

Parametric modeling in the initial design of an architectural project: a didactic experience

Cristiana Griz¹, Max Andrade², Camila Resende³

^{1, 2, 3} Universidade Federal de Pernambuco, Recife, Brasil

¹cristiana.sgriz@ufpe.br;

²max.andrade@ufpe.br;

³camila.cresende@ufpe.br

Abstract. The last decades computational advances brought transformations into the Architecture design process and to its didactics, especially parametric modeling (PM), leading to a formal freedom and the automation of design requirements analysis since design early stages. This paper reports results of one of the BIM Cell activities – a project that aims to encourage the new technologies application related to BIM in Higher education. It highlights discussions about teaching PM in a didactic experience. The methodological basis is PLB, which PM definitions, BIM and computational programming are introduced so that students develop a feasible study adequate to mixed-use building using Revit and Dynamo (Autodesk). Initial results suggest parametric solutions allowed students to instantly evaluate them, according to the building area, legal constraints, and budget. Therefore, students envision the design process optimization of solving issues through the main decisions codification using computational means and the PM.

Keywords: Parametric Modeling, Algorithmic Design, Design Process, BIM Teaching, Didactic Experience.

1 Introdução

O século XXI está marcado por uma profunda expansão do conhecimento, o que faz moldar nossa realidade e expandir os limites das possibilidades de atuação profissional. A busca de conhecimento na área da Arquitetura, Engenharia, Construção e Operação (AECO) associada à capacidade tecnológica de coletar, associar e conectar informações tem transformado o processo de emergência das soluções projetuais na capacidade de incorporar parâmetros variados, obtidos de diferentes fontes de conhecimento, em intrincados sistemas computacionais, resultando em processos interativos e colaborativos (Sprecher, Ahrens, 2016).

Essa necessidade de colaboração e interdisciplinaridade no campo da AECO fez destacar o paradigma do *Building Information Modeling* (BIM). Entendido como uma tecnologia de modelagem e um conjunto associado de

processos que visam produzir, comunicar e analisar a informação da construção (Sacks et al, 2018, p. 21), o BIM se tornou sinônimo de transformação tecnológica na Indústria da construção civil. Por meio das tecnologias que suportam o BIM, associado a novos processos e a uma mudança na cultura organizacional, é possível construir um protótipo virtual do edifício e organizar as informações de maneira estruturada, de modo a facilitar a coordenação, o compartilhamento de dados e a distribuição da informação em todas as fases do projeto, passando pela construção, até o uso e manutenção do edifício (Ingram, 2020).

Segundo Sacks et al (2018), a modelagem paramétrica (MP) é considerada uma das tecnologias em que se apoia o BIM. Entretanto, sua aplicação no processo projetual tem sido variável e, nem sempre, feita de maneira a contemplar toda a sua potencialidade. Inicialmente, sua aplicação trouxe inovação em relação apenas à representação e à documentação do projeto, mas o processo projetual permanecia inalterado. Entretanto, a popularização e disseminação de softwares de MP, abordando questões de topologia do edifício, possibilitou que a abordagem paramétrica pudesse ser melhor explorada no processo de projeto, na etapa de morfogênese, possibilitando sua incorporação como um instrumento ativo de auxílio à concepção projetual (Oxmam, 2006).

Baseado em vários autores (Shumacher, 2009; Leach, 2014; Voltolini, 2016), este artigo entende a MP como uma metodologia de definição de modelos paramétricos, onde a concepção da forma, apoiada em suas propriedades geométricas, são definidas através de parâmetros fixos (definições invariantes ou restrições) e parâmetros variáveis (capacidade de evoluir). Quando essa modelagem está sendo feita em Plataformas BIM, esses parâmetros agregam, também, informações de natureza não-geométrica, como material, função, e outros tipos de metadados com informações sobre custo, clima, etc.

