador, que solucionem as mesmas problemáticas evidenciadas nos casos em estudo no capítulo anterior, sem as dificuldades e limitações da abordagem que foi aplicada. Deste modo, o seguinte capítulo pretende discutir e verificar os benefícios hipotéticos da evolução tecnológica aplicada ao trabalho de atelier, distanciando-se de conclusões provenientes da mera especulação teórica, pela fundamentação dos argumentos segundo as experiências práticas e implementação de processos alternativos pelo uso da modelação paramétrica e da inteligência artificial. A seguinte tabela pretende evidenciar as oportunidades de melhoria dos processos perante as dificuldades e limitações identificadas.

Dificuldades / Limitações	Oportunidades de resolução
Produção exaustiva e repetitiva.	Otimização de processos de trabalho, como adaptação e modelação do modelo mais rápido e eficiente.
O tempo de produção aumenta consoante os momentos de excepção.	Processo automatizado de geração de variantes de soluções.
Objetos com configurações diferentes obriga a soluções diferentes ou ajuste/adaptação manual das existentes.	Processo automatizado de geração de soluções perante objetos com características diferentes.
A testagem de variantes de soluções obriga a um novo modelo.	Atualização automática dos modelos com alterações pontuais de momentos de excepção.
Difícil visualização de variantes e/ou o resultado final, em momentos iniciais do modelo.	Geração controlada sob o resultado produzido.
Obtenção de resultado por tentativa e erro torna-se exaustiva, repetitiva e demorada.	Potencializar o processo criativo pela testagem do maior número possível de variações ou soluções no menor tempo possível.
Compromisso de recursos na produção de soluções não consolidadas.	Noção e visualização rápida do resultado pretendido.

Tabela 2 | Tabela comparativa das dificuldades verificadas com as oportunidades de resolução (Tabela do autor)

6.2 | Estrutura de apresentação dos processos alternativos de trabalho

A estrutura de apresentação dos processos alternativos de trabalho foca-se em três parâmetros.

- **Análise preliminar do processo manual e natureza das restrições** - Leitura crítica sobre a abordagem manual, de modo a inferir os parâmetros necessários para o processo alternativo
- **Organograma das etapas do algoritmo** - Demonstra a lógica do sistema desenvolvido, salientando as relações entre os Inputs, parâmetros, operações e outputs.
- **Construção do processo alternativo** - Descrição das diversas etapas e da lógica do sistema desenvolvido, apresentando dificuldades e processos de projeto/design computacional.
- **Possíveis Variantes de solução geradas** - Imagens de possíveis soluções geradas a partir dos sistemas construídos, juntamente com os parâmetros (variável) utilizados para gerar a solução.

6.3 | Processos Alternativos de Trabalho: Modelação Paramétrica

01 | Processo alternativo: Processos de produção de estudo prévio - Otimização de área

Análise preliminar do processo manual e natureza das restrições

Planta de Lote

- Descrição: A área disponível para a implantação do edifício; Dimensões e geometria do espaço disponível.
- Restrição(ões): Planta de lote.

Planta de Implantação de edifício

- Descrição: Planta de implantação do edifício que se pretende maximizar; Dimensões e geometria da implantação do edifício.
- Restrição(ões): Planta de Implantação de edifício.

Escala da Forma

- Descrição: O fator de escala aplicado à forma anteriormente concebida para maximizar a área de implantação.
- Restrição(ões): Valor numérico do fator de escala..

Índice Volumétrico

- Descrição: A relação entre o volume construído e a área do terreno, usado para avaliar a densidade de desenvolvimento.
- Restrição(ões): Volume do edifício; Área de implantação.

Organograma das etapas do algoritmo

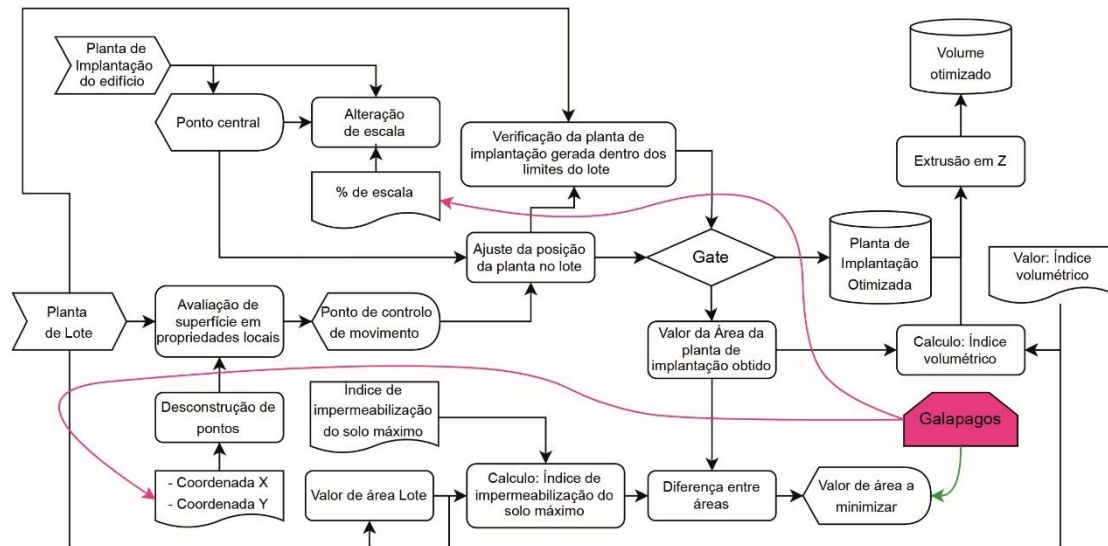


Fig. 59 | Organograma processo alternativo 04 | Processos de produção de estudo prévio - Otimização de área (Imagem do autor)

Construção do processo alternativo

O sistema paramétrico desenvolvido consiste em três partes distintas: dois processos de modificação de input, o cálculo da área pretendida, a comparação entre a área pretendida e a área obtida, e a utilização de um componente evolutivo. Os processos de modificação de input referem-se às transformações realizadas a partir dos inputs fornecidos através da alteração dos parâmetros (variável).

O primeiro processo tem início com um input (*curve*) correspondente à planta de implantação do edifício a ser otimizado. Nesta etapa, o edifício é escalado pela modificação dos valores do fator de escala parâmetro (variável), que permite explorar diferentes tamanhos da planta dentro dos limites introduzidos.

O segundo processo também parte de um input (*curve*), que neste caso representa a planta do lote onde o edifício está inserido, uma linha que limita o movimento e a mudança de escala da planta de implantação para fora dos limites do lote. O reposicionamento da planta de implantação é realizado pelo deslocamento do seu ponto central através da alteração de coordenadas, onde estes parâmetros variáveis estão limitados a um domínio de 0 a 1. Este procedimento é viável pela parametrização da superfície, que garante que o comprimento e a largura correspondam ao mesmo domínio.

Por sua vez, o cálculo da área pretendida é realizado por um sistema que determina o valor correspondente ao índice de impermeabilização do solo máximo. Neste caso, o parâmetro variável desejado é 80% da área total do lote, uma limitação imposta pelo Plano Diretor Municipal (PDM) da zona de construção deste projeto. Posteriormente, é feita uma comparação entre a área pretendida e a área obtida da planta de implantação com escala modificada. Para alcançar o valor máximo permitido pelo PDM, utiliza-se o componente *Galapagos Evolutionary Solver*, uma ferramenta de otimização que utiliza algoritmos evolutivos para encontrar a(s) solução(ões) ideal(is) em processos de design paramétrico.

Este componente ajusta automaticamente os parâmetros do modelo para minimizar a diferença entre a área da planta de implantação modificada e a área pretendida.

Após a primeira testagem do sistema com o *Galapagos*, verificou-se que a planta escalada ultrapassava frequentemente os limites impostos na busca pelo valor exato. Para evitar esse problema, desenvolveu-se um sistema que, caso a planta de implantação ultrapasse os limites impostos, não é fornecido nenhum output. Assim, durante a fase de testagem por tentativa e erro, o *Galapagos* irá anular os valores dos parâmetros (variável) que resultem nessa hipótese.

Por fim foi adicionado a este sistema a geração de um volume que, conforme o output obtido da planta maximizada, produz um objeto que corresponde ao índice volumétrico máximo permitido pelo PDM. Dessa forma, o sistema proporciona resultados otimizados e maximizados em conformidade com os requisitos impostos pelo PDM do município.

Possíveis Variantes de solução geradas

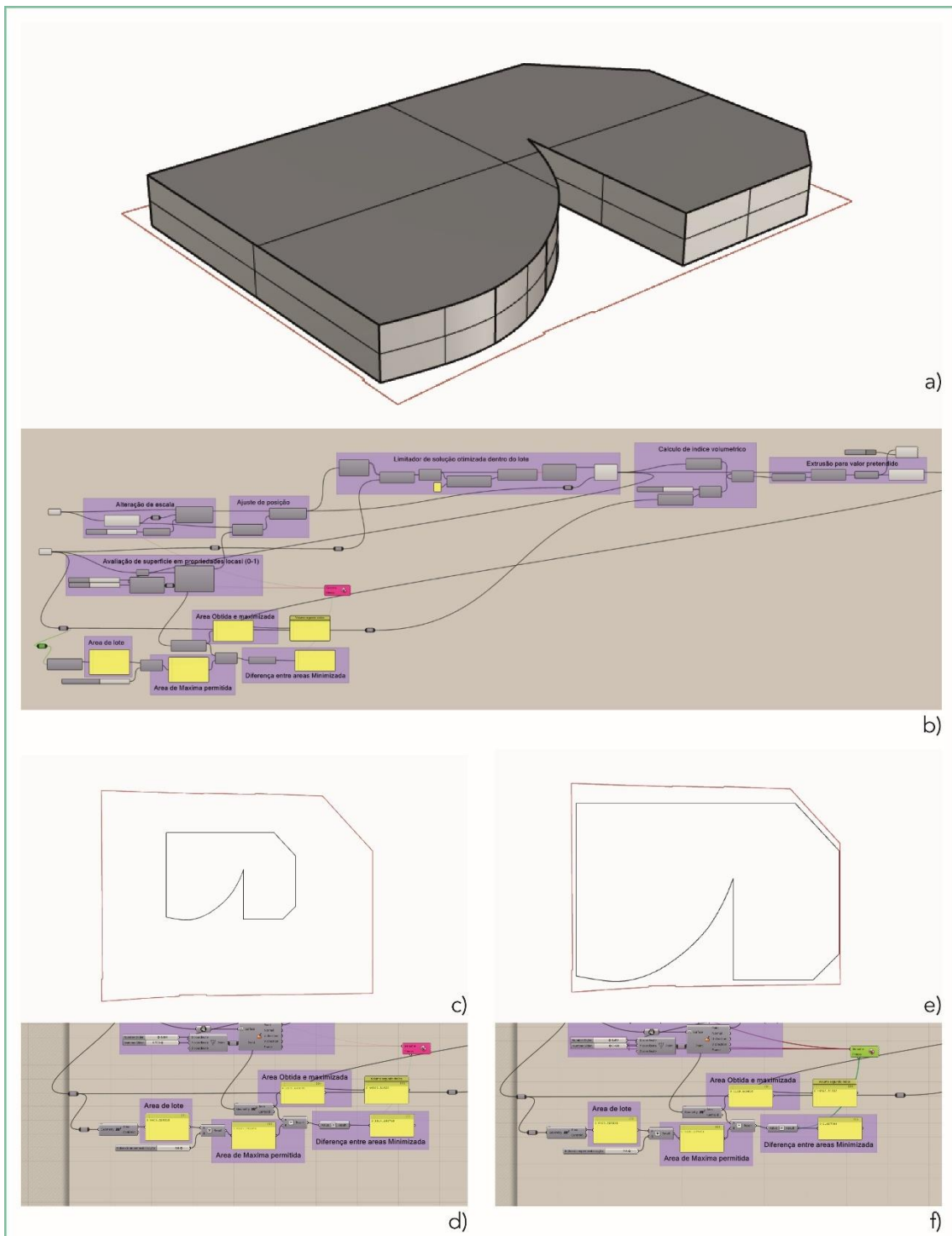


Fig. 60 | Processo de otimização de solução (Imagens do autor): a) Resultado Final com volumetria e área otimizadas + Sistema Grasshopper gerador do objeto; b) Solução inicial a otimizar + Parâmetros implementados; c) Solução com planta de implantação otimizada + Informações obtidas.

01 | Processo alternativo: Processos de produção de estudo prévio - Proposta de múltiplos edifícios

Análise preliminar do processo manual e natureza das restrições

1. Planta de Lote
 - a. Descrição: A área disponível para a implantação do edifício; Dimensões e geometria do espaço disponível.
 - b. Restrição(ões): Planta de lote.

2. Área destinada a espaços verdes
 - a. Descrição: A área disponível para alocar os espaços verdes; Dimensões e geometria do espaço disponível.
 - b. Restrição(ões): Número de fragmentos; Área do fragmento.

3. Área de implantação de Edifícios
 - a. Descrição: Divisão dos dois edifícios iniciais em volumes distintos, cada um com diferentes configurações.
 - b. Restrição(ões): Número de fragmentos; Área de fragmento; Tipo de edifício.

4. Número de cada tipo Edifícios
 - a. Descrição: Número total de cada tipo de edifício a gerar
 - b. Restrição(ões): Número de fragmentos; Área de fragmento; Tipo de edifício.

5. Configuração dos Volumes
 - a. Descrição: Posições e orientações dos volumes gerados da fragmentação da área de intervenção; Formas geométricas básicas utilizadas como ponto de partida para o desenvolvimento dos volumes; Processos utilizados para modificar os sólidos geométricos, criando diferentes formas e configurações volumétricas; Número de pisos.
 - b. Restrição(ões): Coordenadas de posição e orientação, Tipos de sólidos geométricos; Tipos de operações (subtração/adição); Número de pisos.

Organograma das etapas do algoritmo

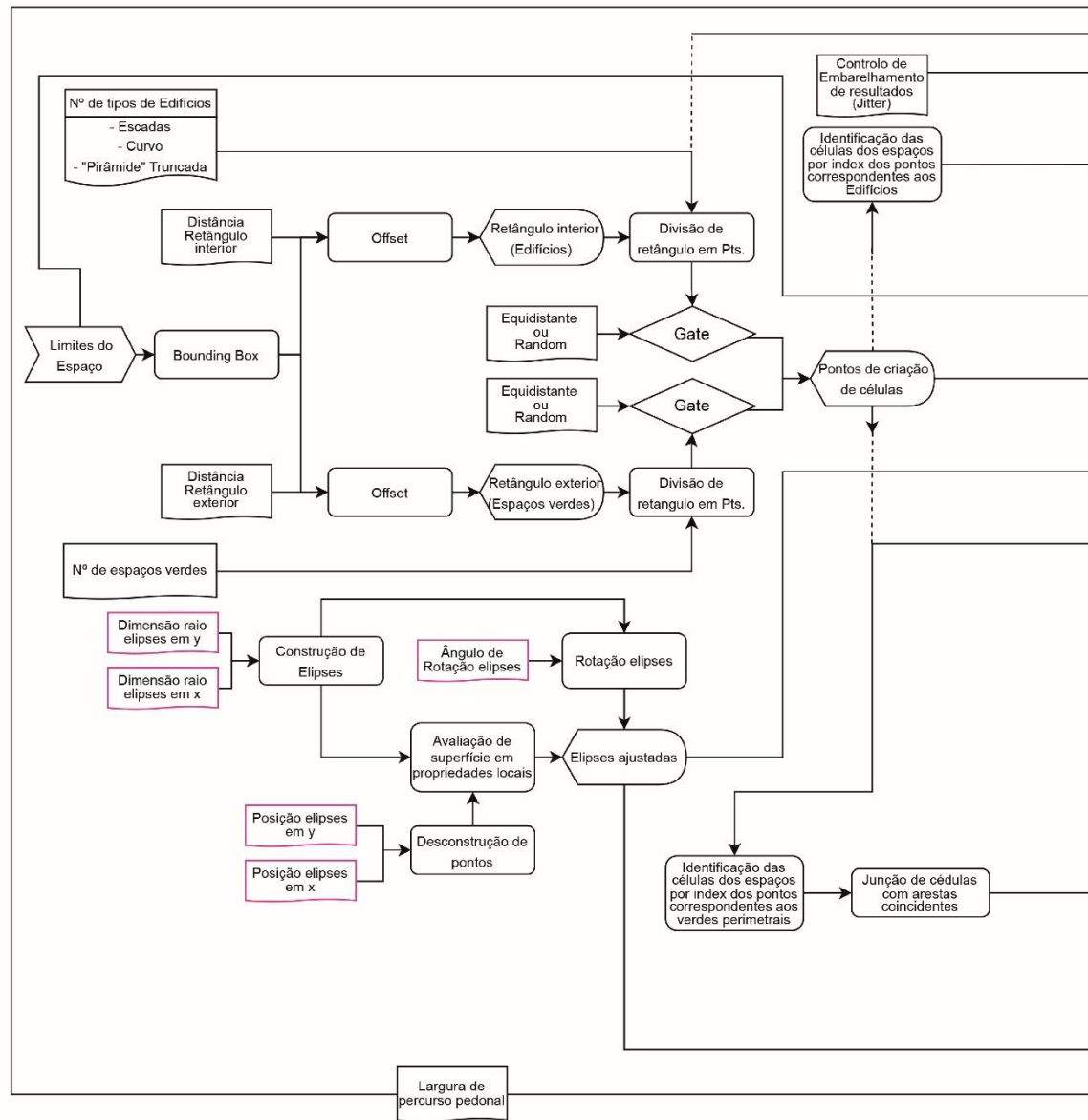
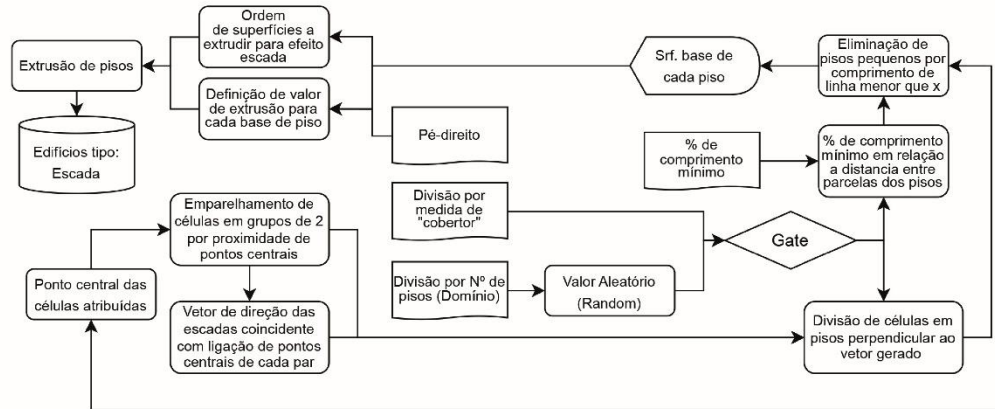
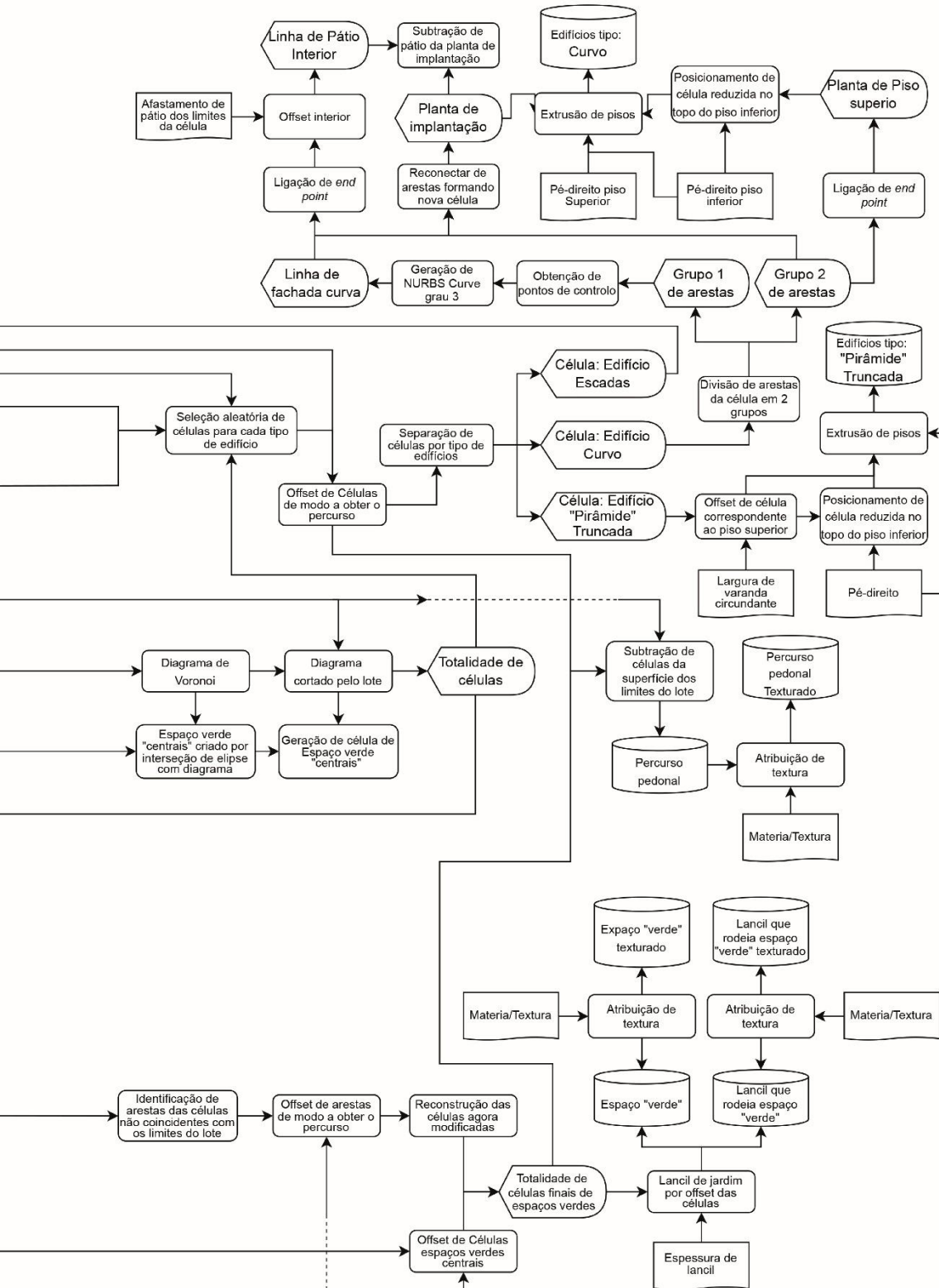


Fig. 61 | Organograma processo alternativo 04 | Processos de produção de estudo prévio – Proposta de múltiplos edifícios (Imagem do autor)



Construção do processo alternativo

O sistema criado destina-se a produzir um estudo prévio pela fragmentação do lote em três elementos principais: edifícios (com três tipologias), espaços verdes (com duas tipologias), e zonas de percurso (ou momentos de ligação). O foco está em controlar essas peças para gerar diferentes hipóteses de forma eficiente, possibilitando um trabalho exploratório sem que sejam alocados demasiados recursos, como tempo, na produção variantes ou hipóteses de ideias pouco consolidadas.

A base do sistema é uma malha urbana aplicada ao espaço pretendido, produzindo a fragmentação dos espaços conforme os parâmetros ajustados. Esta malha é gerada a partir de um diagrama de Voronoi, que divide o plano em regiões (ou células), onde cada região corresponde ao conjunto de pontos mais próximos de um ponto específico fornecido, em que cada célula do diagrama é equidistante dos pontos mais próximos. Para simplificar o processo inicial do sistema, quando deparado com uma planta que representa uma figura geométrica irregular, a mesma é inserida dentro de uma área quadrangular. A partir desse "retângulo", são criados dois "retângulos" via offset: um interior, de menor dimensão, e um exterior, de maior dimensão, de onde é gerada a população de pontos de um modo mais controlado, evitando o uso de métodos de distribuição aleatória de pontos, por sua vez, permite ajustar a quantidade, dimensão e forma das células do diagrama de Voronoi.

Os pontos que formam o diagrama são criados dividindo a linha perimetral dos retângulos pelo número de células desejadas, em que os pontos coincidentes com o retângulo exterior geram as células que criam os espaços verdes nos limites do lote, enquanto que, os pontos coincidentes retângulo interior gerem as células que produzem os edifícios. Deste modo, se o utilizador desejar cinco edifícios, serão gerados cinco pontos no retângulo interior. A distância entre pontos pode ser igual (equidistantes) ou variar essas distâncias via geração de valores de forma aleatória, uma opção que é escolhida e alternada perante a intuição do utilizador a partir do mecanismo introduzido no sistema. Apesar dos dois tipos de elementos (espaços verdes e edifícios) serem gerados com a mesma lógica, o procedimento que gera os pontos dos edifícios permite maior controle sobre a opção de pontos não equidistantes, adicionalmente o número total de células é o resultado da soma do valor total de cada tipo de edifício definido. Em suma, a forma das células do diagrama está dependente dos parâmetros ajustados tanto para os pontos, como para a dimensão dos retângulos, e ajustado para existir apenas dentro do espaço do lote em estudo. Para uma fácil compreensão do restante processo, a explicação é dividida em três etapas que correspondem aos outputs obtidos: zonas de percurso, espaços verdes e edifícios.

Zonas de percurso (momentos de ligação)

A zona de percurso, no contexto deste projeto, refere-se ao passeio criado para ligar os edifícios e espaços verdes da proposta. O processo de produção dos mesmos é facilitado pois cada célula gerada no diagrama é um elemento independente, bastando uma redução de escala das mesmas para obter o dito espaço, via uso do componente *Offset* onde o parâmetro variável fornecido é a largura de passeio pretendida. Com a redução de escala das células basta a subtração das mesmas da superfície do lote obtendo os espaços de percurso. No entanto, de modo a obter o valor escolhido, o sistema divide o valor fornecido pela metade de modo a fazer o offset igual para cada lado, produzindo o passeio com a medida pretendida. Esta lógica para melhor entendimento pode ser comparada a relação entre raio e diâmetro, em que o raio é a medida de offset de modo a obter a largura total de passeio, o diâmetro.

Espaços verdes (2 tipos)

Seguindo para o sistema que gera os 2 tipos de espaços verdes: os “centrais” e periféricos.

Espaços verdes “centrais” - produzidos a partir de duas elipses, que irão por sua vez, intercepta as células dos diagramas produzindo o dito espaço verde. Estas elipses são ajustadas manualmente pela alteração dos parâmetros (variável) respetivos as mesmas, proporcionando modificações de: dimensão, forma, rotação e posição no lote, inclusive caso as mesmas se interessem iram gerar apenas um único espaço contínuo entre a interceção da forma com as células do diagrama, gerando uma nova célula correspondente a este espaço.

Espaços periféricos - gerados como referido anteriormente pelos pontos do retângulo de maior dimensão exterior. No entanto, o offset destas células que gera o passeio de modo a evitar que as mesmas se afastem dos limites do lote, foi necessário implementar um procedimento que apenas aplica o offset nas arestas que estão no interior e não coincidentes com os limites do lote. Do movimento destas arestas são reconstruídas as células que geram os ditos espaços verdes.

Edifícios (3 tipos)

A construção das possíveis soluções de edifício começa pela atribuição de células a um dos tipos de edifícios, contudo apesar do utilizador ter controlo sobre o número de edifícios (parâmetro variável) que são gerados para cada tipo, o sistema foi construído para que a seleção das células que produzem cada edifício seja aleatória, ou seja o utilizador não tem controlo sobre a célula que irá produzir por exemplo o tipo de edifício Curvo. No entanto, caso o resultado não seja satisfatório o utilizador pode mudar o nível de “embaralhamento” (*Jitter*) e o processo do algoritmo de “embaralhamento” (*Seed*), dois parâmetros (variável) disponíveis no sistema. Os edifícios gerados, como referido anteriormente, dividem-se em 3 tipos inspirados na proposta original apresentada pelo atelier: Escada, Curvo e “Pirâmide” truncada.

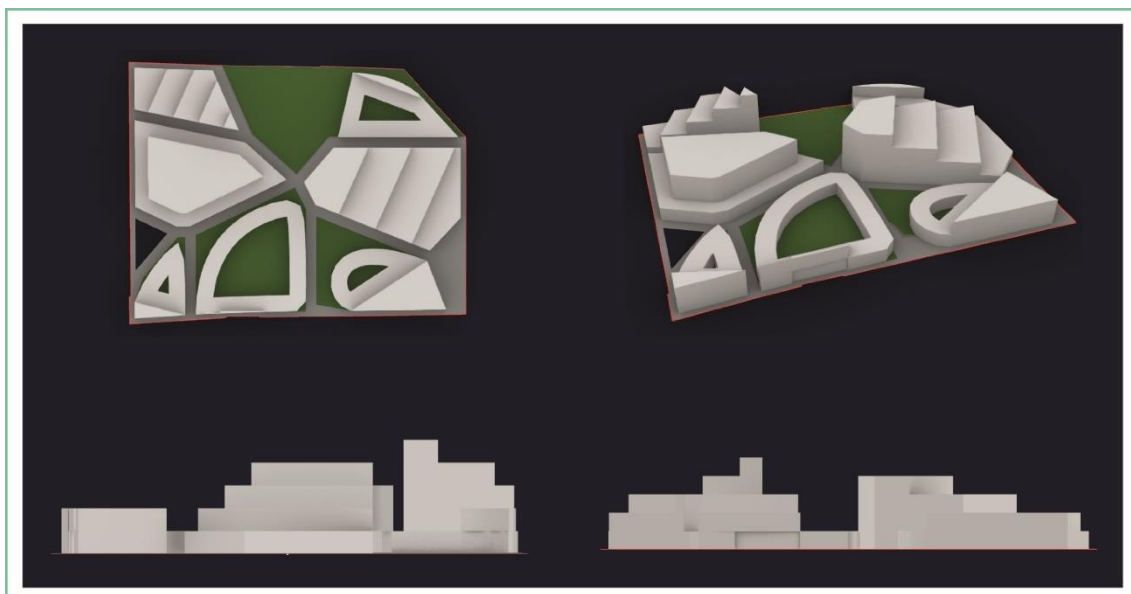
Escada - um edifício em que pela sequência dos diferentes pisos produz uma forma semelhante a uma escada. Esta solução só é possível de ser gerada em pares, logo o *number slider* deste parâmetro (variável) foi restringido para essa condição. A geração de quatro ou mais edifícios resulta na divisão destes em grupos de dois, em que o critério de emparelhamento é a proximidade entre o ponto central. Por sua vez, a direção da escada, ou seja, o lado para o qual os degraus estão virados, é definido para cada par, seguindo um vetor criado pela ligação dos pontos centrais de cada par. Por sua vez, as células geradas são divididas de modo a obter os vários pisos, tendo sido implementado no sistema duas possibilidades de geração dos mesmos pela medida do cobertor deste “degrau” (parâmetro variável) ou pelo número de pisos pretendido (parâmetro variável), uma seleção feita aleatoriamente a partir do sistema dentro do domínio fornecido. Por fim, cada piso é extraído seguindo o valor definido de pé direito para esta solução pelo utilizador (parâmetro variável), obtendo assim uma possível solução para este tipo de volume.

Curvo - Um edifício composto por faces curvas e faces planas, constituído por 1 ou mais pisos e com um pátio interior, em que o piso superior corresponde a uma porção menor da planta de implantação. Esta solução começa pela divisão das arestas da célula atribuída para a geração desta solução em dois grupos. O primeiro grupo permite obter os pontos de controlo que permitem gerar uma *Nurbs Curve* de grau 3 que corresponde às fachadas curvas. Por sua vez, este output é utilizado para gerar a planta de implantação do edifício agora com fachada curva, pelo reconectar com o segundo grupo de arestas. Posteriormente da superfície da planta de implantação do edifício obtida é subtraído o pátio

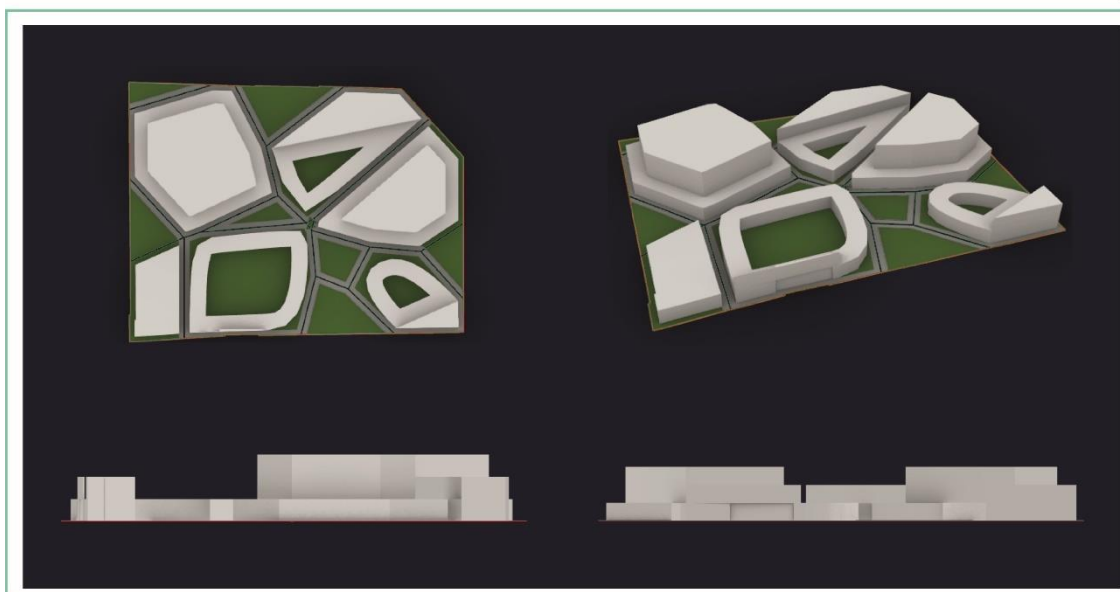
interior, que surge do offset da figura geométrica gerada a partir da ligação dos *end points* da *Nurbs Curve*. O segundo grupo de arestas, por sua vez, permite obter a secção da planta de implantação que gera os subsequentes pisos, a partir da figura geométrica gerada da ligação dos *end points* das arestas. Os dois elementos base são extrudidos com base no pé direito definido para esta solução, no entanto de modo a facilitar a construção da solução o(s) piso(s) superior só é extrudido após ser sobreposto com o anterior. A solução produzida pelo sistema também permite a construção do edifício sem um piso superior que corresponde a uma secção da planta de implantação, bastando apenas fornecer o valor 0 para o parâmetro (variável) de pisos superiores.

“Pirâmide” truncada - Um conceito apenas possível em edifícios que tenham no mínimo dois pisos, em que é sobreposta a planta de implantação com escala reduzida sobre o primeiro piso gerado, criando uma varanda circundante, um conceito apenas possível em edifícios que tenham no mínimo dois pisos. O edifício é gerado a partir do offset da célula atribuída à solução, pela distância definida pelo utilizador. Por sua vez das duas plantas (planta de implantação e planta do piso superior) é extrudido o volume com base no pé direito definido para esta solução, no entanto de modo a facilitar computação e uso posterior da solução o(s) piso(s) superior só é extrudido após ser sobreposto com o anterior.

O sistema construído como referido serve para exploração de soluções, estes foram algumas das hipóteses abordadas e testadas, no entanto produzir os edifícios a partir destas células que compõe o diagrama de Voronoi permite infinitas abordagens, cada uma com as suas características próprias apenas limitadas por quem cria o sistema, onde o resultado obtido é da autoria de quem controla o sistema que gera este objeto. Contudo, apesar de não ter sido testado neste sistema, é possível usar o *Galapagos Evolutionary Solver* que combinado com o componente *Data Recorder*, em que em vez de maximizar ou minimizar propriedades de modo a atingir uma meta, irá gerar o máximo possível de soluções (até manualmente parado ou atingir um limite pré-definido). O que permite obter infinitas opções e soluções geradas pelo próprio sistema, em que o papel do utilizador será filtrar e avaliar as propostas geradas e gravadas no sistema obtendo em resposta - caso programado para tal os parâmetros que geraram o melhor resultado segundo o intuito do utilizador.



d)



e)

Fig. 62 | Variantes de solução gerada (Imagens do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4.

02 | Processos Alternativos: Hipótese de fachada 1

Análise preliminar do processo manual e natureza das restrições

1. Limites da forma da Fachada
 - a. Descrição: Linha perimetral da fachada onde será integrada a solução.
 - b. Restrição(ões): Linha perimetral da fachada
2. Altura dos Pisos
 - a. Descrição: A altura individual de cada piso do edifício.
 - b. Restrição(ões): Altura dos Pisos
3. Moldura “Metálica”
 - a. Descrição: A moldura “metálica” que delimita a composição da fachada em cada piso; As configurações da moldura “metálica” limita a extensão de cada perfil.
 - b. Restrição(ões): Altura dos Pisos; Linha perimetral da fachada.
4. Perfil Diagonal
 - a. Descrição: Desenho sequencial de forma arbitrária dos perfis diagonais que geram os espaços a serem preenchidos alternadamente por superfícies; O desenho está limitado pela moldura “metálica”.
 - b. Restrição(ões): Coordenadas dos pontos de controle das linhas que geram os elementos; Altura dos Pisos; Linha perimetral da fachada.
5. Base do Perfil
 - a. Descrição: Extrusão da base quadrangular do perfil segundo a orientação do desenho reticulado da malha.
 - b. Restrição(ões): Coordenadas dos pontos de controle das linhas que geram os elementos; Perfil pretendido; Valor de extrusão.
6. Preenchimento dos Espaços Vazios
 - a. Descrição: Preenchimento dos espaços vazios gerados pela malha, através do desenho do perímetro do espaço entre perfis para gerar a superfície opaca.
 - b. Restrição(ões): Valor de espaços preenchidos; Coordenadas dos vértices dos espaços vazios.
7. Módulo de Repetição (parâmetro existente no processo manual, mas é também a limitação que se está a tentar ultrapassar)
 - a. Descrição: Módulos espelhados utilizados para reduzir o tempo de trabalho ao invés de criar soluções únicas não repetidas.
 - b. Restrição(ões): Módulos selecionados; Número de repetições; Posições na composição.

Organograma das etapas do algoritmo

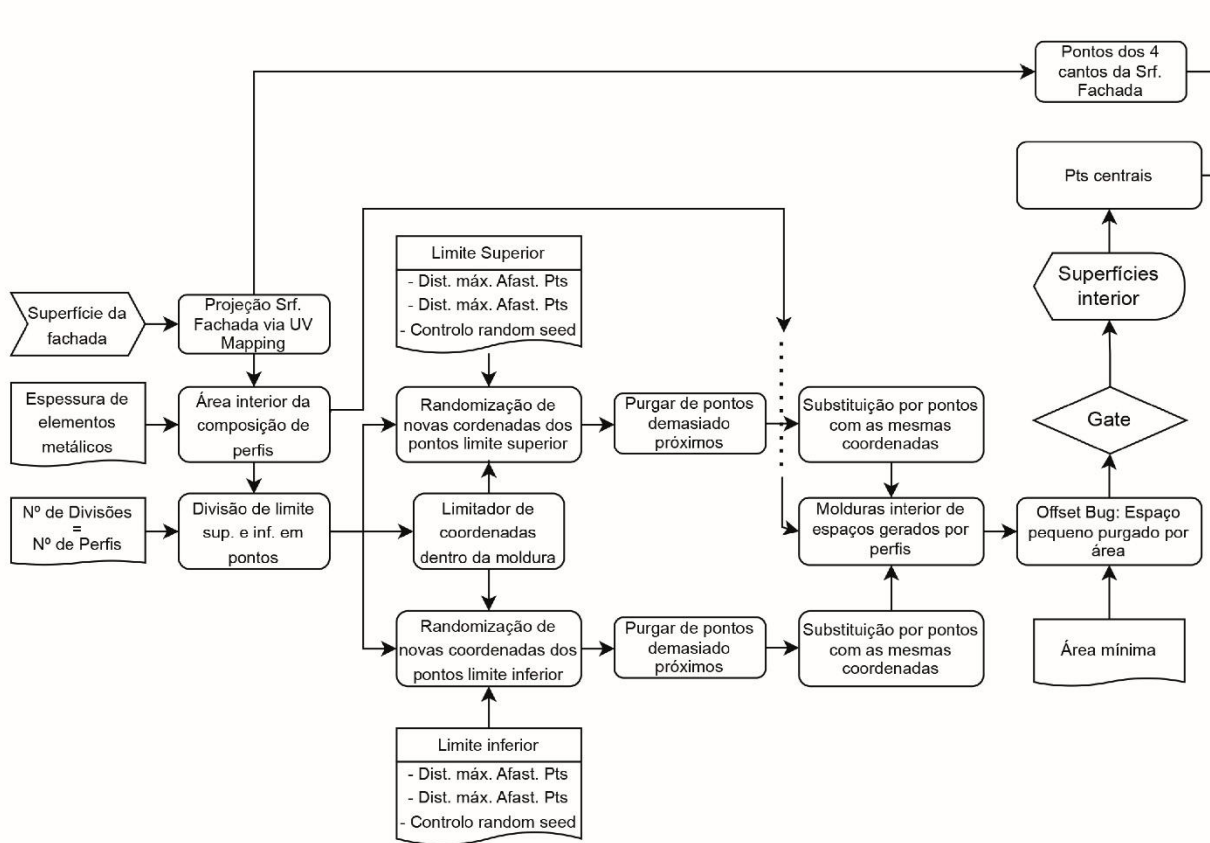
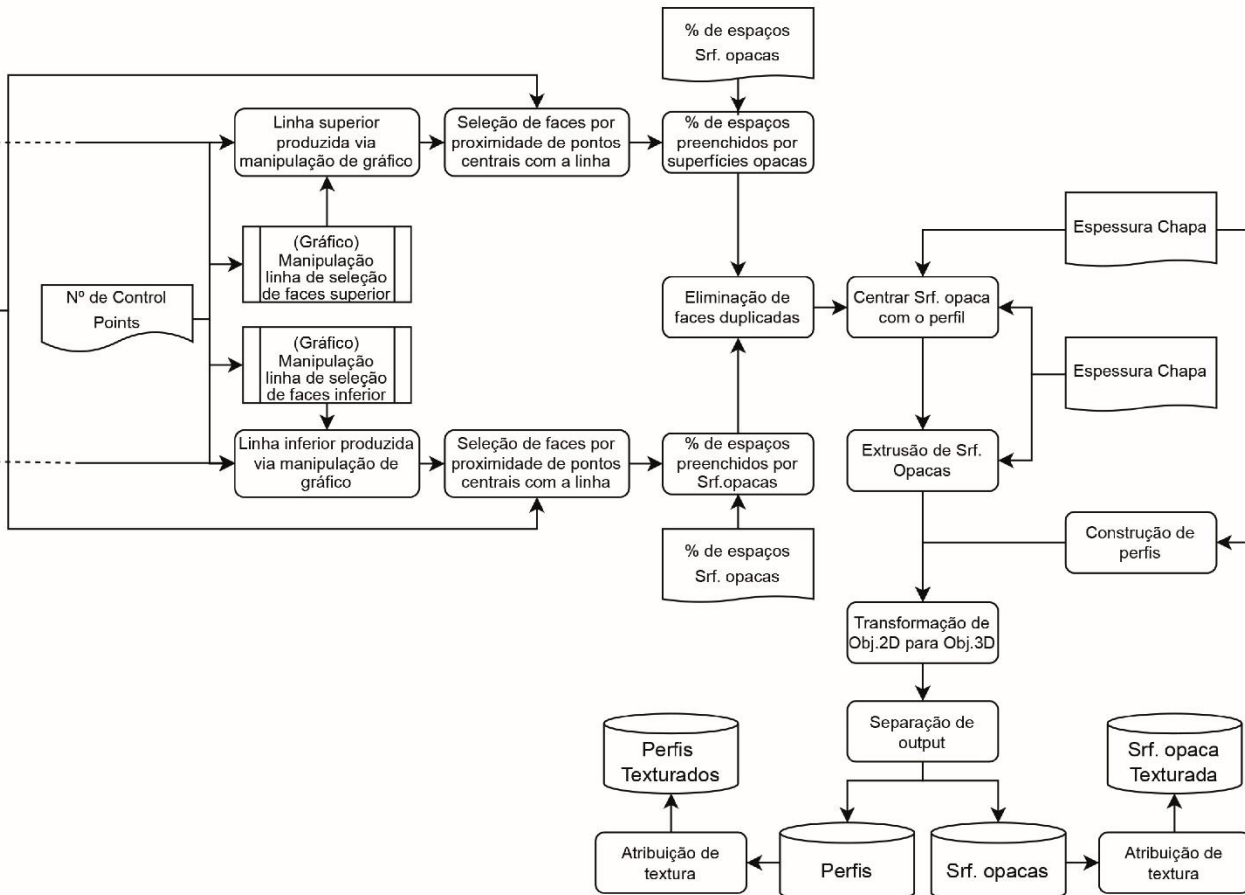


Fig. 63 | Organograma processo alternativo 02 | Hipotese de Fachada 1. (Imagem do autor)



Construção do processo alternativo

O sistema paramétrico desenvolvido para a construção da fachada em questão, começa com a definição do parâmetro *input surface* que corresponde à superfície segmentada por piso, da fachada pretendida. De modo a garantir a maior adaptabilidade possível da solução a diferentes configurações de superfície de fachada, o modelo foi construído à parte da superfície original, gerando-se uma representação bidimensional (2D) da superfície tridimensional (3D) através de mapeamento UV (*UV mapping*). Esta projeção bidimensional produz uma figura quadrangular, que corresponde à segmentação da fachada por piso, permitindo um controlo facilitado sobre qualquer manipulação de coordenadas, considerando que, nas etapas iniciais é possível prescindir de movimentos e manipulações de coordenadas em Z.