É justamente a definição e manipulação computadorizada desses parâmetros que causam o impacto disruptivo no processo projetual. Isso porque as soluções são produtos emergentes do sistema colaborativo entre projetistas (arquiteto, engenheiro, designer, etc.), consultor, cliente (e outros stakeholders), apoiado em métodos que estruturam os dados e no uso do computador que os processa. O resultado possibilita que os projetistas tenham mais liberdade para selecionar, sintetizar e avaliar as soluções projetuais (Bernstein, 2018). Nesse sentido, o nível maior ou menor de interação dos projetistas com novos métodos de pensar e fazer o projeto, apoiadas em tecnologias emergentes pode alterar profundamente não apenas o tipo e a granularidade da informação a ser usada no projeto arquitetônico, mas também a qualidade da solução projetual.

Somado a isso, Tedeschi (2014) coloca que a modelagem baseada em algoritmos permite manipular os softwares de “dentro”. Com o uso e difusão desses softwares, os projetistas começaram a entender que eles podem gerenciar a complexidade estruturando rotinas e procedimentos, a partir de seus fundamentos. Este tipo de modelagem está diretamente vinculado às linguagens de programação que expressam instruções da forma executadas pelo computador, ou seja, o algoritmo (Tedeschi, 2014).

A partir dessas transformações que vêm ocorrendo no campo projetual, vê-se a importância na rediscussão do ensino dessas novas práticas nos cursos de AU. A Universidade Federal de Pernambuco (UFPE) já vem oferecendo disciplinas apoiadas no uso de tecnologias digitais voltadas para o projeto há alguns anos. Entretanto, assim como discutido em Nardelli (2019), a princípio, o foco não passava do aprendizado de manipulação de ferramentas em si, não abordando novas práticas projetuais. Mais do que aprender a usar softwares de manipulação de formas e representação dentro de práticas tradicionais de concepção de projeto, o desafio está em trazer o seu uso para apoiar o processo de concepção e desenvolvimento do projeto. Ou seja, atividades que, apoiadas em novos métodos e no uso da computação, possam trazer novas formas de experimentar, colaborar, trocar, inovar e ensinar esses paradigmas de processo digital.

Sabe-se que ainda não há um consenso sobre de que maneira deve ser feita a incorporação do projeto computacional nos currículos de graduação. Entretanto, sejam em disciplinas isoladas, sejam em workshops pontuais, alguns autores (Nardelli, 2019; Voltolini, 2016) sugerem o uso do *Problem Based Learning* (PBL) como metodologia de ensino (Bashir et al, 2013). Segundo os autores supracitados, o PBL apresenta muitas vantagens para a aprendizagem e o ensino, especialmente em profissões orientadas para a prática, como é o caso da AU.

Nesse sentido, tendo como objetivo analisar os impactos da MP e PBL no processo projetual, para fomentar a criação de novos paradigmas pedagógicos no projeto arquitetônico, foi feita uma experiência didática em uma disciplina eletiva de 30 horas do terceiro ano do curso de AU da UFPE. Embora o foco tenha sido a aplicação da MP no processo de concepção, foram revisitados os conceitos básicos do BIM e introduzidas definições sobre design generativo, junto com o treinamento no software de programação visual Dynamo (*plug-in* do Revit da Autodesk). Esses conhecimentos foram trabalhados através do desenvolvimento de um estudo de viabilidade para um empreendimento arquitetônico, cuja solução formal atendessem alguns requisitos - parâmetros urbanísticos, alcançar o máximo potencial construtivo e não ultrapassar o orçamento previsto.

2 Sobre a teoria e o método

2.1 A modelagem paramétrica e algorítmica

A base conceitual que guia o desenvolvimento do experimento didático parte do entendimento sobre a modelagem paramétrica e a modelagem algorítmica. Sendo assim, é preciso esclarecer a diferença entre uma e outra. Como comentado, a MP surge a partir de um modelo que possui dois tipos de atributos: os parâmetros variáveis e os fixos (as restrições). A partir dessa definição, entende-se que o projetista não desenha, mas cria ligações entre estes atributos do projeto (Voltolini, 2016).