Prosegue-se então para a definição do espaço correspondente ao interior da moldura "metálica", responsável por limitar a composição gerada, mantendo-se a lógica de desenho bidimensional. Após a definição dos limites, é gerado um conjunto de pontos de controlo da composição, pela divisão do limite superior e inferior paralelos ao eixo X, em partes iguais. A quantidade de pontos é definida pelo número de divisões, um parâmetro variável ajustado pelo utilizador para obter a solução pretendida. Cada ponto é movimentado através da alteração da sua coordenada X, valor que é gerado aleatoriamente dentro de um domínio inserido pelo utilizador (parâmetro variável). Para evitar que existam perfis demasiado próximos, é estabelecido um valor de distância mínima entre os pontos, uma operação que resulta na criação de um ou mais perfis a partir de pontos coincidentes. Estes procedimentos aplicam-se tanto aos pontos presentes na linha superior quanto na linha inferior, permitindo que os valores inseridos pelo utilizador sejam distintos para o conjunto de pontos de cada linha.

A partir dos pontos, são gerados os segmentos de reta que constituem a composição dos perfis, que define a forma das superfícies opacas que preenchem o espaço entre os perfis. Durante a criação dos perfis através do afastamento dos segmentos de reta (*offset*), existe a possibilidade de serem gerados pequenos espaços nas interseção dos mesmos, estes erros vão produzir anomalias no resultado final. Para evitar esta situação, foi necessário criar um mecanismo que elimina os espaços que não cumprem a área mínima definida. O preenchimento dos espaços correspondentes às superfícies opacas é feito pela proximidade dos pontos centrais destas áreas com uma linha, onde os pontos de controlo são geridos por um gráfico. A lógica deste processo é aplicada em duas linhas, uma superior e outra inferior, sendo que cada linha é coordenada pelo seu próprio gráfico. Por sua vez, o mesmo se aplica ao número de faces que são preenchidas pelas superfícies opacas, um parâmetro (variável) em que o valor representa a percentagem de superfícies a serem preenchidas.

Os elementos, conforme referido no início, são produzidos previamente a partir de um desenho 2D. Contudo, para a "materialização" do modelo, é necessário que estes elementos sejam extrudidos no eixo Z, seguindo os valores definidos nos respectivos parâmetros (variáveis) ajustados pelo utilizador. Uma particularidade deste objeto é que as superfícies opacas são centradas com a largura dos perfis. Para garantir um posicionamento centrado com os perfis metálicos, as superfícies opacas são movidas antes de serem extrudidas para esse ponto médio, descontando-se metade da espessura da superfície opaca neste movimento de modo a garantir uma superfície centrada com o perfil.

Uma das premissas do processo de trabalho proposto foi a versatilidade e adaptabilidade para exploração de variantes de solução. No entanto, o sistema não está preparado para produzir soluções para momentos de junção de superfícies podendo gerar anomalias nesses momentos. Após alguma testagem e averiguação sobre a questão, concluiu-se que no contexto do caso em estudo, seria mais

eficiente uma adaptação via processos manuais de modelação 3D desses momentos de excepção pouco frequentes. De qualquer modo, a solução produzida é transferida para a superfície inicialmente utilizada como *input* (parâmetro), por meio de um processo de mapeamento das coordenadas UVW. A solução final é separada em dois *outputs* (perfis e superfícies), facilitando a atribuição de materiais e texturas, por exemplo, para a produção de renders arquitetónicos.

Possíveis Variantes de solução geradas

Grayscale HSB color picker with sliders for Hue, Saturation, and Brightness.

Exp. Bem. Metaicos: 0.06

Nº de Divisões: 40

Linha Superior

Dist. Max. Abast. Pts: 0.016

Random Opt: 7

Dist. Min. entre Pts: 0.20

Linha Inferior

Dist. Max. Abast. Pts: 0.050

Random Opt: 2

Dist. Min. entre Pts: 0.27

Seleção de faces preenchidas

Volume: [Graph]

Source: [Graph]

Target: [Graph]

Seleção de faces preenchidas

Volume: [Graph]

Source: [Graph]

Target: [Graph]

% de faces revestidas: 0.2

% de faces revestidas: 0.3

Largura Partic: 0.200

Espessura Chapé: 0.000

Slide if bug present: 0.0160

True if bug: 100%

Three horizontal panels showing different structural patterns of a truss structure, generated from the parameters on the left.

a)

Grayscale HSB color picker with sliders for Hue, Saturation, and Brightness.

Exp. Bem. Metaicos: 0.06

Nº de Divisões: 34

Linha Superior

Dist. Max. Abast. Pts: 0.033

Random Opt: 3

Dist. Min. entre Pts: 0.30

Linha Inferior

Dist. Max. Abast. Pts: 0.041

Random Opt: 7

Dist. Min. entre Pts: 0.07

Seleção de faces preenchidas

Volume: [Graph]

Source: [Graph]

Target: [Graph]

Seleção de faces preenchidas

Volume: [Graph]

Source: [Graph]

Target: [Graph]

% de faces revestidas: 0.3

% de faces revestidas: 0.3

Largura Partic: 0.200

Espessura Chapé: 0.000

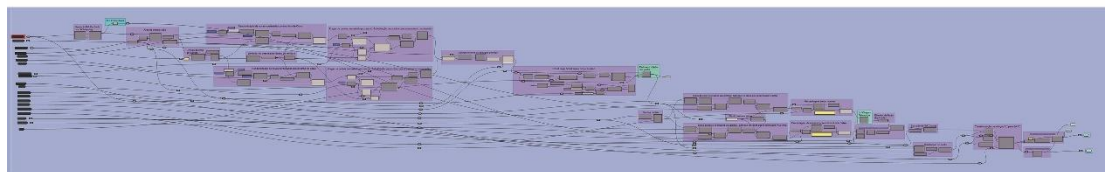
Slide if bug present: 0.0160

True if bug: 100%

A 3D perspective view of a curved truss structure, showing its complex internal geometry.

Two horizontal panels showing different structural patterns of a truss structure, generated from the parameters on the left.

b)



c)

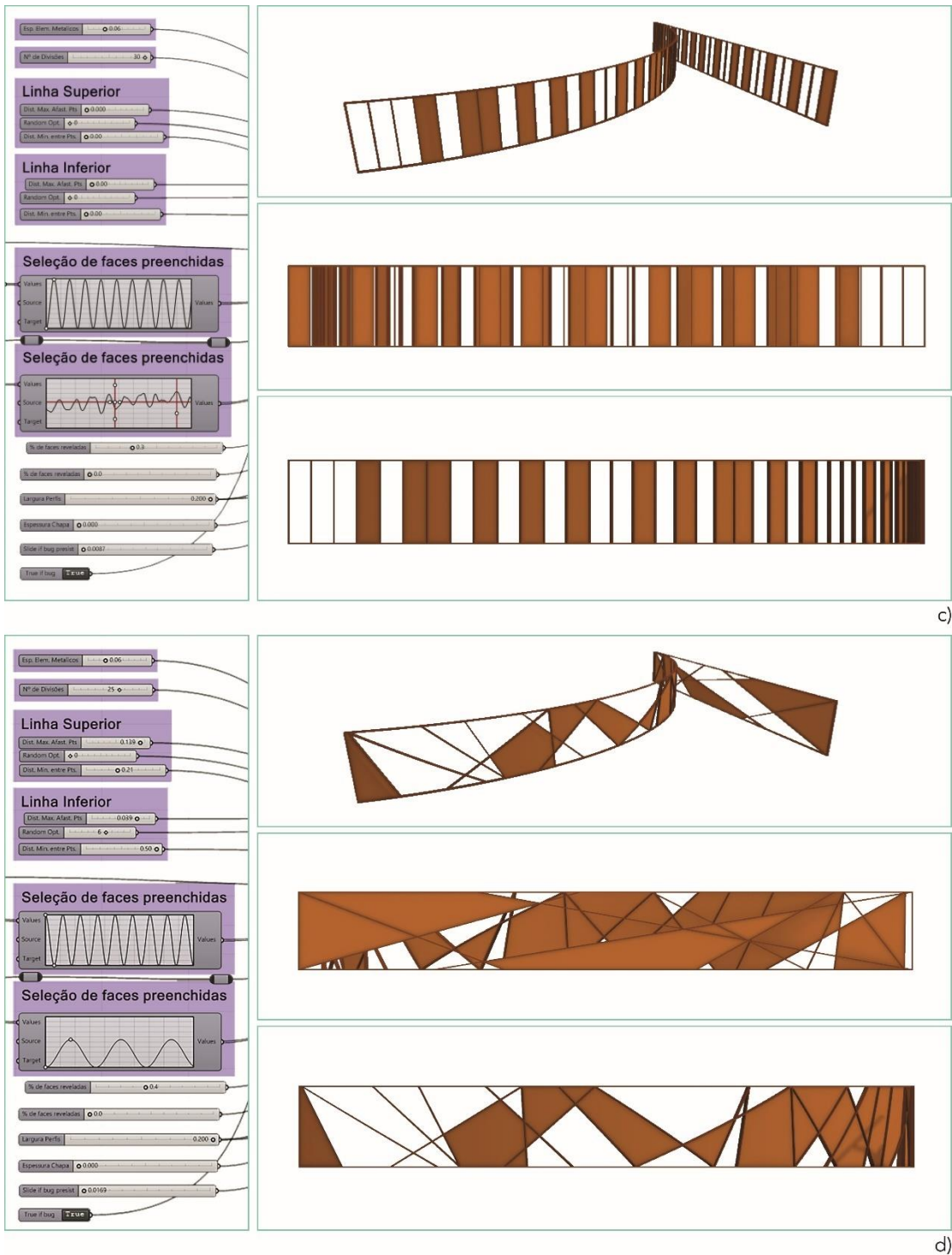


Fig. 64 | Variantes de solução gerada (Imagem do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4;

03 | Processos Alternativos: Hipótese de fachada 2

Análise preliminar do processo manual e natureza das restrições

1. Limites da forma da Fachada
 - a. Descrição: A dimensão horizontal total da parede onde os módulos serão colocados.
 - b. Restrição(ões): Linha perimetral da fachada e sua extensão.
2. Altura da Superfície da Parede
 - a. Descrição: A dimensão vertical total da parede.
 - b. Restrição(ões): Altura da parede.
3. Número de Módulos ao longo da Parede
 - a. Descrição: Divisão do comprimento da parede em X partes, determinando a distância entre pontos coincidentes com a superfície da parede.
 - b. Restrição(ões): Número de Módulos; Linha perimetral da fachada.
4. Ângulo de triangulação do Módulo
 - a. Descrição: O ângulo de triangulação das linhas retas que formam o módulo triangular;
 - b. Restrição(ões): Distância de afastamento da saliência em relação à parede; Amplitude do ângulo.
5. Altura do Módulo
 - a. Descrição: A altura da extrusão das linhas retas que definem a altura do módulo; Definida pela divisão da altura da fachada em X partes iguais.
 - b. Restrição(ões): Altura do módulo; Cércea da fachada.

Organograma das etapas do algoritmo

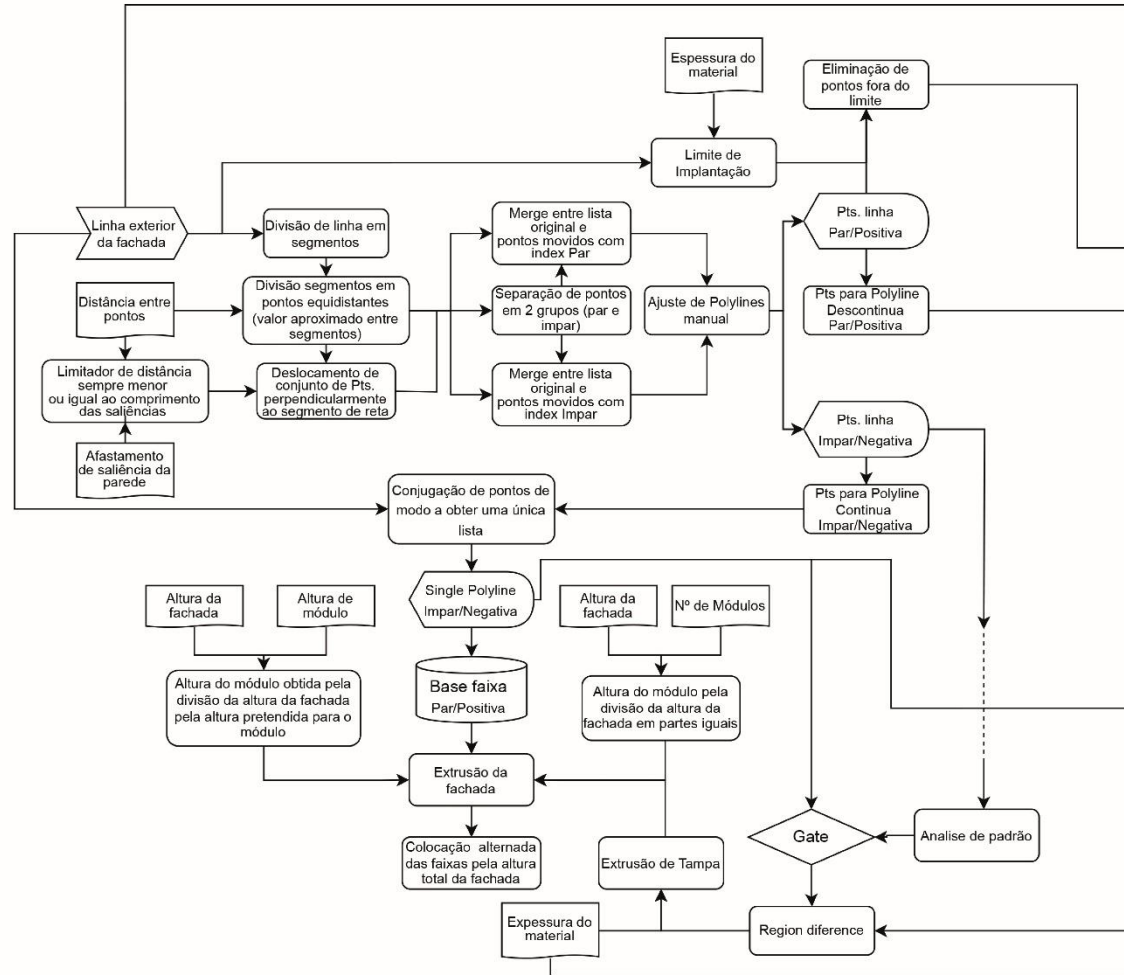


Fig. 65 | Organograma processo alternativo 03 | Hipótese de Fachada 2 (Imagem do autor)

Construção do processo alternativo

O sistema construído é iniciado por um input (Curve) correspondente à linha interior da fachada vista em planta. Esta linha é dividida em segmentos de reta, em que cada segmento gerado é subdividido em pontos equidistantes. A divisão é realizada com base em um parâmetro variável que define a distância desejada entre os pontos. Para garantir uniformidade do padrão em diferentes configurações de parede, a distância entre pontos é ajustada para ser o mais próxima possível do parâmetro pretendido, evitando assim disparidades entre segmentos com diferentes comprimentos. Posteriormente, o conjunto de pontos é movido perpendicularmente ao segmento de reta de onde foi gerado, criando as saliências dos módulos que compõem o padrão. A distância desses movimentos, que geram as saliências nos módulos, está limitada a um valor menor do que o parâmetro variável que ajusta o comprimento dos módulos, evitando anomalias no objeto final.

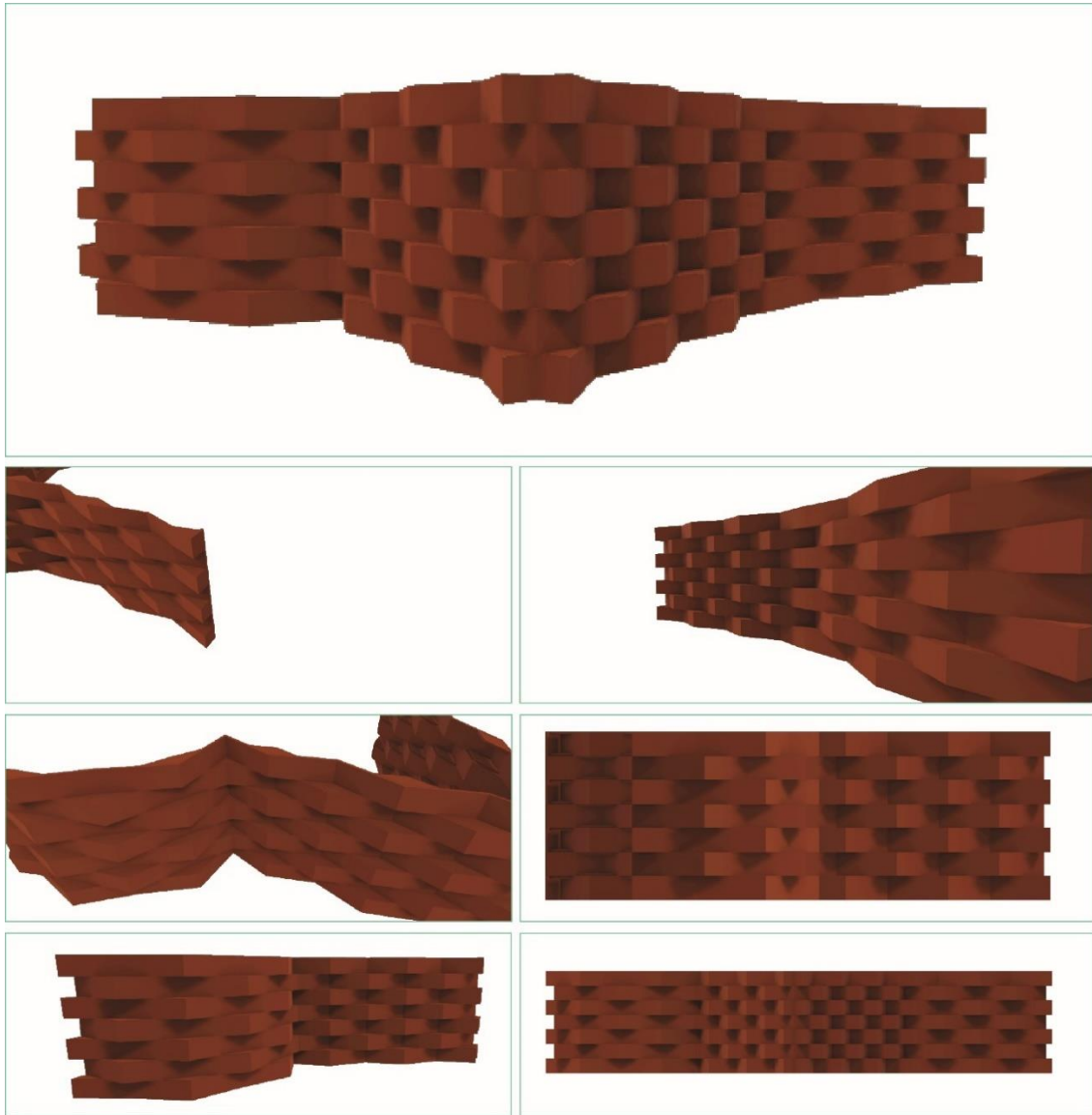
Deste modo, foram criados dois procedimentos para resolver diferentes momentos de exceção. O primeiro foca-se em manter a forma do elemento dentro dos limites pretendidos, substituindo os pontos que ultrapassam o limite por pontos de interseção das linhas entre si, uma anomalia geralmente causada por uniões que produzem ângulos agudos. O segundo procedimento procura a união das diversas *polylines*, substituindo os *end points* pelos pontos de interseção entre as *polylines* do conjunto par ou ímpar.

De seguida, para finalizar a linha interior é necessário fazer os devidos ajustes nos momentos em que as *polylines* terminam com pontos finais afastados da linha correspondente à superfície da parede, é necessário criar uma união com o *end point* da *polyline* e um ponto coincidente com a superfície da parede. Contudo, é preciso que essa união seja feita mais para dentro levando ao encurtamento da *polyline* para que, no passo seguinte, o *offset* que cria a *polyline* exterior tenha o seu ponto final coincidente com o fim da superfície, neste caso a parede. Partindo para a finalização do desenho bidimensional da base das faixas (positiva e negativa), pela união da linha interior com a linha exterior.

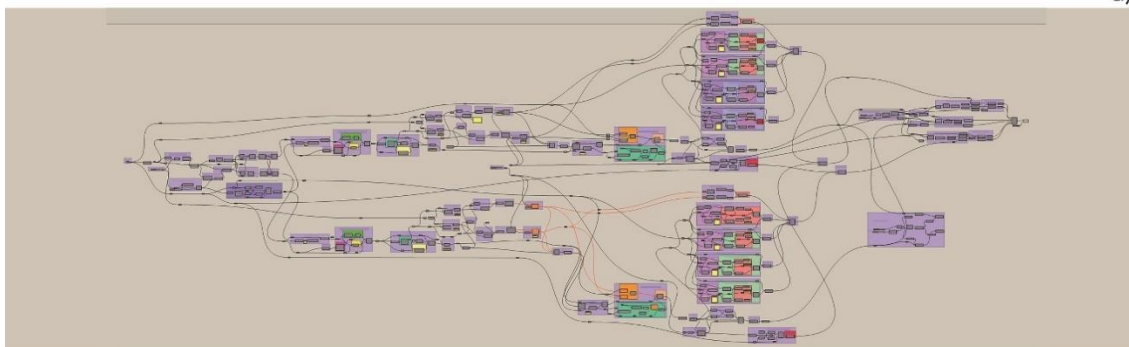
A “materialização” do objeto é feita através de um procedimento de extrusão, para o qual foram testados dois processos que permitem outputs diferentes, ajustados por diferentes parâmetros (variável). Os valores obtidos em cada um desses processos são usados para empilhar em sequência alternada as duas faixas geradas, criando o padrão de fachada que alterna entre cheios e vazios. O primeiro procedimento define a altura do módulo pela divisão da altura da fachada em partes iguais, seguindo o valor introduzido nos parâmetros (variável). O segundo processo define o valor da extrusão (altura do módulo), ao dividir a altura da fachada de modo a obter um valor aproximado ao parâmetro (variável) definido, neste caso a altura do módulo. Um processo que permite obter uma composição uniforme em fachadas com diferentes cêrceas. O modelo é finalizado pelo preenchimento dos espaços gerados pela alternância entre faixas (positiva e negativa). Um processo que envolve a identificação da área entre as *polylines* positivas e negativas (par e ímpar), pelo uso do componente *Region Difference*. A seleção dessas *polylines* é feita pela avaliação do padrão de coincidência ou afastamento dos *end points* das *polylines*. Em seguida, a superfície gerada é extrudida com a mesma espessura definida para os módulos e unida às faixas antes destas serem empilhadas alternadamente, resultando no padrão de fachada que alterna entre cheios e vazios.

O sistema desenvolvido, que gera o objeto em questão, transcende a problemática inicial ao abordar a complexidade das fachadas de maneira mais abrangente. A solução produzida por processos manuais foi idealizada apenas para paredes retas, sem considerar a continuidade do padrão em momentos de união em ângulo entre paredes. Em contraste, o sistema paramétrico desenvolvido no Grasshopper Rhino oferece uma solução mais adaptável e sofisticada, capaz de lidar com essas complexidades adicionais.

Possíveis Variantes de solução geradas



a)



b)

Fig. 66 | a) Variantes de solução gerada (Imagem do autor); b) Sistema grasshopper gerador do Objeto

04 | Processo alternativo: Teto Acústico Escola de Dança

Análise preliminar do processo manual e natureza das restrições

1. Direção das Linhas Guia
 - a. Descrição: A direção das linhas guia é perpendicular à direção pretendida para os painéis. No entanto, os painéis é que dependem da direção das linhas guia; Direção igual para todos as linhas.
 - b. Restrição(ões): Direção das linhas guia escolhida pelo utilizador.
2. Número de Linhas Guia
 - a. Descrição: Número total de linhas guia pretendidas.
 - b. Restrição(ões): Quantidade de Linhas Guia.
3. Manipulação dos Pontos de Controlo
 - a. Descrição: As linhas guia são formadas por uma sequência conectada de segmentos de reta (polylines abertas) moldadas pelo ajuste de coordenadas dos pontos de controlo. As linhas produzidas vão condicionar a forma dos painéis gerados.
 - b. Restrição(ões): Coordenadas dos pontos de controlo.
4. Comprimento dos Painéis
 - a. Descrição: O comprimento de cada painel está condicionado pelo afastamento (offset) dos limites da forma do espaço em que o objeto se insere.
 - b. Restrição(ões): Afastamento dos limites do espaço (offset); Limites da Forma do Espaço.
5. Distância entre Planos (e entre Painéis)
 - a. Descrição: A distância entre planos é igual à distância entre painéis, um valor arbitrário, mas sempre igual para toda a composição, que condiciona o número de painéis/planos utilizados segundo o espaço de aplicação.
 - b. Restrição(ões): Distância entre painéis; Limites da Forma do Espaço.
6. Limites da Forma do Espaço
 - a. Descrição: Linha perimetral da superfície do teto do espaço que integrará o objeto.
 - b. Restrição(ões): Limites da Forma do Espaço

7. Espessura dos Painéis

- a. Descrição: Os painéis têm todos a mesma medida de espessura, decidida pelo sujeito que fez o objeto. Valor arbitrário sem dependência ou restrição, aplicado na extrusão da superfície correspondente à face do painel.
- b. Restrição(ões): Espessura dos painéis

Organograma das etapas do algoritmo

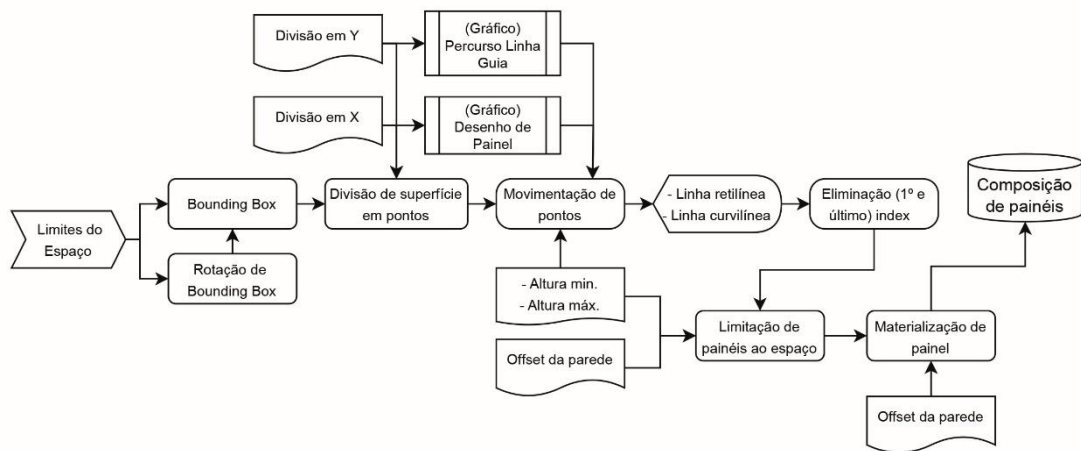


Fig. 67 | Organograma processo alternativo 04 | Teto acústico Escola de Dança (Imagem do autor)

Construção do processo alternativo

Um sistema paramétrico começa sempre por um input (parâmetro), neste caso um parâmetro Curve definidor do limite do espaço que irá integrar o objeto pretendido, em que a linha perimetral deve corresponder ao teto do espaço usado, que nas etapas finais do algoritmo irá funcionar também, como os limites em z.

Um dos parâmetros realçados na análise preliminar deste caso de estudo é a direção final dos painéis, para tal foi desenvolvida uma solução que permite a rotação dos painéis para a direção pretendida através de um *number slider* com range 180. O objeto é igualmente produzido através da modificação das coordenadas dos pontos que criam os painéis, como na modelação por processos manuais. Neste processo alternativo estes pontos são gerados pela divisão da superfície, correspondente ao espaço que integra o objeto, numa grelha de pontos definida por dois parâmetros (variável): Divisão em X produz os pontos que definem o número de vértices no painel; Divisão em Y determina o número de pontos de controlo da linha guia implícita no sistema. Pois neste processo alternativo, o impacto da linha guia traduz-se na interação dos valores proporcionados pela manipulação dos gráficos, em vez da sua configuração física no modelo digital como é verificado no processo manual.

Os gráficos por sua vez, são os parâmetros (variável) ajustados manualmente pelo usuário que permitem um maior controlo e influência sobre o objeto gerado pelo sistema. Uma solução aplicada em alternativa ao gerar de um objeto final por processos “aleatórios”, como pelo uso de um sistema que se apoia no componente *Random* para produzir uma série de novas coordenadas para os pontos. Contudo, essa

manipulação está confinada pelos valores definidos nos parâmetros (variável), que estabelecem uma altura máxima e uma altura mínima para os pontos. Desta forma, as novas coordenadas dos pontos de controlo, derivadas do ajuste manual nos gráficos, são sempre limitadas pelo domínio introduzido através dos parâmetros (variável). Esta é uma das restrições que possibilita a produção de uma solução viável ao contexto de aplicação.

As linhas que delineiam cada painel são assim subsequentemente produzidas a partir dos pontos manipulados anteriormente. Nesta abordagem com o intuito de, do mesmo sistema, produzir a maior variedade possível de soluções, optou-se por permitir que o sistema gere automaticamente uma linha curvilínea ou retilínea, através da seleção manual de uma das duas opções definidas. Deste modo, para além da configuração da linha que desenha o painel passar a ser também um parâmetro (variável) do sistema, também permite uma maior flexibilidade no design final, adaptável às necessidades específicas do projeto. Por fim, estas linhas são previamente ajustadas para que o comprimento dos painéis criados, seja limitado pelo afastamento dos limites do espaço, antes da extrusão da linha e implementação da pretendida espessura para o painel. Dois parâmetros (variável) que o usuário pode ajustar conforme entende, garantindo que o objeto se adequa ao projeto.

O sistema criado quando usado na exploração de soluções, que dependendo do novo valor e dos parâmetros alterados, obrigava a constante computação de outputs. Esta é uma particularidade destes softwares de modelação paramétrica, em que a não ser que estas operações sejam travadas, as ligações irão estar constantemente a transmitir a nova informação. Uma situação no mínimo desagradável, pela lentidão de resposta num processo automatizado devido às constantes alterações a serem computadas. Deste modo, foi criada uma interface de utilização que permite alternar entre o output final (objeto) e a exploração a partir de um output simplificado, que fornece os requisitos mínimos para o entendimento da solução, possibilitando a alteração dos parâmetros (variável) sem qualquer tipo de “travagem” no software.

O objeto produzido é gerido pela interação contínua do usuário com o sistema, pelo ajuste dos parâmetros (variáveis), possibilitando a exploração rápida e eficiente de soluções. Este sistema paramétrico, programado em Grasshopper, produz uma série de possíveis soluções para o problema arquitetónico analisado, via processos de design/projeto computacional.

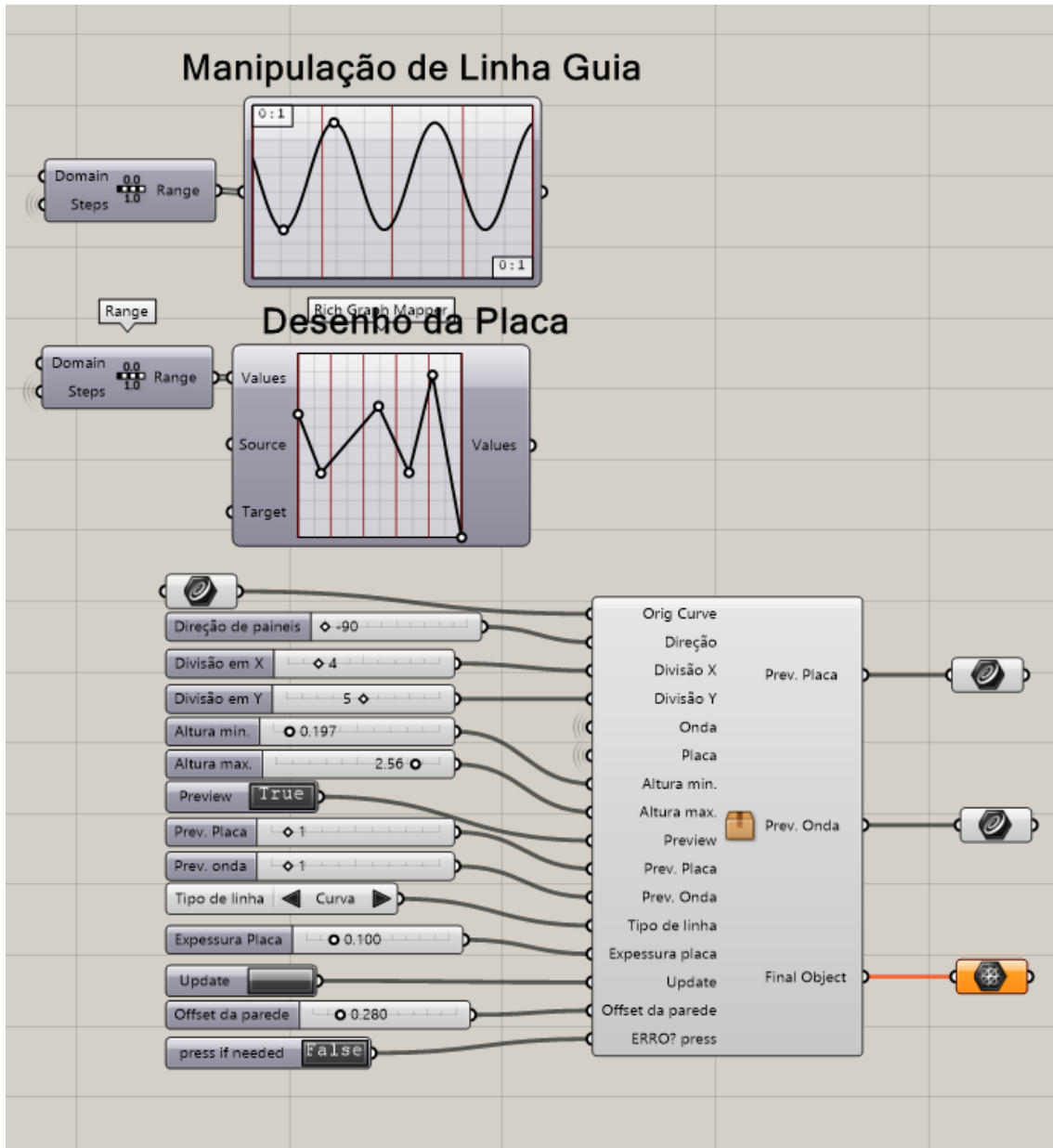
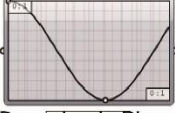


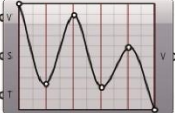
Fig. 68 | Sistema grasshopper com cluster e “interface” de utilização. (Imagem do autor)

Possíveis Variantes de solução geradas

Manipulação de Linha Guia

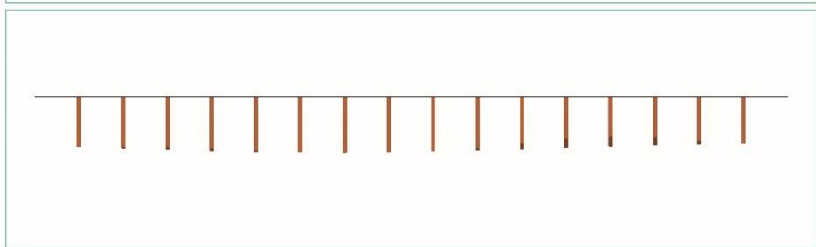

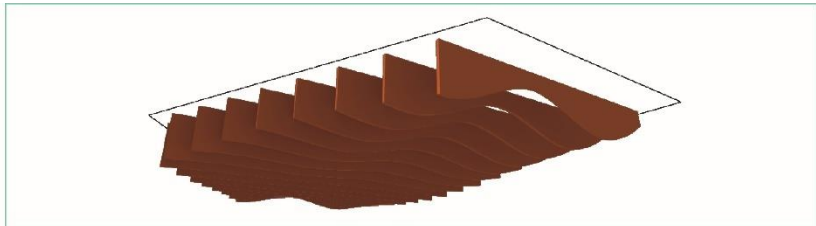


Desenho da Placa



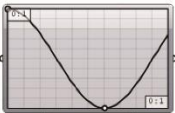
Controles de parâmetros:

- Direção de painéis: -90
- Divisão em X: 17
- Divisão em Y: 5
- Altura min: 0.288
- Altura max: 1.54
- Preview: Pa1.5e
- Previ. Placa: 1
- Previ. onda: 1
- Tipo de linha: Curva
- Espessura Placa: 0.100
- Update
- Offset da parede: 0.280
- press if needed: Pa1.5e

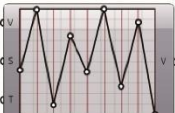


a)

Manipulação de Linha Guia

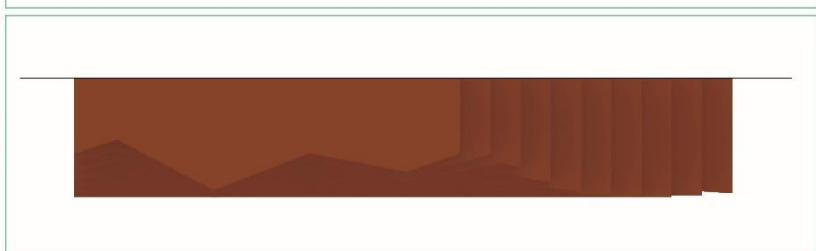
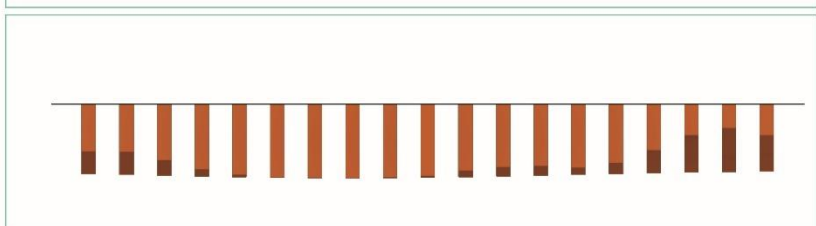
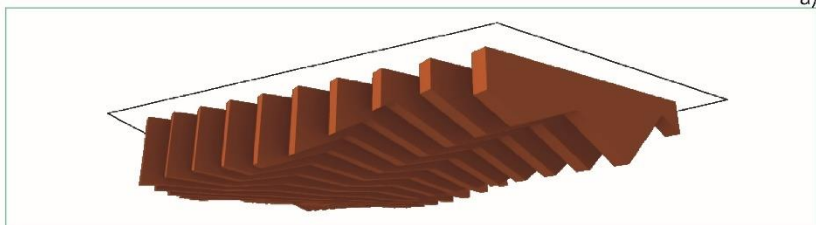


Desenho da Placa

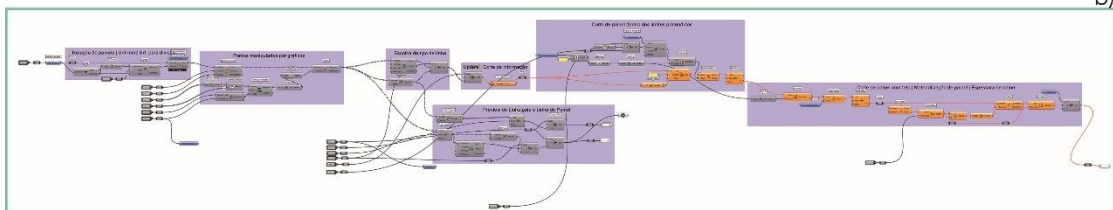


Controles de parâmetros:

- Direção de painéis: -90
- Divisão em X: 20
- Divisão em Y: 8
- Altura min: 0.288
- Altura max: 2.03
- Preview: Pa1.9e
- Previ. Placa: 1
- Previ. onda: 1
- Tipo de linha: Reta
- Espessura Placa: 0.385
- Update
- Offset da parede: 0.917
- press if needed: Pa1.9e



b)



c)

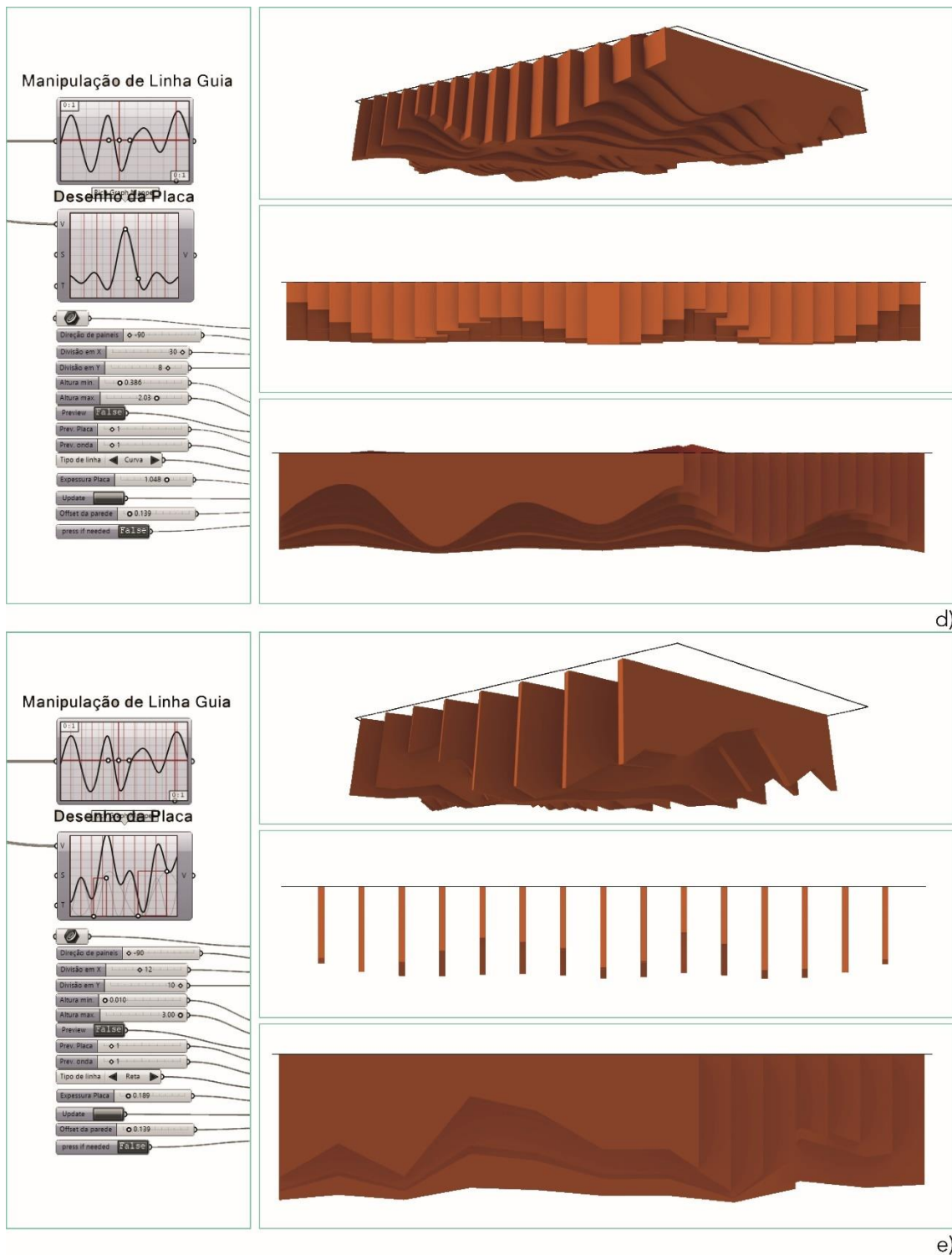


Fig. 69 | Variantes de solução gerada (Imagem do autor): a) Variante de solução 1; b) Variante de solução 2; c) Sistema grasshopper gerador do Objeto; d) Variante de solução 3; e) Variante de solução 4.