Já a modelagem algorítmica (Também conhecida como *Algorithmic Modeling* ou *Algorithmic Design*), embora também trabalhe com parâmetros fixos e variáveis, refere-se ao uso de uma linguagem de código (ou *script*). Neste caso, o projeto é alterado através da manipulação de um código e não da forma geométrica do objeto projetado (Leach, 2014; Voltolini, 2016), o que abre um amplo caminho para exploração formal. Além disso, esse procedimento indica que o projetista não irá mais representar a forma intuitivamente (*form-making*), e sim, procurar e determinar a forma em um momento posterior (*form-finding*) (Polonini, 2014). Desse modo, a modelagem algorítmica é uma técnica de representação que tem um certo nível de abstração, pois é através de instruções e comandos que serão executados pelo computador que a forma surgirá. Em muitos casos, esse código é desenvolvido por uma linguagem computacional em forma de texto (Voltolini, 2016).

No entanto, há a possibilidade de gerar códigos através de um modo de programação com uma interface gráfica – a programação visual, o que facilita o aprendizado dos projetistas devido a sua familiaridade com a linguagem gráfica. Apesar da programação com linguagem textual proporcionar um domínio maior sobre o que se deseja parametrizar, a segunda se mostra mais próxima da realidade do projetista não especializado em linguagem computacional. A programação visual permite concatenar elementos gráficos, tornando o fluxo de dados mais perceptível, e, em conjunto com a modelagem, tem se difundido entre os projetistas porque permite a criação de processos variados e integrados conforme interesse projetual. Neste tipo de programação, em vez de apresentar uma interface para escrever linhas em um compilador, esses programas contêm uma área de trabalho onde se insere os componentes que comporão o código para realizar uma determinada tarefa. O avanço desse tipo de programação se deu, especialmente, com o surgimento do *Generative Components* da Bentley, por volta de 2003, e em seguida, com a criação do plug-in *Grasshopper* para Rhinoceros da Mcneel, em 2007 (Davis,

Burly & Burly, 2012). Uma alternativa mais recente, em 2015, é o Dynamo Studio, que funciona dentro da plataforma BIM da Autodesk, o Revit.

No contexto da modelagem paramétrica e algorítmica é importante ressaltar que o conhecimento da geometria está no cerne do processo de projeto e, para criar modelos tridimensionais e, mais ainda, para que o computador execute o que ainda está no campo das ideias, é imprescindível ter um bom conhecimento das propriedades das formas geométricas (Pololini, 2014) e do pensamento algorítmico (detalhado e sistematizado). Deve-se, também, levar em consideração que a concepção formal depende de um repertório de formas gerado pelo projetista (Carpo, 2011). Por isso, “o sistema de representação gráfica, e conseqüentemente, os recursos geométricos utilizados nele, exercem influência no processo de concepção das formas arquitetônicas” (Voltolini, 2014, p. 38-39).

2.2 O método didático

No curso de AU da UFPE, as primeiras disciplinas que abordavam temas do BIM e MP foram introduzidas desde 2010. No entanto, para superar o ensino destes temas tirando o foco do mero aprendizado da ferramenta de representação, o experimento aqui apresentado foi concebido e organizado tendo como base o método do *Problem Based Learning* (PBL).

No PBL, o processo de aprendizado é iniciado por um problema, ou o cenário de um problema. Neste, os alunos e professores discutem situações da prática profissional, levantando questões que podem ser transformadas em objetivos de aprendizagem. Assim, o professor passa a ser um facilitador do processo de aprendizagem, diminuindo seu papel de tutor e único agente de transferência de conhecimento (Bashir et al, 2013).