6.4 | Processos Alternativos de Trabalho: Inteligência Artificial (IA)

05 | Processos Alternativo: Renders

Análise preliminar do processo manual e natureza das restrições

O caso de estudo para o qual é proposto o seguinte processo alternativo procura responder a situações em que existe a necessidade de criar ou fabricar imagens, quer seja na exploração da ideia ou para visualização do cliente do projeto em que investe. Assim, foram simulados três possíveis casos de trabalho com base no contexto do atelier: Esboço manual para imagem digital; Melhoramento de imagem digital semi-produzida; Renderização a partir de volumes simplificados. Deste modo, o desenvolvimento deste processo de trabalho obrigou o entendimento do algoritmo e da teoria por detrás destes softwares, em que para facilitar o controlo sobre a IA, optou-se por softwares user-friendly especializados na produção de imagens em arquitetura, tendo sido testados: Lookx AI, rendair e Prome AI.

Organograma de metodologia de uso da IA

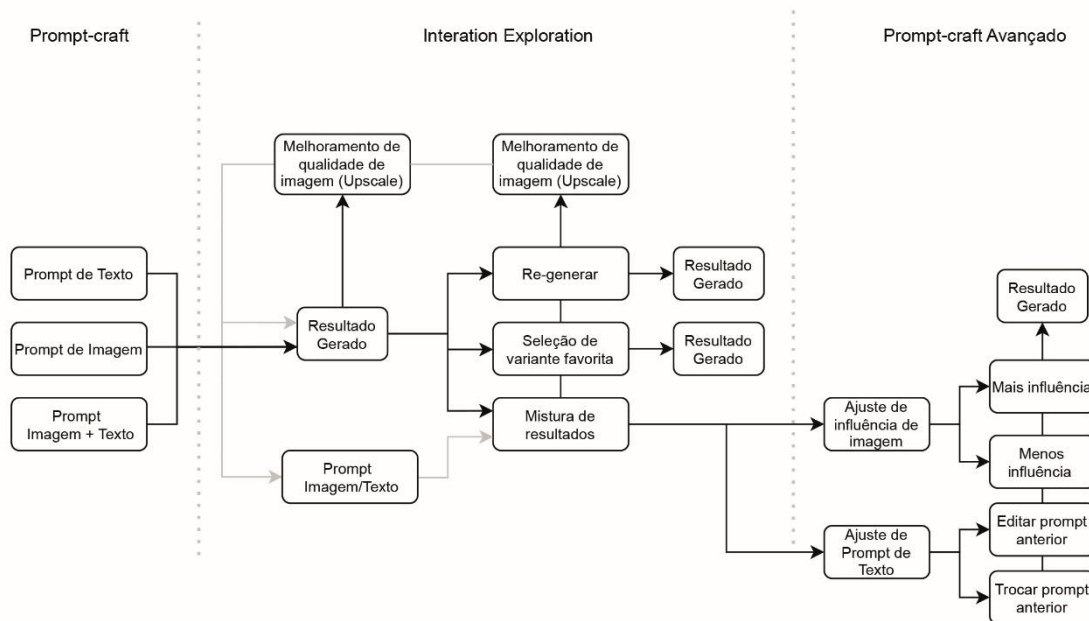


Fig. 70 | Organograma processo alternativo 05 | Renders (Imagem do autor)

Construção do processo alternativo

Esboço manual para imagem digital

A partir de um esboço desenhado pelo arquiteto Tomás Taveira, tentou-se reproduzir o seu estilo arquitetónico pós-modernista, reconhecido pela explosão de cores e uso de volumes geometrizados. Este desenho (fig.68) foi introduzido nos três programas no final e abordado com o mesmo prompt escrito e prompt visual, no entanto cada um acabou por gerar e promover diferentes processos de trabalho.

O primeiro teste foi no software Lookx AI, onde foram verificadas algumas dificuldades em obter resultados consistentes e controlados. Uma situação proveniente, sem qualquer tipo de margem para dúvidas, da inexperiência do autor da dissertação com este tipo de ferramentas, no entanto desta exploração “caótica” foi possível obter resultados inesperados (ou emergentes) que se demonstraram bastante interessantes e úteis.

De qualquer modo, uma das ferramentas que este software oferece é o treino de modelos LoRA (Low-Rank Adaptation)⁷, que foi treinado com uma série de imagens de projetos de referência do arquiteto para que a IA tivesse informação de referência quando o pedido fosse requisitado. Este software permite a produção de imagens de dois modos: *Explore Mode* - produção exploratória de imagem via os prompts fornecidos (imagem ou escrito); *Render Mode* - transformação de uma imagem base num render de alta resolução. A primeira abordagem (fig.71, a) de geração de imagem foi com base em três inputs (parâmetros) em Render Mode: o esboço em questão; um prompt explicativo que descreve o estilo arquitetónico do arquiteto Tomás Taveira gerado pelo ChatGpt (Comunivisão pessoal, junho 13, 2024); uma imagem de referência do edifício das Amoreiras. No entanto, os resultados obtidos não foram satisfatórios, partindo-se para uma re-tentativa, mas desta vez com uma redução do prompt a três palavras obtendo um resultado mais promissor (fig.71, b), mas longe do pretendido. Foi testado em Paralelo o Explore Mode utilizando o mesmo esboço, o processo gerou uma série e soluções pela variação dos parâmetros introduzidos. Um aspeto desta testagem que produziu a significativa diferenciação de resultados foi a mudança do algoritmo que computa a solução (Fig.72).

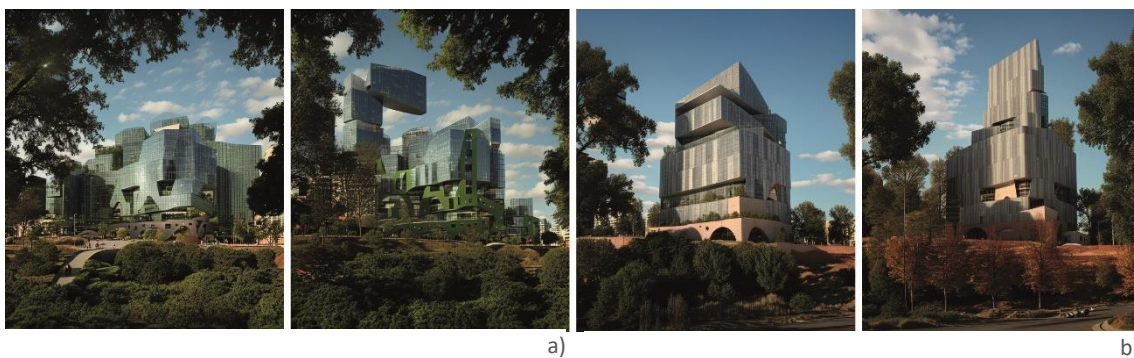
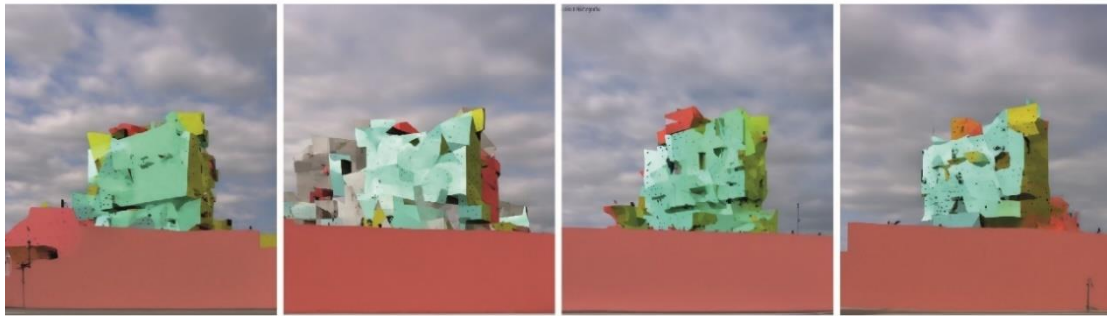


Fig. 71 | a) Primeira abordagem; b) Segunda abordagem. (Imagem do autor)

⁷ LoRA ou *Low-Rank Adaptation*, é um método de refinamento de resultados apresentado pela equipa investigadores da Microsoft em 2021. Desde então, tornou-se uma abordagem popular para refinamento de grandes modelos de linguagem (LLM), modelos de difusão (*Stable-Diffusion Models*)(como os de geração de imagens) e outros tipos de modelos de IA.

Content Control



a)

Segmentation



b)

Scribble_xdog



c)

MIsd



d)

Fig. 72 | Tipos de algoritmo testados (Imagens do autor): a) Content control; b) Segmentation; c) Scribble_xdog; d) MIsd. (www.semanticscholar.org)

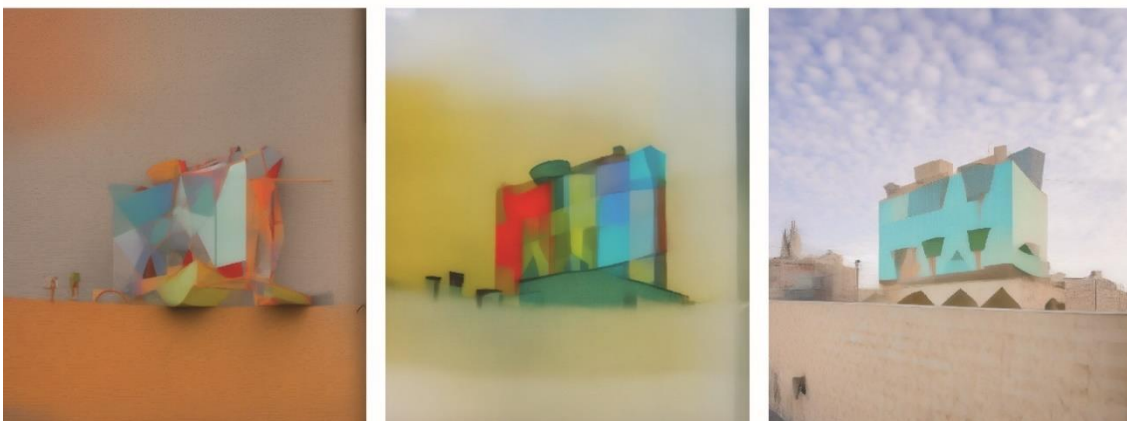
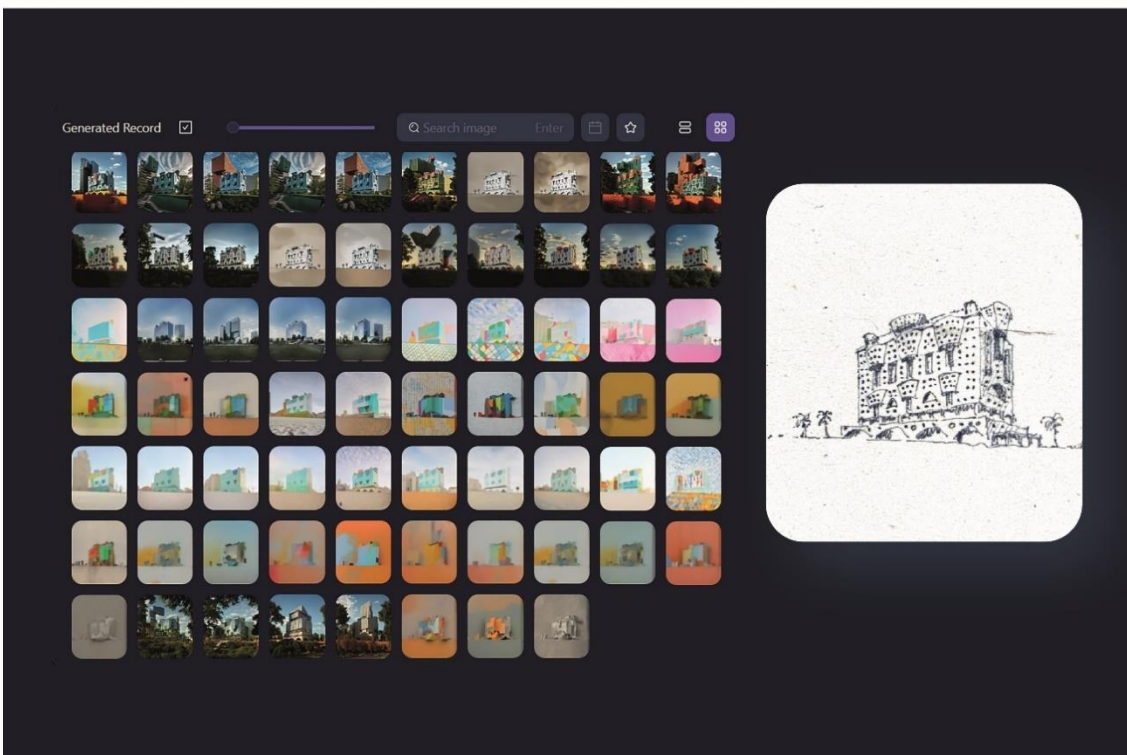
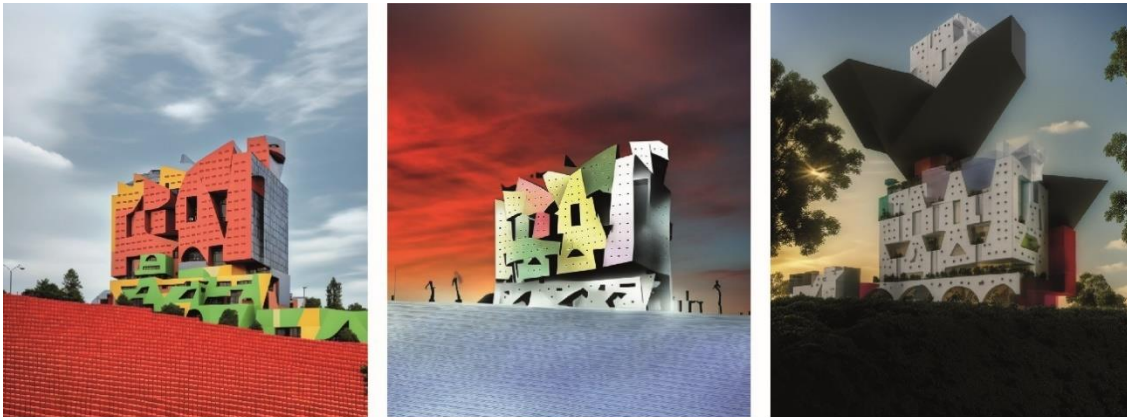


Fig. 73 | Recolha de resultados com maior interesse. Esta amostra pretende mostrar os bons e os resultados "caóticos" gerados do esboço base do arquiteto Tomás Taveira (Imagem do autor)

A recorrente testagem e obtenção de variadas soluções permitiu verificar um padrão onde a simplicidade do prompt escrito e a não modificação constante do algoritmo, permitia um controlo maior sobre o resultado. Conclusões obtidas por tentativa e erro pela adição e subtração de parâmetros consoante o resultado obtido. Facilmente verificamos na figura X, momentos de controlo com resultados consistentes sobre a ferramenta e por sua vez, as chamadas “alucinações” de IA, em que o output gerado afasta-se do objetivo⁸. No Explore Mode, a testagem “caótica” de parâmetros e análise crítica de resultados permitiu um melhor entendimento sobre o algoritmo, tendo-se decidido testar duas metodologias diferentes⁹:

A primeira metodologia através de *Prompt Crafting*, onde cada interação gera uma nova imagem sem relação com as gerações anteriores, um método focado em afinar os parâmetros que produzem o resultado pretendido, muito a base da testagem de palavras-chave específicas sem frases gramaticalmente corretas, e evitando palavras genéricas como Arquitetura ou Render no prompt escrito (fig.74).



Fig. 74 | Resultados de Prompt Crafting (Imagens do autor)

A segunda metodologia por *Interactive Design*, onde o utilizador ajusta os parâmetros fornecidos ao mesmo tempo que é substituída a imagem base, neste caso o esboço, pelas variantes geradas. A composição (fig.76), salienta a progressão do trabalho exploratório pelos outputs mais relevantes. Em paralelo, os resultados mais interessantes do *Explore Mode* foram testados no *Render Mode*, de modo a obter imagens que se aproximem da realidade (fig. 77), este processo foi mais focado em gerar resultados rápidos do que precisos, inclusive, voltou-se a testar e produzir o render do esboço original, mas desta vez com os prompts refinados provenientes das testagens por tentativa e erro (fig.80).

⁸ No caso de softwares como o ChatGpt diz-se que este alucina quando fornece uma resposta errada

⁹ Tentativa de aplicação das metodologias explicadas por Tim Fu numa entrevista feita para o canal de youtube BlessedArch: <https://www.youtube.com/watch?v=kOORzFcT080&t=1484s>



Fig. 75 | Resultados de Interactive Design (Imagens do autor)



Fig. 76 | Melhores resultados de Interactive Design testados no Render mode (Imagem de autor)

O segundo teste foi no software PromeAI, que ao contrário do anterior apresenta uma série de parâmetros (variável) ajustados por associação visual, facilitando a obtenção rápida de resultados próximos do pretendido, através da seleção desses modelos específicos que irão condicionar o algoritmo durante a computação dos resultados. Um aspecto do software que apesar de não depender do prompt escrito, não significa que o mesmo seja dispensável, inclusive foi utilizado o prompt que obteve os melhores resultados nas experiências anteriores (Fig.77). O resultado obtido, no entanto não foi totalmente satisfatório, tendo associado a falta de contexto no desenho como um muro que corta o edifício. Deste modo, existiu a necessidade de refinar o resultado mas em vez de se optar pela geração de uma nova imagem, foi usada uma ferramenta que permite eliminar e/ou trocar elementos específicos,

neste caso foi removido o muro e o software gerou o estacionamento e a entrada deste edifício (fig.78,b). O procedimento foi novamente repetido, mas desta vez para transformar os pilares quadrangulares em colunas amarelas (fig.78, c), este processo aplicado no PromeAI, em que o ajuste manual do resultado permitiu o refinamento do resultado (fig.78)



a)



b)

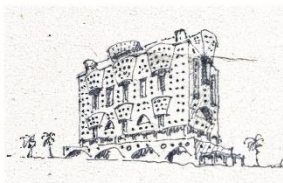


c)

LookX Ai

Prompt escrito:

Postmodernism building design, modern urban environment, mix of Geometric volumes, streets, cars, roads, people, colorfull building, building base with arch, concrete piramid facade Perspective.



d)



e)

Fig. 77 | Resultados obtidos com o software LookX AI (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido.



a)



b)

Rendair

Prompt escrito:

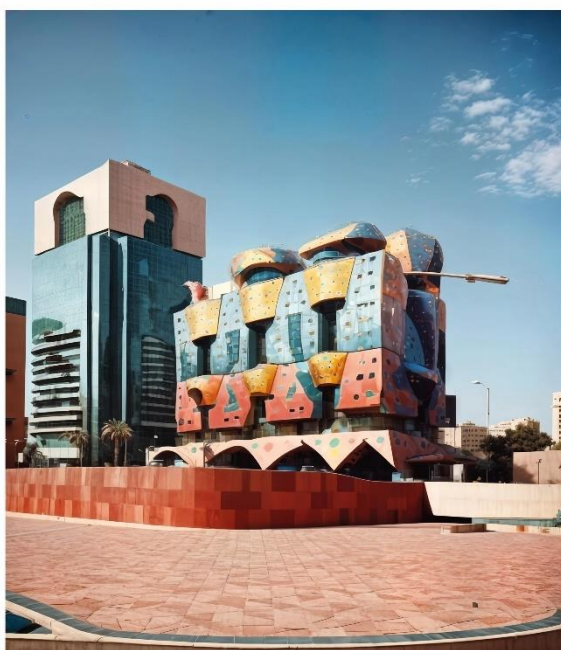
Postmodernism building design, modern urban environment, mix of Geometric volumes, streets, cars, roads, people, colorfull building, building base with arch, concrete piramid facade Perspective.



e)



d)



c)

Fig. 78 | Resultados obtidos com o software rendair (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido.



a)



b)

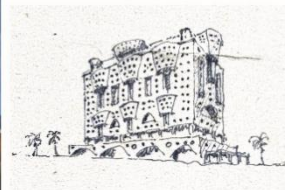


c)

PromeAI

Prompt escrito:

Postmodernism building design, modern urban environment, mix of Geometric volumes, streets, cars, roads, people, colorfull building, building base with arch, concrete piramid facade Perspective.



d)



e)

Fig. 78 | Resultados obtidos com o software PromeAI (Imagens do autor) a) Solução 1; b) Solução 2; c) Solução 3; d) Esboço guia de resultado; e) Estilo de inspiração fornecido.

Melhoramento de imagem digital semi-produzida

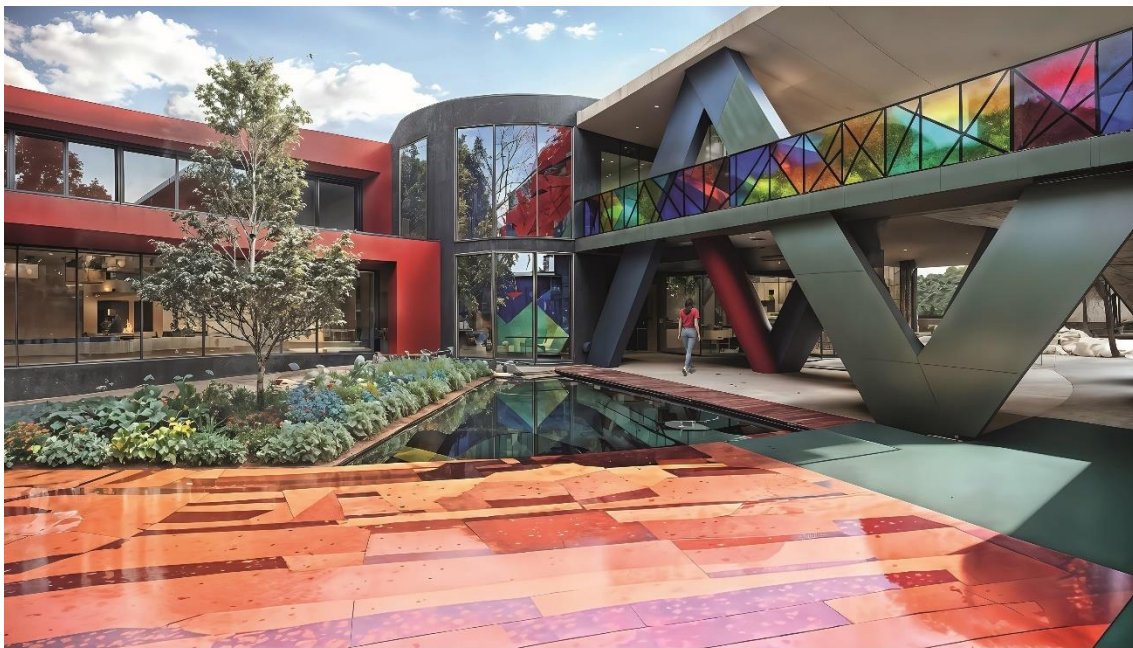
Para esta simulação de uma possível situação em ambiente de atelier, foi utilizado como objeto de estudo uma das primeiras abordagens ao projeto Maia Tech, onde por falta de tempo apenas foi possível atribuir texturas simplificadas no modelo digital sem qualquer tipo de aperfeiçoamento do ambiente transmitido na imagem muito superficialmente editada na altura no software Twinmotion. O primeiro resultado (fig.82 a) foi produzido a partir de uma breve descrição de intenções e materiais (prompt escrito), em que a imagem gerada numa primeira análise superficial realça uma série de características e qualidades do projeto, inclusive preencheu os espaços interiores que não tinham definição de função (fig.82 a), proporcionando uma visualização mais atrativa do mesmo. No entanto, algumas características e propriedades desapareceram ou foram modificadas, como as cores e materialidade da guarda da ponte e o quebrar da sua cobertura ou o pavimento no primeiro plano. Para além de que, quando feita uma análise mais pormenorizada do resultado obtido, verifica-se uma série de elementos e artefactos desfigurados, mas que na totalidade da imagem facilmente passam despercebidos e são associados ao que pretendem transmitir. Estas anomalias na imagem são provocadas, em parte, pela falta de distinção dos elementos representados na imagem, onde o uso de sombras ou a simulação de um céu azul com poucas nuvens dificulta a leitura da IA usada neste software sobre a imagem (parâmetro) fornecida.

De qualquer modo, o output obtido foi refinado via processos de adição e subtração, através do ajuste de elementos ou partes da imagem menos que produziram resultados insatisfatórios. Um processo de melhoramento do resultado que se verificou mais fácil e eficiente que a geração de novas imagens por afinamento de parâmetros, em que por exemplo para alterar apenas uma árvore (fig.82 d), não é necessário descrever a totalidade da imagem.

A seguinte composição de imagens pretende sumarizar o processo de tentativa e erro que permitiu gerar a imagem representada na figura 81, salientando os elementos ou partes da imagem alteradas e o que era pretendido.



a)



b)

Fig. 81 | Comparação de a) objeto de estudo (Imagens com direitos autorais de atelier TT) com b) resultado final (Imagem de autor).



Primeiro Passo

Prompt escrito:

Used of red green and blue on the varius elements of the building, a small water mirror side by side with small garden Street-level, coworking building.

a)



Ponte em vidro

Prompt escrito:

Colored glass balcany fence

b)



Eliminação de erros

Explicação:
Eliminação de artefactos indesejados criado gerados pela IA, por selecção dos mesmos.



c)



Alteração de árvore

Prompt escrito:
recessed wide windows with clear glass on a red slab



d)

Fig. 82 | Principais mudanças aplicadas por ordem alfabética (Imagens do autor): a), b), c) e d)

Renderização a partir de volumetria simplificadas

A seguinte simulação de uma possível situação de trabalho em ambiente de atelier, foi focada na procura de gerar uma série de hipóteses e variantes sobre uma ideia ou conceito não consolidada, onde o ajuste e refinamento individual é aplicado quando necessário. O processo de trabalho executado no software AI foi com base na experiência e conhecimentos adquiridos nos testes anteriores, com uma pequena variante, pois os objetos/volumes utilizados neste estudo foram gerados em Grasshopper pelo autor desta Dissertação, mais especificamente a partir do sistema paramétrico criado para o processo alternativo 04 | Processos de produção de estudo prévio - Proposta de múltiplos edifícios.



a)



b)



c)



d)



e)

Fig. 83 | Imagens geradas em render (Imagem do autor): a) Hipótese de solução 1; b) Variante 1 de solução 1a; c) Variante 2 de solução 1a; d) Hipótese de solução 2; e) Hipótese de solução 2.

6.5 | Síntese

Comparação de resultados

Capacidade	Tipos de processos		
	Caso de estudo	Modelação Paramétrica	Inteligência Artificial
Processo automatizado de geração de variantes de soluções.	NÃO	SIM	SIM
Adaptação rápida da mesma solução a diferentes configurações de objeto (parede curva vs parede plana).	NÃO	+/-	+/-
Produção automatizada de resolução de momentos de exceção.	NÃO	SIM	—
Tomada de decisão informada por testagem de soluções (metodologia de falhar rápido).	NÃO	SIM	SIM
Controlo sob o resultado produzido.	SIM	SIM	+/-
Potencializar o processo criativo durante o trabalho de exploração da ideia.	SIM	SIM	SIM
Constante compromisso de recursos na produção de variantes de solução não consolidada.	SIM	NÃO	NÃO
A testagem de variantes de soluções obriga a um novo modelo.	SIM	NÃO	NÃO
O tempo do processo de trabalho aumenta consoante os momentos de exceção.	SIM	SIM	—
Produção exaustiva pela repetição de procedimentos.	SIM	NÃO	NÃO
Maximização e otimização com precisão, de elementos e componentes do projeto.	NÃO	SIM	NÃO

Tabela 3 | Tabela comparativa de Casos estudados com Processos alternativos executados. (Imagem de autor)

Conclusões

A partir da tabela comparativa, facilmente verificamos que existem vantagens na implementação dos processos de projeto/design computacional, onde os mesmos foram capazes de responder às mesmas problemáticas que os processos convencionais. Inclusive estes processos alternativos foram capazes de colmatar as dificuldades e limitações dos processos convencionais na obtenção da solução, ao proporcionarem ao arquiteto uma grande capacidade de produção e exploração de soluções em diferentes fases de projeto. No entanto, é importante ter em conta que algumas dificuldades e limitações poderão se manter, ou simplesmente novas dificuldades irão surgir do uso destes processos.

No caso da modelação paramétrica, via programação por nodes, existe uma certa liberdade sobre como o problema pode ser abordado de diferentes formas, inclusive o processo e a análise da forma pretendida pode chegar a conclusões diferentes. Isto permite por exemplo a produção de dois sistemas paramétricos, com parâmetros e abordagens diferentes, que apresentam a mesma solução ou objeto, apenas o nível de computação poderá ser menos exigente num dos processos. Uma das questões que o programador deve ter em conta no uso da ferramenta e na criação do sistema, como o número de procedimentos e o nível de computação dessas operações. Pois caso a computação dos sistemas que gera a solução seja demasiado complicada ou pesada para o hardware disponível, poderá fazer com que a abordagem seja inviável, existindo diversas formas de contornar o problema, como por exemplo (fig.84) em vez de ser usado o componente *Regio Difference* num retângulo populado com círculos para obter uma superfície esburacada. Opta-se por um processo que têm mais passos, mas as operações são mais simples de computar, como pelo uso do *Surface split* para produzir o tal retângulo esburacado, mas que obriga a identificação do output pretendido, sendo neste caso identificado por ser o resultado na lista com o maior número de arestas/edges, que por sua vez corresponde último item no Index, após o *Sort List*.

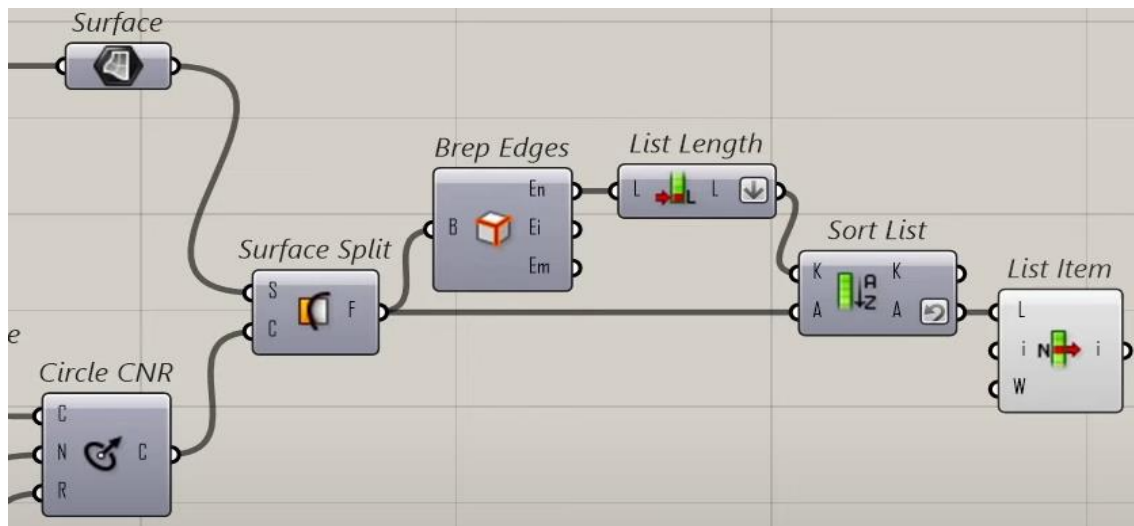


Fig. 84 | Processo alternativo para Regio Difference. (Imagem do autor)

Em suma não existe apenas um processo para obter a mesma solução, um facto curioso tendo em conta que um dos focos destas abordagens, e por sua vez uma das maiores vantagens nestes

processos de trabalho computacional, é a produção não de uma solução ou um objeto fixo mas sim a produção de um sistema que gera “infinitas” soluções.

Uma outra questão deste processo de trabalho, é o sistema construído, não ter capacidade para gerar novas hipóteses, a não ser que tenha sido criado para tal. Ou seja, de modo geral o sistema paramétrico irá ser desenhado para produzir variantes de uma solução para um problema (por exemplo, caso de estudo 02 | Hipótese de fachada 1 e 04 | Teto Acústico Escola de Dança), algo que por vezes, é confundido com hipóteses de solução. No entanto, como referido o sistema pode ser produzido para gerar diferentes hipóteses de solução, como por exemplo o caso de estudo 03 | Hipótese de fachada 2, em que o sistema gera soluções diferentes para momentos de exceção diferentes. Ou no caso de estudo 01 | Processos de produção de estudo prévio - Proposta de múltiplos edifícios, em que são geradas diferentes hipótese de edifícios cada uma com configurações próprias, no entanto para que tal seja possível, esta questão deve ser ponderada no início da construção do sistema paramétrico e na ideação do algoritmo do sistema.

A versatilidade do sistema para responder a diferentes problemas, ou sub-problemas derivados de momentos de exceção pode levar a um processo de construção do sistema mais demorado devido a avaliação das diferentes variáveis, facilmente exageradas na procura de resolver todos os problemas. Deste modo, é importante considerar se compensa a perda desse tempo, por exemplo no processo alternativo do caso de estudo 02 | Hipótese de fachada 1 é referido que o sistema não permite produzir soluções para cantos ou momentos de ligação de duas paredes de fachada. A conclusão a que se chegou foi que, para o contexto do trabalho em análise e o objetivo a atingir, esses momentos de canto por não serem uma situação frequente no projeto, iria compensar mais o ajuste manual do modelo nesses momentos do que a adaptação é possível reconstrução do sistema para ser adaptado a essas situações. Esta situação aconteceu com o processo do caso de estudo 02 | Hipótese de fachada 1, em que o sistema tem capacidade de apresentar hipóteses de solução para problemáticas que não existem na proposta de projeto feita em ambiente de atelier.

Por outro lado, uma das vantagens destes processos de trabalho é a possibilidade de reutilizar o sistema ou procedimentos, ou seja, um sistema consolidado pode voltar a ser utilizado em diferentes projetos que permitam a inserção dos inputs necessários para o funcionamento do algoritmo. Por exemplo, uma aplicação testada na modelação paramétrica, foi um sistema que produzia janelas com a moldura do caixilho quadrangular a partir de uma superfície, de modo a facilitar a obtenção de elementos básicos para visualizar rapidamente algo mais que um volume simples. Outra possibilidade é o reaproveitar de excertos do código, criando *Clusters* que executam procedimentos, por vezes repetidos, reduzindo o tempo de trabalho. Em relação a inteligência artificial é possível treinar a ferramenta para que produza imagens que se aproximam do estilo próprio do arquiteto, reduzindo a necessidade de constantemente refinar o algoritmo para que forneça o resultado pretendido. Vantagens das tecnologias de informação que requer que o arquiteto vá construindo a sua biblioteca de informação digital, de modo a otimizar o seu trabalho.

No entanto, apesar deste procedimento não ser difícil de executar e permitir gerar um trabalho refinado pelo intuito de quem o produz, o refinamento de elementos individuais das imagens pode-se tornar um processo de ajustes manuais apenas com produção automatizada de resultados, algo que em ambiente de atelier, caso o intuito seja obter imagens precisas e sem qualquer tipo de anomalia ou defeito, o uso de softwares de renderização e a aplicação dos processos convencionais de geração de imagem poderá ser uma melhor opção. Contudo, a proficiência do utilizador com este tipo de sistemas e/ou inclusivo o software utilizado poderá limitar a velocidade com que o resultado é produzido. Para

além de que, muitos destes softwares trabalham por ligação a servidores via internet, onde a velocidade da mesma e a afluência à aplicação poderão condicionar os resultados.

O tempo, como referido nesta dissertação, é uma questão que principalmente em contexto de trabalho deve ser tido em conta durante a escolha da ferramenta e/ou processo a implementar, de modo a respeitar as metas estabelecidas. Contudo o tempo como um parâmetro de avaliação foi excluído propositadamente, pois seria um dado incoerente de comparação de processos. Onde a proficiência no uso da ferramenta por quem realiza a tarefa, neste caso o autor desta dissertação, irá influenciar o tempo de produção necessário para obter a solução.

Salienta-se por exemplo, a produção do caso de estudo 03 | Hipótese de fachada 2 via processos convencionais, levou cerca de um dia de trabalho (8 horas) para completar a tarefa na integridade do projeto modelado em 3D via SketchUp. No entanto, o processo alternativo pelo uso de modelação paramétrica via Rhino levou, sem qualquer dúvida, muito mais tempo que o processo convencional (não havendo estimativa possível), a chegar ao sistema com as capacidades apresentadas para produzir a “mesma” solução, tendo registado a criação de doze ficheiros, cada um correspondente a um avanço significativo no sistema construído. Contudo, se compararmos o tempo que foi necessário para desenvolver os processos de trabalho dos casos de estudo 02 | Hipótese de fachada 1 (3h) e 04 | Teto Acústico Escola de Dança (5h), com o referido anteriormente, facilmente fica comprovado o impacto da experiência e conhecimento no uso destas ferramentas. Por este simples facto, o tempo se considerado como parâmetro de estudo comparativo, iria transmitir uma avaliação errada dos processos em questão de tempo de conclusão de tarefa, pois o autor desta dissertação no início da mesma, já estava mais familiarizado com o software SketchUp e os processos de trabalho e metodologias inerentes ao mesmo, que os processos de modelação paramétrica pelo uso de programação visual (node).

O mesmo se aplica nos softwares IA que produzem resultados aceitáveis e úteis, via procedimentos que não são propriamente difíceis de executar, inclusive permitir gerar um trabalho refinado pelo intuito de quem o produz. No entanto, o refinamento de elementos individuais das imagens pode-se tornar um processo de ajustes manuais apenas com produção automatizada de resultados, algo que em ambiente de atelier, caso o intuito seja obter imagens precisas e sem qualquer tipo de anomalia ou defeito, o uso de softwares de renderização e a aplicação dos processos convencionais de geração de imagem poderá ser uma melhor opção. Contudo, a proficiência do utilizador com este tipo de sistemas e/ou inclusivo o software utilizado poderá limitar a velocidade com que o resultado é produzido. Para além de que, muitos destes softwares trabalham por ligação a servidores via internet, onde a velocidade da mesma e a afluência à aplicação poderão condicionar os resultados.

Deste modo, o uso destas ferramentas requer treino e masterização para poder produzir soluções e sistemas que geram soluções rapidamente, a diferentes níveis de complexidade. Sendo errado pensar que os processos computacionais garantem a obtenção dos resultados pretendidos facilmente, pois mesmo as IA que são publicitadas como tal, requerem a compreensão do algoritmo e o modo como este funciona para obter resultados refinados. O simples facto do uso da ferramenta ser bastante versátil em que todas as abordagens são válidas e dependem do contexto de trabalho e dos objetivos. Ou seja, a ferramenta pode ser utilizada simplesmente para resolver o problema momentâneo de projeto (abordagem de arquiteto), ou para criar um sistema com um certo nível de abstratividade para que a solução possa ser implementada em diferentes objetos com diferentes configurações (neste caso sistemas que geram objetos em objetos diferentes), como o gerar de uma solução numa parede plana como uma parede curva (abordagem de programador), ou levar o sistema produzido mais além ao desenvolver uma aproximação de uma interface específica com planeamento de utilização (abordagem de software designer).

No final são ferramentas da era da informação que dão ao arquiteto um maior controlo sobre as decisões tomadas, quer seja pela análise de diversas hipóteses mais ou menos aproximadas com a realidade (05 | Caso de estudo: Renders, 01 | Processos de produção de estudo prévio - Proposta de múltiplos edifícios), ou pela otimização de uma solução (01 | Processos de produção de estudo prévio - Otimização de área), enquanto a última palavra que prevalece ,ou decisão final ainda for a do arquiteto, não se pode dizer que o que a máquina gerou não é do sujeito que a controlou.





7. CONSIDERAÇÕES FINAIS

Considerações Finais

O trabalho realizado aborda o impacto e as possibilidades da modelação paramétrica e da geração de imagens por IA na arquitetura a partir da reflexão propiciada por uma experiência de trabalho em atelier em que essas ferramentas não eram usadas. Partindo de uma análise abrangente dos conceitos teóricos, metodologias práticas e estudos de caso, destacamos como essas tecnologias estão a transformar o processo de design arquitetónico. Ao longo da pesquisa, ficou claro que a modelação paramétrica oferece uma abordagem inovadora, permitindo aos arquitetos superar os limites dos processos de trabalho convencionais. Onde as ferramentas computacionais possibilitam a criação de estruturas mais complexas e eficientes, otimização de recursos e maior precisão na execução dos projetos. Exemplos reais de arquitetos de referência, como Zaha Hadid e Frank Gehry, mostram como a adoção destas tecnologias permite desenvolver projetos arquitetónicos inovadores.

A introdução da IA na arquitetura, especialmente na geração de imagens e na automação do projeto, abre novas fronteiras para a criatividade e eficiência no trabalho dos arquitetos. A IA não só amplia as capacidades dos profissionais, mas também traz novas formas de abordar as problemáticas de projeto, permitindo maior liberdade durante o processo criativo. No entanto, é crucial reconhecer os desafios e limitações destas inovações, que requerem treino e masterização de modo a obter resultados controlados. Além disso, questões éticas relacionadas ao uso de IA, como direitos de autor e transparência sobre a informação que treina os algoritmos, são questões que irão ser (eventualmente) consideradas no desenvolvimento das mesmas.

A revisitação das experiências práticas, experienciadas em primeira mão, no contexto de trabalho em ambiente de atelier mostraram que a integração de métodos computacionais pode melhorar o fluxo de trabalho e promover a exploração criativa tão valorizada por esta profissão. Em que, o combinar da criatividade humana com a precisão das máquinas, quando realizada de uma forma consciente e educada, oferece uma série de novas abordagens ao projeto, que irão sem qualquer dúvida moldar o futuro da disciplina e da profissão.

Em conclusão, esta dissertação destaca a importância de uma avaliação equilibrada e crítica das novas tecnologias na arquitetura, em particular a modelação paramétrica e a IA. Um trabalho que não pretende convencer aqueles que o leem a adotar de uma forma acrítica estas ferramentas, mas sim motivar a procura de novas possibilidades de trabalho em que as conclusões sobre as mesmas devem resultar da experiência real e prática avaliada em contexto, e não de especulações baseadas em preconceitos, por vezes arrogantes, sobre o desconhecido.





8. BIBLIOGRAFIA

8 | Bibliografia

30X40 Design Workshop. (2023, May 30). *Using AI as a design tool in my architecture practice* [Video]. YouTube. <https://www.youtube.com/watch?v=zhKb9l6LBS0>

Abideen, Z. U. (2023, June 26). Reinforcement Learning from Human Feedback (RLHF): Empowering ChatGPT with User Guidance. *Medium*.
<https://medium.com/@zaiinn440/reinforcement-learning-from-human-feedback-rlhf-empowering-chatgpt-with-user-guidance-95858592fdbb>

Ahlquist, S., & Menges, A. (2011). Introduction. In S. Ahlquist & A. Menges (Eds.), *Computational Design Thinking* (pp. 10-29). Wiley.

AIA round-table discussion. (2014, August). *ar-The Architectural Review*, 1410, 101-105.