Tendo em vista o principal conteúdo a ser abordado pela disciplina – introdução à modelagem paramétrica e algorítmica no processo de projeto, foi concebido um problema prático do campo da AU que inevitavelmente levaria os alunos a recorrerem a conhecimentos de outra natureza para aprimorar a resolução da questão: o desenvolvimento de um estudo de viabilidade para um empreendimento de uso misto (descrito a seguir). Inicialmente, é apresentada a problemática projetual - área do estudo, necessidades projetuais, parâmetros urbanísticos, etc., para, em seguida, serem introduzidos os conteúdos sob os quais a questão projetual será abordada. Com isso, as definições de BIM são revisitadas, porém o foco é direcionado para o entendimento do design generativo (Ficher; Herr, 2001), da parametrização (Sacks et al, 2018; Oxman, 2006) e introdução básica à programação computacional (Pereira; Vaz, 2013). Temas introduzidos de maneira a proporcionar aos estudantes um

conhecimento básico que auxilie na resolução do problema estabelecido pela disciplina.

3 A experiência didática

A disciplina apresentada neste experimento é uma eletiva ofertada pelo curso de AU da UFPE. Tem uma carga horária teórica de 8h e prática de 22h - totalizando 30h. Esta é localizada no 5º período (terceiro ano) da graduação. Neste estágio, a maior parte dos estudantes cursaram disciplinas que introduzem o BIM, e mesmo que não seja pré-requisito nenhum conhecimento de softwares de MP, a maioria já utiliza alguma ferramenta para produção de modelos 3D.

Como comentado, a disciplina inicia com a apresentação do problema a ser resolvido. Pede-se aos estudantes que desenvolvam um estudo de viabilidade de um empreendimento de uso misto, com comércio e serviço nos pavimentos inferiores (base com no máximo três pavimentos), e uso habitacional nos andares superiores (torre). Além do aspecto plástico da proposta, o projeto deverá atender aos parâmetros urbanísticos - afastamentos, coeficiente de aproveitamento e gabarito (já configurados no *template* com o código geral – Fig. 4), ao custo total da construção (sendo definido o valor do m²) e deverá explorar o máximo do potencial construtivo do terreno. Sendo assim, os critérios de avaliação utilizados foram: (1) criatividade; (2) atendimento aos requisitos de projeto/legislação; (3) qualidade da modelagem; e, (4) geração de relatório e análise das propostas volumétricas adequadas e explicação das descartadas.

Esta é uma questão projetual que poderia ser resolvida através de processos tradicionais de projeto. Entretanto, como o objetivo é introduzir a lógica da MP no processo de resolução de problemas arquitetônicos, os primeiros encontros são destinados à apresentação dos conceitos teóricos sobre design generativo, MP e lógica de programação, principalmente dentro de plataformas de programação visual.

Associado a esse conteúdo teórico são propostas pequenas atividades práticas definidas estrategicamente de maneira que a soma dessas atividades envolve tanto o raciocínio teórico, quanto habilidades com a ferramenta em que o problema será resolvido. Assim, o foco é desenvolver: (a) habilidade para construir um modelo paramétrico (uma geometria criada com a programação visual) que atenda aos parâmetros urbanísticos; (b) habilidade de extrair desse modelo informações sobre a edificação proposta (área total, área por pavimento, gabarito, etc.), e por fim, checar se atendem aos requisitos de custo e máximo potencial construtivo.

Essas pequenas atividades práticas são, primeiramente, resolvidas do ponto de vista teórico, e em seguida, são implementadas no Dynamo. Dessa maneira, o ensino da ferramenta se torna coadjuvante no processo de aprendizagem. Isso porque o objetivo principal é encontrar respostas para a resolução da questão, à luz da lógica paramétrica e do ponto de vista teórico, para, depois, ensinar comandos disponíveis que possam implementar e automatizar a estratégia de modelagem definida. Somado a isso, estas atividades objetivam ensinar como os softwares de programação visual operam a lógica de programação, por isso, são apresentados os tipos de dados – *boolean* (condicional, verdadeiro ou falso), *number*, *string* (texto) e *geometry* (Tedeschi,2014) – e propostos exercícios para o seu entendimento e manipulação.

Essas atividades práticas iniciais também são pensadas para ter soluções relativamente simples, que, a princípio, requerem apenas o raciocínio geométrico. Entretanto, ao final, é mostrado que a mesma lógica paramétrica pode ser utilizada na resolução de questões projetuais mais complexas dentro da área da AU. A Figura 1 exhibe a modelagem da superfície de um jarro, onde o foco está no ensinamento da lógica paramétrica para a construção de uma forma dessa natureza. Em seguida, é mostrado como com algumas pequenas alterações, seja de inputs, seja do tipo de curva que conforma a superfície do jarro (de círculos para quadriláteros, por exemplo), podem fazer com que o produto dessa mesma lógica paramétrica resulte na forma inicial de um edifício.

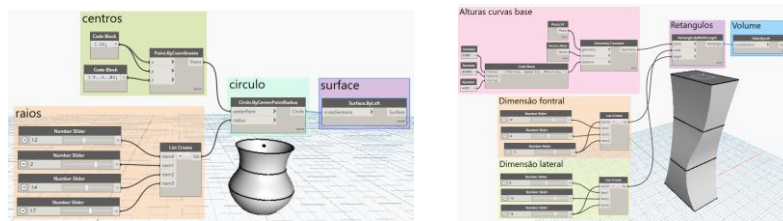


Figura 1. À esquerda, código para modelar parametricamente a superfície de um jarro. À direita, mesma lógica da modelagem paramétrica do jarro, adaptada para gerar o partido inicial de um edifício (alterando *inputs* de altura e formato da curva que forma o volume – de círculo para retângulo). Fonte: autoria própria.

Acredita-se que, ao final desses exercícios, que incluem também o aprendizado e manipulação de dados no Dynamo (listas, sequenciais e operadores matemáticos), e como extraí-los e organizá-los em planilhas, os estudantes estão aptos a resolver, do ponto de vista paramétrico, a questão projetual colocada. Entretanto, como a disciplina é relativamente curta, o material didático fornece um código completo com os *inputs* necessários para o desenvolvimento da geometria do edifício (parâmetros urbanísticos, número

de pavimentos, altura do pavimento etc.), bem como o código necessário para a extração das informações construtivas do modelo – o *template* geral (Fig. 4). Assim, cabe, aos discentes, duas tarefas principais: (1) desenvolver o código que resulta na geometria do edifício; (2) adaptar esse código da geometria para inseri-lo no código geral (substituindo os blocos da cor roxa da Fig. 4), fazendo com que os *inputs* da geometria sejam os parâmetros urbanísticos da área em questão e que, desta geometria sejam extraídas as informações de checagem dos requisitos inicialmente definidos.

Em resumo, seguindo a “teoria da caixa de vidro”, na qual o processo de criação se torna transparente (JONES, 1971), os estudantes percebem que a concepção projetual, apoiada na lógica paramétrica, pode ser aproveitada, adaptada e ajustada de um contexto para outro, podendo gerar múltiplas soluções, e que estas podem ser analisadas facilmente a cada alteração, facilitando a tomada de decisão.

3.1 Desenvolvimento dos trabalhos

Antes dos estudantes começarem a pensar na estratégia de modelagem da geometria que resultaria no estudo de viabilidade, foi apresentado o *template* com código geral (Fig. 2). Neste código, os parâmetros urbanísticos do bairro onde o terreno está situado já estão configurados como *inputs* – sendo esses, um dos parâmetros variáveis (bloco verde à esquerda do código).