Alexander, C. (2011). Systems Generating Systems. In S. Ahlquist & A. Menges (Eds.), *Computational Design Thinking* (pp. 58-68). Wiley. (Original work published 1968)

Algorithmix Design. (2023, June 5). *AI-Powered Skyscraper design: 20 RENDERS in 5 mins! (VERAS for Rhino)* [Video]. YouTube. <https://www.youtube.com/watch?v=VWde4kBGmal>

Aligning grid to a edge of polygon(Pls help me). (2022, May 7). McNeel Forum.
<https://discourse.mcneel.com/t/aligning-grid-to-a-edge-of-polygon-pls-help-me/142243/3>

Animal Welfare Research Network. (2024, April 3). *Computational approaches and AI in animal welfare science*. <https://awrn.co.uk/event/computational-approaches-and-ai-in-animal-welfare-science/>

Apd, R. (2024, January 3). *Os 4 tipos de inteligência artificial que devemos conhecer*. APD Portugal. Retrieved February 22, 2024, from <https://www.apd.pt/4-tipos-de-inteligencia-artificial-que-devemos-conhecer/>

Arch Viz Artist. (2023, September 13). *Get better results with AI by using stable diffusion for your arch Viz projects!* [Video]. YouTube. <https://www.youtube.com/watch?v=4Na4JOgX7Yc>

Architectures. (2023a, September 27). *¿Cuáles son los beneficios de ARCHITEChTURES?* Architectures. Retrieved December 9, 2023, from <https://architectures.com/es/blog/posts/cuales-son-los-beneficios-de-architectures?v=4302a5f4ed53>

Architectures. (2023b, September 28). *Diseñando con IA: Diseño Auto + Diseño Manual*. Architectures. Retrieved December 8, 2023, from

<https://architectures.com/es/blog/posts/disenando-con-ia-diseno-auto-diseno-manual?v=4302a5f4ed53>

Architectures. (2023c, September 28). *¿Qué es ARCHITECTURES?* Architectures. Retrieved December 9, 2023, from <https://architectures.com/es/blog/posts/que-es-architectures?v=4302a5f4ed53>

Architectures. (2023d, September 28). *¿Qué es el Diseño Asistido por Inteligencia Artificial?* Architectures. Retrieved December 9, 2023, from <https://architectures.com/es/blog/posts/que-es-el-diseno-asistido-por-inteligencia-artificial?v=4302a5f4ed53>

Ashby, M. (2020). Ethical regulators and Super-Ethical systems. *Systems*, 8(4), 53. <https://doi.org/10.3390/systems8040053>

Aussie BIM Guru. (2020, June 17). *Generative design in Rhino 3D (+Rhino Inside)!* [Video]. YouTube. <https://www.youtube.com/watch?v=7OeICkk-IYk>

Bansal, S. (2020, October 2). *The yin & yang of AI*. <https://www.linkedin.com/pulse/yin-yang-ai-supriya-bansal/>

Barker, N. (2024, January 22). You are now entering the Altopia. *Dezeen*. Retrieved April 23, 2024, from <https://www.dezeen.com/2023/06/19/aitopia-artificial-intelligence-series/>

Bianca, R. (2017, March 1). *Do Projeto Paramétrico ao Projeto Generativo - Parte 1: O que é projeto generativo | Mundo AEC - Autodesk*. Mundo AEC - Autodesk. <https://blogs.autodesk.com/mundoaec/projeto-generativo-parte-1/>

BIM A+. (2018, December 13). *The very beginning of the digital representation – Ivan Sutherland Sketchpad*. <https://bimaplus.org/news/the-very-beginning-of-the-digital-representation-ivan-sutherland-sketchpad/>

Bob Ross Quote: "All you need to paint is a few tools, a little instruction, and a vision in your mind." (n.d.). <https://quotefancy.com/quote/1391856/Bob-Ross-All-you-need-to-paint-is-a-few-tools-a-little-instruction-and-a-vision-in-your>

Browne, M. (2023, July 11). *"Robots, DNA, and Future Careers/ Unveiling the possibilities."* <https://www.linkedin.com/pulse/robots-dna-future-careers-unveiling-possibilities-mike-browne/>

Burry, M. (2011). Architecture and Pratical Design Computation. In S. Ahlquist & A. Menges (Eds.), *Computational Design Thinking* (pp. 102-119). Wiley.

Burry, M., Burry, J., & Faulí, J. (2001a). *Sagrada Família Rosassa: Global Computeraided Dialogue between Designer and Craftsperson (Overcoming Differences in Age, Time and Distance)*. <https://api.semanticscholar.org/CorpusID:54654192>

Burry, M., Burry, J., & Faulí, J. (2001b). *Sagrada Família Rosassa: Global Computeraided Dialogue between Designer and Craftsperson (Overcoming Differences in Age, Time and Distance)*. <https://www.semanticscholar.org/paper/Sagrada-Fam%C3%ADlia-Rosassa%3A-Global-Computeraided-and-Burry-Burry/ff685235ed8b6d789f9950ff7e4e387e927b78b1/figure/8>

Burton, D. (2020, July 31). *The scientific and technological advances of World War II*. The National WWII Museum | New Orleans. <https://www.nationalww2museum.org/war/articles/scientific-and-technological-advances-world-war-ii>

Carl, P. (2011). Type, Field, Culture, Praxis: Projective cities. In C. C. M. Lee & S. Jacoby (Eds.), *Typological urbanism: Vol. 81 No 1*. Academy Press.

Carl Sandburg quote. (n.d.). Lib Quotes. <https://libquotes.com/carl-sandburg/quote/lbi8b7d>

Carlson, C. (2023, August 17). AI design could “bring back the beauty and aesthetics of the classical era” says Tim Fu. *Dezeen*. Retrieved April 23, 2024, from <https://www.dezeen.com/2023/08/16/ai-aesthetics-classical-era-tim-fu-aitopia/>

Catalá, A. P. (2010). Numerical Ecosystems. In K. Lydia & H. Castle (Eds.), *EcoRedux: EcoRedux: Remedies for an Ailing Planet: Vol. 80 No 6* (pp. 86–93). John Wiley & Sons.

Catia V1 on an IBM station in 1982. Didier Gaillard gives a demonstration. The screen on the right is used to display geometric information on the part and its characteristics such as the center of gravity & the mass. [Online forum post]. (n.d.). Reddit. https://www.reddit.com/r/EngineeringPorn/comments/u4vjpi/catia_v1_on_an_ibm_station_in_1982_didier/?rdt=47603

Chadha, T. (n.d.). *The influence of AI on architectural visualization*. CUUB Studio. Retrieved April 13, 2024, from <https://www.cuubstudio.com/blog/the-influence-of-ai-on-architectural-visualization/>

Change true to false and false to true. (n.d.). Grasshopper. <https://www.grasshopper3d.com/forum/topics/change-true-to-false-and-false-to-true>

Chaturvedi, V. a. P. B. M. (2020, June 8). *Third and higher order cybernetics*. Viewpoints Which Matter. <https://chaturvedimayank.wordpress.com/2020/01/12/third-and-higher-order-cybernetics/>

Che, R. (2010, December 25). *inside Sagrada Familia - hanging model thought by Gaudi for the design of the Colònia Güell*. Flickr.

<https://www.flickr.com/photos/57908463@N00/5290017381>

Check if points in a list are contained in another list. (2022, October 22). McNeel Forum.

<https://discourse.mcneel.com/t/check-if-points-in-a-list-are-contained-in-another-list/149510>

Cianciolo, B., & Chia, J. (2023a, September 5). Opinion: Here are the jobs AI will impact most. CNN. Retrieved April 23, 2024, from <https://edition.cnn.com/2023/09/05/opinions/artificial-intelligence-jobs-labor-market/index.html>

Cianciolo, B., & Chia, J. (2023b, September 5). *Opinion: Here are the jobs AI will impact most* (By R. Anadol). CNN. <https://edition.cnn.com/2023/09/05/opinions/artificial-intelligence-jobs-labor-market/index.html>

COE.INT. (n.d.). *History of Artificial Intelligence: Artificial Intelligence*. Council of Europe. Retrieved December 21, 2023, from <https://www.coe.int/en/web/artificial-intelligence/history-of-ai>

Colomina, B., Choi, E., Galan, I., & Anna-Maria Meister. (2012, October). Radical pedagogies. *ar-The Architectural Review*, 1388, 78-81.

Complex systems and clouds. (n.d.). ScienceDirect.

<https://www.sciencedirect.com/book/9780128040416/complex-systems-and-clouds#book-description>

Contribuidores da Wikipédia. (2023a, April 26). *Demografia*.

<https://pt.wikipedia.org/wiki/Demografia>

Contribuidores da Wikipédia. (2023b, October 19). *Pontilhismo*. Retrieved April 13, 2024, from <https://pt.wikipedia.org/wiki/Pontilhismo>

Contribuidores da Wikipédia. (2024, June 13). *ENIAC*. <https://pt.wikipedia.org/wiki/ENIAC>

Contribuidores dos projetos da Wikimedia. (2023, January 2). *Carl Sandburg*. Wikiquote.

https://pt.wikiquote.org/wiki/Carl_Sandburg

Count amount of number combinations. (2019, July 18). McNeel Forum.

<https://discourse.mcneel.com/t/count-amount-of-number-combinations/86395>

- Counting the number of branches in a tree branch.* (2021a, June 9). McNeel Forum.
<https://discourse.mcneel.com/t/counting-the-number-of-branches-in-a-tree-branch/125671>
- Counting the number of branches in a tree branch.* (2021b, June 9). McNeel Forum.
<https://discourse.mcneel.com/t/counting-the-number-of-branches-in-a-tree-branch/125671>
- Cox, H. (2018, November 28). *Cracking stuff: how Turing beat the Enigma*. MANCHESTER 1824 the University of Manchester. Retrieved February 21, 2024, from
<https://www.mub.eps.manchester.ac.uk/science-engineering/2018/11/28/cracking-stuff-how-turing-beat-the-enigma/>
- Craven, J. (2019, August 1). *The meaning of ‘Form follows function’*; ThoughtCo.
<https://www.thoughtco.com/form-follows-function-177237>
- Create Range Slider - Human UI - component for Grasshopper | Grasshopper Docs.* (n.d.). Grasshopper Docs.
<https://grasshopperdocs.com/components/humanui/createRangeSlider.html>
- Cronologia do Pensamento Urbanístico. (n.d.). *Realizada no MoMA a Exposição “Architecture without Architects”, por Bernard Rudofsky*. Cronologia Do Pensamento Urbanístico. Retrieved February 3, 2022, from
<https://cronologiadourbanismo.ufba.br/apresentacao.php?idVerbete=1268>
- Crook, L. (2023, August 22). “I don’t think AI knows what humans like” says Daniel Escobar. *Dezeen*. Retrieved April 23, 2024, from <https://www.dezeen.com/2023/08/22/online-lab-architecture-interview-daniel-escobar-aitopia/>
- Crook, L., & Crook, L. (2023, June 6). “Architecture without architects” can teach valuable lessons says Norman Foster. *Dezeen*. <https://www.dezeen.com/2023/05/19/architecture-without-architects-valuable-lessons-norman-foster/>
- Crypto Museum. (2024, January 15). *Bombe*. <https://www.cryptomuseum.com/crypto/bombe/>
- Cui, J., & Tang, M. X. (2017). Towards generative systems for supporting product design. *International Journal of Design Engineering*, 7(1), 1. <https://doi.org/10.1504/ijde.2017.085639>
- Curve Evaluation using existing points.* (n.d.). Grasshopper.
<https://www.grasshopper3d.com/forum/topics/curve-evaluation-using-existing-points>
- Curve.Trim method (Double, double).* (n.d.). https://mcneel.github.io/rhinocommon-api-docs/api/RhinoCommon/html/M_Rhino_Geometry_Curve_Trim_2.htm

Cvetković, D. (2023, August 30). *Drafting Tools for Architects: Your Essential guide*. How to Rhino | Tutorials for Architects. <https://howtorhino.com/blog/architecture-education/drafting-tools/>

DamiLee. (2023, March 19). *We tried to compete with AI. . . [AI vs. ARCHITECT]* [Video]. YouTube. <https://www.youtube.com/watch?v=N709ZrxoIP0>

Davis, D. (2021a, January 4). *Chapter 2 - The Challenges of Parametric Modelling*. Daniel Davis. <https://www.danieldavis.com/thesis-ch2/>

Davis, D. (2021b, January 5). *Academic publications*. Daniel Davis. <https://www.danieldavis.com/publications/>

Davis, D. (2022, August 31). *A history of parametric*. Daniel Davis. <https://www.danieldavis.com/a-history-of-parametric/>

DCO Parametric. (2022, November 18). *Parametric / Algorithmic Architecture - Multi-floor building w/ Structure using Rhino & Grasshopper* [Video]. YouTube. <https://www.youtube.com/watch?v=NaUwy5s4n5s>

De Molina Rius, A. D. (2023). Foundations of artificial intelligence and machine learning. In N. Remolina (Ed.), *Artificial Intelligence in Finance: Challenges, Opportunities and Regulatory Developments* (pp. 2-18). Edward Elgar Publishing. <https://doi.org/10.4337/9781803926179.00009>

Design Input. (2023a, March 24). *Create Realistic Render from Sketch Using AI (you should know this. . .)* [Video]. YouTube. <https://www.youtube.com/watch?v=nFBVy73UnJg>

Design Input. (2023b, April 11). *This will change EVERYTHING in architectural visualization FOREVER!* [Video]. YouTube. <https://www.youtube.com/watch?v=9oQt6sg-nFs>

Design Input. (2023c, July 9). *How I Created This Render From Google Maps with AI (sketch into photo)* [Video]. YouTube. <https://www.youtube.com/watch?v=9WXqMRAGyPo>

Detect that one closed curve is inside another closed curve. (2020, December 30). McNeel Forum. <https://discourse.mcneel.com/t/detect-that-one-closed-curve-is-inside-another-closed-curve/115331/9>

Devsupport. (2020, May 5). *The history of computer-aided design and computer-aided manufacturing (CAD/CAM)*. Technical Foam Services. <https://technicalfoamservices.co.uk/blog/blog-history-of-cad-cam/>

Dicio, Dicionário Online de Português. (n.d.). *pontilhismo*. Dicio, Dicionário Online De Português. Retrieved April 13, 2024, from <https://www.dicio.com.br/pontilhismo/>

Divide to fit specified surface. (2020, August 31). McNeel Forum. <https://discourse.mcneel.com/t/divide-to-fit-specified-surface/107862/5>

Dividing a trimmed surface. (n.d.). Grasshopper. <https://www.grasshopper3d.com/forum/topics/dividing-a-trimmed-surface>

Dos Reis, D. (n.d.). *Edifício Guitarra Portuguesa - Antiga Sede do BNU by Dias dos Reis*. PBase. <https://pbase.com/diasdosreis/image/165641539>

Easy question: Scaling pre-made curve to grasshopper made box. (2020a, April 14). McNeel Forum. <https://discourse.mcneel.com/t/easy-question-scaling-pre-made-curve-to-grasshopper-made-box/100075>

Easy question: Scaling pre-made curve to grasshopper made box. (2020b, April 14). McNeel Forum. <https://discourse.mcneel.com/t/easy-question-scaling-pre-made-curve-to-grasshopper-made-box/100075>

Esmaeilian Toussi, H. (2020). *The application of evolutionary, generative, and hybrid approaches in architecture design optimization* (Vol. 2).

Estação Das Olaias (Metro de Lisboa) - Zagope - Construções e Engenharia S.A. (n.d.). Zagope - Construções E Engenharia S.A. <https://www.zagope.pt/portfolio/estacao-das-olaias-metro-de-lisboa/>

EvolveLAB. (2018, March 24). *Computational Design vs. Generative Design vs. Parametric Modeling* [Video]. YouTube. <https://www.youtube.com/watch?v=Fe64EMAt5zc>

Exteriors < FRAME studio Architectural visualizations and animations. (n.d.). <http://framestudio.eu/3d-visualizations/exteriors-2/>

Farrow, E. (2019). To augment human capacity—Artificial intelligence evolution through causal layered analysis. *Futures*, 108, 61–71. <https://doi.org/10.1016/j.futures.2019.02.022>

Fillet curve at a parameter. (2020, May 3). McNeel Forum. <https://discourse.mcneel.com/t/fillet-curve-at-a-parameter/101376/4>

Finding same geometries from two lists. (n.d.). Grasshopper. <https://www.grasshopper3d.com/forum/topics/finding-same-geometries-from>

- Fitzpatrick, L. (2021, March 25). *Life in Barcelona: Mark Burry - The Gaudí Code*. (barcelona-metropolitan.com). <https://www.barcelona-metropolitan.com/features/life-in-barcelona-the-gaud%C3%AD-code/>
- Fjelland, R. (2020). Why general artificial intelligence will not be realized. *Humanities & Social Sciences Communications*, 7(1), 10. <https://doi.org/10.1057/s41599-020-0494-4>
- Frazer, J., & Frazer, J. (1998). The Groningen Experiment: Architecture as an Artificial Life Form. In M. Toy & N. Spiller (Eds.), *Architectural Design: Architects in Cyberspace II* (Vols. 68, 11-12, pp. 12-15). John Wiley & Sons.
- Frazer, J., & Rastogi, M. (1998). The New Canvas. In M. Toy & N. Spiller (Eds.), *Architectural Design: Architects in Cyberspace II* (Vols. 68, 11-12, pp. 8-11). John Wiley & Sons.
- Fry, C. (1980). [Review of Artificial Intelligence, by P. Winston]. *Computer Music Journal*, 4(3), 61-63. <https://doi.org/10.2307/3679640>
- Ganesan, V. (2017, season-01). Comfortable in my own skin. *DA - Daylight & Architecture*, 27, 4-11.
- Garcia-Murillo, M., & MacInnes, I. (2019). *The impact of AI on employment: a historical account of its evolution*. <https://hdl.handle.net/10419/205178>
- Geddes, D. (2020, May 5). *The history of computer-aided design and computer-aided manufacturing (CAD/CAM)*. Technical Foam Services. <https://technicalfoamservices.co.uk/blog/blog-history-of-cad-cam/>
- Gillis, A. S. (2023, July 31). *algorithm*. WhatIs. <https://www.techtarget.com/whatis/definition/algorithm>
- Glover, E. (2024, April 2). *What is artificial intelligence (AI)?* Built In. <https://builtin.com/artificial-intelligence>
- Goldberg, M. (2022, July 21). What the terminator teaches us about AI and the need for better data. *AIThority*. <https://aithority.com/guest-authors/what-the-terminator-teaches-us-about-ai-and-the-need-for-better-data/>
- Golden AI. (2023, April 4). *Stable Diffusion*. GOLDEN. Retrieved April 13, 2024, from https://golden.com/wiki/Stable_Diffusion-NYXG8M4
- Grasshopper Points - TOI-Pedia*. (n.d.). http://wiki.bk.tudelft.nl/toi-pedia/Grasshopper_Points
- Guide to 3D Printing Architectural Models*. (n.d.). Formlabs. <https://formlabs.com/blog/3d-printing-architectural-models/>

Guinness, H. (2024a, April 17). *Midjourney vs. DALL·E 3: Which image generator is better?* [2024]. <https://zapier.com/blog/midjourney-vs-dalle/>

Guinness, H. (2024b, April 17). *Midjourney vs. DALL·E 3: Which image generator is better?* [2024]. <https://zapier.com/blog/midjourney-vs-dalle/>

Guinness, H. (2024c, April 17). *Stable Diffusion vs. DALL·E 3: Which image generator is better?* [2024]. <https://zapier.com/blog/stable-diffusion-vs-dalle/>

Hardware, W. (2023, October 10). *What is CAD (Computer-aided Design)?* Worthy Hardware. <https://www.worthyhardware.com/news/computer-aided-design/>

Hennings, M. (2023, November 4). *LoRA Fine-tuning & Hyperparameters Explained (in Plain English) | Entry Point AI*. Entry Point. Retrieved June 14, 2024, from <https://www.entrypointai.com/blog/lora-fine-tuning/>

Henriques, G. C., Bueno, E., Lenz, D., & Sardenberg, V. (2019). Generative Systems: Intertwining Physical, Digital and Biological Processes, a case study. *Blucher Design Proceedings*. https://doi.org/10.5151/proceedings-ecaadesigradi2019_100

Herr, C., & Fischer, T. (2001). Teaching Generative design. *ResearchGate*. https://www.researchgate.net/publication/30869860_Teaching_Generative_Design

Hintze, A. (2016, November 14). *Understanding the four types of AI, from reactive robots to self-aware beings*. The Conversation. Retrieved April 13, 2024, from <https://theconversation.com/understanding-the-four-types-of-ai-from-reactive-robots-to-self-aware-beings-67616>

Hobson, B., & Hobson, B. (2022, January 24). Watch Ross Lovegrove's keynote presentation live from Inside 2016. *Dezeen*. <https://www.dezeen.com/2016/11/18/video-live-stream-ross-lovegrove-inside-festival-2016/>

How do I connect curves into closed curves? (2018, July 12). McNeel Forum. <https://discourse.mcneel.com/t/how-do-i-connect-curves-into-closed-curves/66814>

How Renzo Piano Building Workshop creates architectural models with 3D printing. (n.d.). Formlabs. <https://formlabs.com/blog/renzo-piano-architectural-models-3d-printing/>

Howarth, D., & Howarth, D. (2022, January 14). Design schools need to embrace digital says Ross Lovegrove. *Dezeen*. <https://www.dezeen.com/2015/04/14/ross-lovegrove-design-schools-way-behind-need-embrace-digital-milan-2015/>

- Hunt, G. (1998). Architects in the "Cybernetic Age." In M. Toy & N. Spiller (Eds.), *Architectural Design: Architects in Cyberspace II* (Vols. 68, 11-12, pp. 52-55). John Wiley & Sons.
- Individualized Production in Architecture. (2016, November 17). *Grasshopper Tutorial 03 | Dividing curves* [Video]. YouTube. <https://www.youtube.com/watch?v=tJErbes0PCs>
- Ivo Ambrosi Architecture. (2020, November 10). *PURE PARAMETRIC ARCHITECTURAL SPACE by IVO AMBROSI* [Video]. YouTube. <https://www.youtube.com/watch?v=qsJi1rqr4I>
- Junichiro Horikawa. (2018, January 19). *[Grasshopper small tip] 0004 Better curve offset method* [Video]. YouTube. <https://www.youtube.com/watch?v=1BEDWWa-q8E>
- Kallipoliti, L. (2010). No More Shisms. In L. Kallipoliti & H. Castle (Eds.), *EcoRedux: Remedies for an Ailing Planet: Vol. 80 No 6* (pp. 14-23). John Wiley & Sons.
- Keep order of split surface the same as the offset curves.* (2021, October 24). McNeel Forum. <https://discourse.mcneel.com/t/keep-order-of-split-surface-the-same-as-the-offset-curves/132159>
- Kefalas, A. G. (2003a). Cybernetics. In *Elsevier eBooks* (pp. 365-378). <https://doi.org/10.1016/b0-12-227240-4/00024-1>
- Kefalas, A. G. (2003b). Cybernetics. In *Elsevier eBooks* (pp. 365-378). <https://doi.org/10.1016/b0-12-227240-4/00024-1>
- Kerner, S. M. (2023, April). *Dall-E*. TechTarget. Retrieved April 13, 2023, from <https://www.techtarget.com/searchenterpriseai/definition/Dall-E>
- Korn, J. (2023, January 18). *Getty Images suing the makers of popular AI art tool for allegedly stealing photos*. CNN. <https://edition.cnn.com/2023/01/17/tech/getty-images-stability-ai-lawsuit/index.html>
- Kory Bieg. (2020a, August 21). *Grasshopper tutorial: Divide surface* [Video]. YouTube. <https://www.youtube.com/watch?v=K7LTXRuFvXY>
- Kory Bieg. (2020b, August 22). *Grasshopper tutorial: Bounding box* [Video]. YouTube. <https://www.youtube.com/watch?v=EvGP0Ai4Otl>
- Kumar, S. (n.d.). *Product Design Sketches 1*. Behance. <https://www.behance.net/gallery/76522061/Product-Design-Sketches-1>
- Lab, M. (n.d.). *Data Types | The Grasshopper Primer Third Edition*. https://modelab.gitbooks.io/grasshopper-primer/content/1-foundations/1-2/3_data-types.html

- Latif, E., Mai, G., Nyaaba, M., Wu, X., Liu, N., Lu, G., Li, S., Liu, T., & Zhai, X. (2023). AGI: Artificial General Intelligence for Education. *arXiv (Cornell University)*, 1-19. <https://doi.org/10.48550/arxiv.2304.12479>
- L.com Team. (2023, March 24). *Definition of computer science, computational thinking and coding*. Learning. <https://www.learning.com/blog/definition-of-computer-science-computational-thinking-and-coding/>
- Maher, J. (2017, November 30). *The complete history of the IBM PC, part one: The deal of the century*. Ars Technica. <https://arstechnica.com/gadgets/2017/06/ibm-pc-history-part-1/>
- McKay, S. (2023, October 25). *The creative-right vs analytical-left brain myth: debunked!* - Dr Sarah McKay. Dr Sarah McKay. <https://drsarahmckay.com/left-brain-right-brain-myth/>
- Mitchell, W. J., & McCullough, M. (1995). *Digital Design Media*. John Wiley & Sons.
- Mitchell, W. J., & McCullough Malcom. (1995). *Digital Design Media*. Google Books. https://books.google.pt/books?hl=pt-PT&lr=&id=JrZoGgQEhKkC&oi=fnd&pg=PA2&dq=digital+design+media+a+guide+for+the+21st+century+best+quotes&ots=gmdm210e8i&sig=g24AxJ4kfLciPOG2GjBOLf1Zi2w&redir_esc=y#v=onepage&q&f=false
- Mosele, F. (n.d.). *Timeline of text-to-image Machine Learning models*. Fabian Mosele. Retrieved April 13, 2024, from <https://www.fabianmosele.com/ai-timeline>
- Musa_drm: *I will do your sketch to cad file or convert pdf to cad for >0 on fiverr.com*. (n.d.). Fiverr.com. https://www.fiverr.com/musa_drm/draw-your-hand-sketch-for-auto-cad
- Nagy, D. (2021, December 15). Working with DataTrees - Introduction to Grasshopper - Medium. *Medium*. <https://medium.com/intro-to-grasshopper/working-with-datatrees-179be560f086>
- Negromonte, M. (2015). Saber ver arquitetura Bruno Zevi. *Cesed*. https://www.academia.edu/17098588/Saber_Ver_Arquitetura_Bruno_Zevi
- Neves, I. C., Brandão, F., & Alves, V. (2023). A emergencia do projeto computacional em arquitetura. *J-A*, 264, 4-11.
- News - Refik Anadol Studio. (2024, February 29). Refik Anadol Studio. <https://refikanadolstudio.com/news/>
- Novak, M. (1998). Next Babylon, Soft Babylon. In M. Toy & N. Spiller (Eds.), *Architectural Design: Architects in Cyberspace II* (Vols. 68, 11-12, pp. 20-29). John Wiley & Sons.

Offset trim and fillet curves. (2019, January 24). McNeel Forum.

<https://discourse.mcneel.com/t/offset-trim-and-fillet-curves/77804>

Optimized Cellular Facade: Kinetic voronoi architecture. (2021, December 3). IAAC Blog.

<https://www.iaacblog.com/programs/kinetic-voronoi-facade-naderakoum/>

Optimizing Grasshopper definitions explained! (n.d.-a).

<https://www.shapediver.com/blog/optimizing-grasshopper-definitions-explained>

Optimizing Grasshopper definitions explained! (n.d.-b).

<https://www.shapediver.com/blog/optimizing-grasshopper-definitions-explained>

Oracle Cloud Infrastructure (OCI). (2021, March 5). *What Is Natural Language Processing*

(NLP)? Oracle. Retrieved April 12, 2024, from <https://www.oracle.com/pt/artificial-intelligence/what-is-natural-language-processing/>

Orient and Scale Objects along Surface. (n.d.-a). Grasshopper.

<https://www.grasshopper3d.com/forum/topics/orient-and-scale-objects-along?overrideMobileRedirect=1>

Orient and Scale Objects along Surface. (n.d.-b). Grasshopper.

<https://www.grasshopper3d.com/forum/topics/orient-and-scale-objects-along?overrideMobileRedirect=1>

Päivi. (2018, April 27). *Pointillism – a quick way, step by step!* Peony and Parakeet.

<https://www.peonyandparakeet.com/pointillism/>

Paramarch. (2016, July 7). *50 - Grasshopper - Surface box SBox Bounding box BBox Morph to generate morphed geometry* [Video]. YouTube.

<https://www.youtube.com/watch?v=mo2e9oFTVHs>

Parametric Philosophy. (2020, April 15). *Grasshopper Tips: How to Split and Trim (Trim with Region)* [Video]. YouTube.

<https://www.youtube.com/watch?v=ICLLRissfLw>

Parametric vs. Direct Modeling: Which Side Are You On? (2024, April 18).

<https://www.ptc.com/en/blogs/cad/parametric-vs-direct-modeling-which-side-are-you-on>

ParametricCamp. (2020, November 11). *Geometry Gems #11: Point in local coordinates*

[Video]. YouTube. <https://www.youtube.com/watch?v=Brf4huewj-A>

parrot & tortoise. (2022, September 27). *Twisted House - Rhino Grasshopper tutorial* [Video].

YouTube. <https://www.youtube.com/watch?v=jk4yL8v-T4Q>

Pask, G. (2011). The Architectural Relevance of Cybernetics. In S. Ahlquist & A. Menges (Eds.), *Computational Design Thinking* (pp. 68–77). Wiley. (Original work published 1969)

Patodi, P. (2021, January 29). *Is AI a job killer or a job creator?* DataFlair. <https://dataflair.training/news/is-ai-a-job-killer-or-a-job-creator/>

PC Sim. (2021, January 24). *Rhino Grasshopper Tutorial: Dividing a surface* [Video]. YouTube. <https://www.youtube.com/watch?v=ti56vr1wq8A>

Pertigkiozoglou, E. (2018a, April 30). 1969 - Eliza PertigkioZoglou - Medium. *Medium*. <https://eliza-pert.medium.com/1969-ab2783c47cbf>

Pertigkiozoglou, E. (2018b, June 14). 1976 - Eliza PertigkioZoglou - Medium. *Medium*. <https://eliza-pert.medium.com/1976-22121bb498c4>

Pertigkiozoglou, E. (2021, February 13). 1960s - Eliza PertigkioZoglou - Medium. *Medium*. <https://eliza-pert.medium.com/1960s-32969cc82d03>

Phamoxtech. (2024, April 13). *What you need to know about IBM Deep Blue*. Phamox Tech. <https://phamoxtech.com/ibm-deep-blue/>

Pingo Doce. (2022, February 7). *Esparguete à bolonhesa | Receitas | Pingo Doce*. <https://www.pingodoce.pt/receitas/esparguete-a-bolonhesa/>

Pol Foreman. (2018, September 29). *Parametric Generation of City block typologies | Waterfront Tanjong pagar* [Video]. YouTube. <https://www.youtube.com/watch?v=JwxSWJZSpu4>

Rahim, M. (2023, November 15). *The evolution of 3D rendering animation*. ArchXStudio. Retrieved April 13, 2024, from <https://archxstudio.com/the-evolution-of-3d-rendering-animation/>

Random Select from list. (2018, September 18). McNeel Forum. <https://discourse.mcneel.com/t/random-select-from-list/71230/4>

RealSpace. (2023, May 18). *The importance of architectural rendering in real estate Marketing | RealSpace blog*. Retrieved April 13, 2024, from <https://www.realspace3d.com/blog/the-importance-of-architectural-rendering-in-real-estate-marketing/>

reCodando. (2023, March 14). *Qual é a diferença entre: Codificação, Programação e Script?* <https://www.linkedin.com/pulse/qual-%C3%A9-diferen%C3%A7a-entre-codifica%C3%A7%C3%A3o-programa%C3%A7%C3%A3o-e-script-tatubit/?originalSubdomain=pt>

- Reese, B. (2014, April 8). 3D Simulation and CAD, It All Began Somewhere. *Dimensional Control Systems, Inc.* <https://blog.3dcs.com/Blog/bid/74078/3D-Simulation-and-CAD-It-All-Began-Somewhere>
- Removing duplicate surfaces in a brep.* (2021, December 23). McNeel Forum. <https://discourse.mcneel.com/t/removing-duplicate-surfaces-in-a-brep/135401>
- Renderização arquitetônica | Renderização de construção em 3D | Autodesk.* (n.d.). AUTODESK. Retrieved March 23, 2024, from <https://www.autodesk.pt/industry/architecture/architectural-rendering>
- ResearchGate. (n.d.). *Figure 4.* https://www.researchgate.net/figure/Probabilistic-diffusion-model-architecture-from-Diffusion-Models-A-Comprehensive-Survey_fig2_368540509
- Review: [Untitled] on JSTOR. (n.d.). *www.jstor.org.* <https://www.jstor.org/stable/3679640>
- Rhino Grasshopper. (2018a, May 6). *Grasshopper Tutorial (Random)* [Video]. YouTube. https://www.youtube.com/watch?v=nU4oS4_E3y4
- Rhino Grasshopper. (2018b, July 11). *Graph Mapper - Grasshopper tutorial* [Video]. YouTube. <https://www.youtube.com/watch?v=nsBLjHCRFZ8>
- Rhino Grasshopper. (2018c, August 18). *Grasshopper Galapagos tutorial* [Video]. YouTube. <https://www.youtube.com/watch?v=g3CPe1bPhEw>
- Rhino Grasshopper. (2019, December 9). *Parametric facade (Grasshopper tutorial)* [Video]. YouTube. <https://www.youtube.com/watch?v=d6QVMj-pNco>
- RICECS. (2022, December 21). *Computer Science vs Programming | MCS@Rice.* RICE UNIVERSITY Department of Computer Science. <https://csweb.rice.edu/academics/graduate-programs/online-mcs/blog/programming-vs-computer-science>
- Sakshibose. (2016, August 19). CAD Ease Out Technical Drawing And Drafting - sakshibose1 - Medium. *Medium.* <https://medium.com/@sakshibose1/cad-ease-out-technical-drawing-and-drafting-d43830583a4e>
- Salmaan Mohamed. (2023, July 6). *Create Realistic RENDERS from CRUSHED PAPER For FREE | LookX AI* [Video]. YouTube. <https://www.youtube.com/watch?v=66xvLqx4hHc>
- Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN COMPUT. SCI.*, 2(160), 21. <https://doi.org/10.1007/s42979-021-00592-x>

Sawantt, S. (2021, August 14). *8 Structures That Were Impossible to Create Without Parametric Modelling Software*. PA-Parametric Architecture. <https://parametric-architecture.com/8-structures-that-were-impossible-to-create-without-parametric-modelling-software/>

Scale by distance. (n.d.-a). Grasshopper. <https://www.grasshopper3d.com/forum/topics/scale-by-distance>

Scale by distance. (n.d.-b). Grasshopper. <https://www.grasshopper3d.com/forum/topics/scale-by-distance>

Scale by distance. (n.d.-c). Grasshopper. <https://www.grasshopper3d.com/forum/topics/scale-by-distance>

Scale by distance. (n.d.-d). Grasshopper. <https://www.grasshopper3d.com/forum/topics/scale-by-distance>

Scale factor. (2022, September 24). McNeel Forum. <https://discourse.mcneel.com/t/scale-factor/148210>

Schroer, A. (n.d.). *Artificial Intelligence (AI): What Is AI and How Does It Work?* Built In. Retrieved January 19, 2022, from <https://builtin.com/artificial-intelligence>

Schroer, A. (2023, July 27). *Artificial Intelligence (AI): What Is AI and How Does It Work?* Built In. Retrieved December 20, 2023, from <https://builtin.com/artificial-intelligence>

Schumacher, P. (2023, September 1). "I am not at all worried about facing the newly empowered competition enabled by AI." *Dezeen*. Retrieved April 23, 2024, from <https://www.dezeen.com/2023/08/30/patrik-schumacher-zaha-hadid-ai-opinion-aitopia/>

Semrush Team. (2023, October 4). *How does ChatGPT work? (Simple & technical explanations)*. Semrush Blog. Retrieved April 2, 2024, from <https://www.semrush.com/blog/how-does-chatgpt-work/>

Show It Better. (2022, June 30). *I used A.I. to create these RENDERS! 🤖 (in less than a minute)* [Video]. YouTube. <https://www.youtube.com/watch?v=85-MvWlutow>

Show It Better. (2023, April 19). *Midjourney for architects: the ultimate workflow for design and photorealistic renders* [Video]. YouTube. <https://www.youtube.com/watch?v=UVNP-4U8jxY>

Simonite, T. (2020, February 11). *The King of Cloud: Q&A with Marc Benioff*. *MIT Technology Review*. <https://www.technologyreview.com/2011/10/28/189988/the-king-of-cloud-qa-with-marc-benioff/>

- Simply Rhino 3D Tutorials & Events. (2018, February 5). *Grasshopper in Rhino3d v6 - Sporph* [Video]. YouTube. <https://www.youtube.com/watch?v=-TlZgBF6vkE>
- Singarasu, Y. (2023, December 10). *Diffusion model for generative image synthesis*. <https://www.linkedin.com/pulse/diffusion-model-generative-image-synthesis-yogeshwaran-singarasu-jgo4c/>
- Single click trigger (like a button)*. (2022, December 12). McNeel Forum. <https://discourse.mcneel.com/t/single-click-trigger-like-a-button/151853/12>
- Sketch Design Craft. (2023, July 13). *AI for ARCHITECTS and DESIGNERS? - From SIMPLE SKETCHES to realistic RENDERS for FREE* [Video]. YouTube. <https://www.youtube.com/watch?v=8SZgEepHHno>
- Skinner, R. E. (2012). *Building the Second Mind: 1956 and the origins of Artificial Intelligence Computing*. <https://escholarship.org/content/qt88q1j6z3/qt88q1j6z3.pdf>
- Souza, F. (2022, April 30). *Programa, Software, Sistema e Aplicativo. Qual a diferença?* <https://www.linkedin.com/pulse/programa-software-sistema-e-aplicativo-qual-diferen%C3%A7a-f%C3%A1bio-souza/?originalSubdomain=pt>
- Spbim, E. (2022, October 26). *Arquitetura paramétrica no BIM*. SPBIM - ARQUITETURA DIGITAL. <https://spbim.com.br/arquitetura-parametrica/>
- Split curves*. (2021, January 28). Parametric by Design. <https://parametricbydesign.com/grasshopper/how-tos/split-curves/>
- Stablexl. (2023, September 5). *Stable Diffusion XL - SDXL 1.0 Model - Stable Diffusion XL*. Stable Diffusion XL. <https://stablediffusionxl.com/#download>
- Tait, T. (2023a, October 25). *How to generate a surface from points in Grasshopper*. Hopific. <https://hopific.com/surface-from-points-in-grasshopper/>
- Tait, T. (2023b, October 25). *How to join geometry in Grasshopper*. Hopific. <https://hopific.com/how-to-join-geometry-in-grasshopper/>
- Tait, T. (2023c, October 31). *How to extrude perpendicular to a curve in Grasshopper*. Hopific. <https://hopific.com/how-to-extrude-perpendicular-to-a-curve-in-grasshopper/>
- Team Kaarwan. (2024, March). *The Impact of 3D Modeling and Rendering in Architectural Design*. Kaarwan. Retrieved April 13, 2024, from <https://www.kaarwan.com/blog/architecture/the-impact-of-3D-modeling-and-rendering-in-architectural-design?id=211>

- Terzidis, K. (2011). Algorithmic Form. In S. Ahlquist & A. Menges (Eds.), *Computational Design Thinking* (pp. 94-101). Wiley. (Original work published 2003)
- The Adam. (2022a, February 12). *Parametric stadium modeling* [Video]. YouTube. https://www.youtube.com/watch?v=PCDdu_Gfe5E
- The Adam. (2022b, April 12). *Parametric Office Architecture - Grasshopper* [Video]. YouTube. <https://www.youtube.com/watch?v=SWuMGFwob6Y>
- The Adam. (2022c, July 5). *Kinetic Facade Tutorial - Rhino Grasshopper* [Video]. YouTube. <https://www.youtube.com/watch?v=sqotWZBRYeQ>
- The Adam. (2022d, August 8). *Grasshopper Parametric Architecture* [Video]. YouTube. <https://www.youtube.com/watch?v=yZcUMHXdXAA>
- The Editors of Encyclopaedia Britannica. (2023a, December 7). *2023 Year in Review | Timeline of historical events*. Encyclopedia Britannica. <https://www.britannica.com/topic/2023-The-Year-in-Review>
- The Editors of Encyclopaedia Britannica. (2023b, December 7). *2023 Year in Review | Timeline of historical events*. Encyclopedia Britannica. <https://www.britannica.com/topic/2023-The-Year-in-Review>
- The Reconstruction of Medieval German Cities with Large-Scale 3D Printed Models*. (n.d.). Formlabs. <https://formlabs.com/blog/reconstruction-of-medieval-german-cities-with-sla/>
- ThinkParametric. (2021a, June 8). *Galapagos 101 - Fundamentals - Intro* [Video]. YouTube. https://www.youtube.com/watch?v=GCci0_pHbkA
- ThinkParametric. (2021b, June 8). *Galapagos 101 - Fundamentals - Part 1* [Video]. YouTube. <https://www.youtube.com/watch?v=SwzttZGBt3l>
- ThinkParametric. (2021c, June 8). *Galapagos 101 - Fundamentals - Part 2* [Video]. YouTube. <https://www.youtube.com/watch?v=uNK5e1aPPyA>
- ThinkParametric. (2021d, June 8). *Galapagos 101 - Fundamentals - Part 3* [Video]. YouTube. <https://www.youtube.com/watch?v=be4OmN3nCul>
- TIME.GRAPHICS. (n.d.). *Patrick Hanratty develops DAC (1 jan 1961 ano - 1 jan 1962 ano) (Linha do tempo)*. TIMEGRAPHICS. <https://time.graphics/pt/period/188597>
- TOP 25 MONEY AND POWER QUOTES (of 88) | A-Z Quotes*. (n.d.). A-Z Quotes. <https://www.azquotes.com/quotes/topics/money-and-power.html>

Topology - Rectangle inside a curve [Online forum post]. (2016, November 17). Grasshopper. <https://www.grasshopper3d.com/forum/topics/topology-rectangle-inside-a-curve?page=2&commentId=2985220%3AComment%3A1642264&x=1#2985220Comment1642264>

Toy, M. (1998). The New Canvas. In M. Toy & N. Spiller (Eds.), *Architectural Design: Architects in Cyberspace II* (Vols. 68, 11-12, pp. 6-7). John Wiley & Sons.

Tyldum, M. (Director). (2014). *The Imitation Game*. Black Bear Pictures.

Vadym Parnytskyi. (2020, April 19). *04 Grasshopper and Galapagos. Building variant with plot area coefficient* [Video]. YouTube. <https://www.youtube.com/watch?v=FP5s9vzg3ys>

Van Gogh Free AI Generator | *getimg.ai*. (n.d.). <https://getimg.ai/models/van-gogh-diffusion>

Wadhwa, R. [BlessedArch]. (2023a, June 13). *Architects using AI to design :Tim Fu from Zaha Hadid Architects* [Video]. YouTube. Retrieved April 23, 2024, from <https://www.youtube.com/watch?v=kO0RzFcT080>

Wadhwa, R. [BlessedArch]. (2023b, June 25). *AI will not replace architects - Patrik Schumacher from Zaha Hadid Architects* [Video]. YouTube. https://www.youtube.com/watch?v=9Fft67Mq_0c

Weng, L. (2021, July 11). What are Diffusion Models? *Lil'Log*. <https://lilianweng.github.io/posts/2021-07-11-diffusion-models/>

Wigley, M. (2010). Numerical Ecosystems. In K. Lydia & H. Castle (Eds.), *EcoRedux: EcoRedux: Remedies for an Ailing Planet: Vol. 80 No 6* (pp. 50-57). John Wiley & Sons.

Wikipedia contributors. (2024a, March 30). *Sketchpad*. Wikipedia. <https://en.wikipedia.org/wiki/Sketchpad>

Wikipedia contributors. (2024b, June 1). *Second-order cybernetics*. Wikipedia. https://en.wikipedia.org/wiki/Second-order_cybernetics

Wikipedia contributors. (2024c, June 7). *Midjourney*. Wikipedia. Retrieved April 13, 2024, from <https://en.wikipedia.org/wiki/Midjourney>

Wikipedia contributors. (2024d, June 17). *Marc Benioff*. Wikipedia. https://en.wikipedia.org/wiki/Marc_Benioff

Winchester, H. (2022, September 1). Zaha Hadid Architects on the tech behind its iconic designs. *chaos*. <https://www.chaos.com/pt/blog/zaha-hadid-architects-on-the-tech-behind-its-iconic-designs>

Witt, T. (2022, June 20). *From Cyberspace to the Metaverse: the Literary Roots of the Virtual World*. Acceleration Economy. <https://accelerationeconomy.com/metaverse/from-cyberspace-to-the-metaverse-the-literary-roots-of-the-virtual-world/>

Wolfram Research, Inc. (2024a, August 22). *Parametric Equations -- from Wolfram MathWorld*. <https://mathworld.wolfram.com/ParametricEquations.html>

Wolfram Research, Inc. (2024b, August 22). *Voronoi Diagram -- from Wolfram MathWorld*. <https://mathworld.wolfram.com/VoronoiDiagram.html>

Woodbury, R. F. (2010). *Elements of parametric design*. Routledge.

Yan, E. (2022, November 27). *Text-to-Image: diffusion, text conditioning, guidance, latent space*. eugeneyan.com. Retrieved April 13, 2024, from <https://eugeneyan.com/writing/text-to-image/#diffusion-from-data-to-noise-and-back>

Yao, C., & Yao, C. (2023, July 15). "Only by embracing AI can you be involved in controlling it" says LookX founder Wanyu He. *Dezeen*. <https://www.dezeen.com/2023/07/14/lookx-wanyu-he-ai-interview-aitopia/>

Yelmen, B., & Jay, F. (2023). An overview of deep generative models in functional and evolutionary genomics. *Annual Review of Biomedical Data Science*, 6(1), 173-189. <https://doi.org/10.1146/annurev-biodatasci-020722-115651>

Zanzotto, F. M. (2019). Viewpoint: Human-in-the-loop artificial intelligence. *Journal of Artificial Intelligence Research*, 64, 243-252. <https://doi.org/10.1613/jair.1.11345>

ZND. (2020, June 14). *Grasshopper: How to divide curve by specific or random distance?* [Video]. YouTube. <https://www.youtube.com/watch?v=zKKMRUUYIAM>