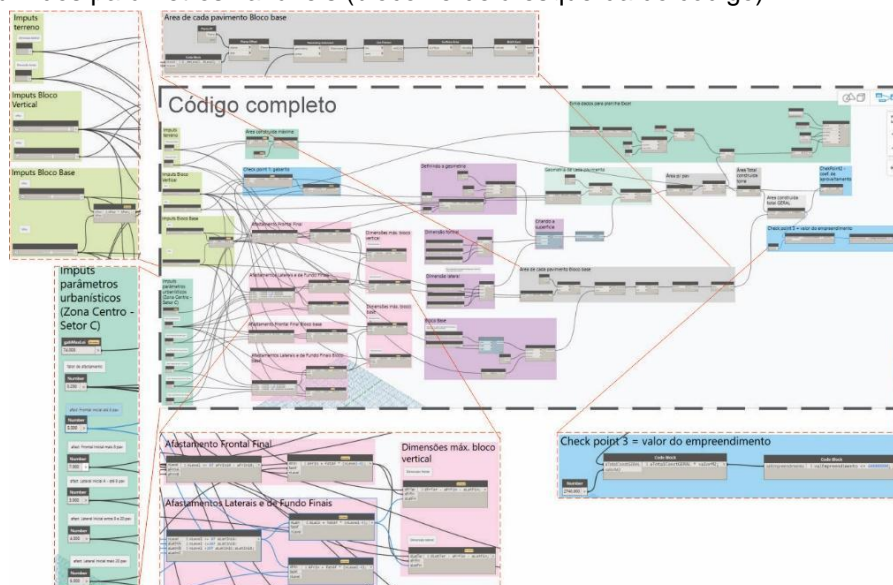


Figura 2. Código para definição do estudo de massa, tendo como inputs os parâmetros urbanísticos e com a criação automática de planilha com informações de áreas do empreendimento. Fonte: autoria própria.

Este código também já inclui diversas informações: as gerais iniciais para a definição da forma geométrica (blocos amarelos à esquerda do código), as dimensões do terreno, o número de pavimentos e a altura do pé direito da base e da torre (todos parâmetros variáveis). Como comentado, o *template* geral já está configurado para: (1) calcular os afastamentos finais da base e da torre - que variam em função do número de pavimentos (blocos rosa do código); (2) extrair as informações construtivas da geometria (blocos cinza do código); e (3) conferir os pontos de checagem (blocos azuis do código). Nestas últimas são verificados se a geometria dos volumes (blocos roxos do código) atende aos requisitos de: (a) gabarito, (b) coeficiente de aproveitamento, e (c) valor total do empreendimento.

Da mesma maneira que nas atividades iniciais, foi pedido aos estudantes que pensassem na forma geral final aproximada do projeto, para depois, definir conceitual e geometricamente a estratégia de modelagem paramétrica dela (Fig. 3). Para tanto, foram feitos rascunhos tanto no lápis e papel, como croquis digitais, em seguida, a lógica da modelagem foi implementada no Dynamo. Nesta fase, o código da geometria foi desenvolvido em um arquivo distinto do *template* geral, para que os alunos pudessem ter maior domínio e liberdade formal, sem se preocupar com as informações sobre parâmetros urbanísticos e dimensões compatíveis com o enunciado da questão de projeto. Em cada um dos trabalhos, após definido o código da geometria, este foi inserido no *template* geral e adaptado para que a geometria fosse gerada a partir dos *inputs* previamente configurados (parâmetros urbanísticos, número de pavimentos e altura do pé-direito). Em seguida, eram adaptadas as ligações para que fossem extraídas as informações construtivas que permitissem a checagem dos requisitos projetuais pré-estabelecidos.

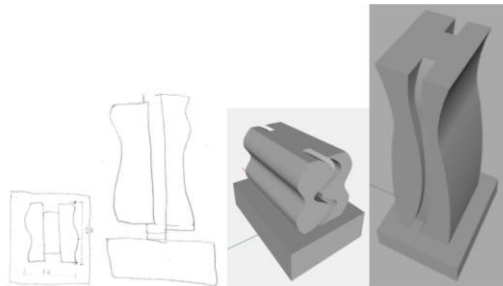


Figura 3. À esquerda, croquis feitos à mão para definição da lógica geométrica. À Direita, estudos iniciais do volume do edifício. Fonte: Trabalhos dos estudantes.

Com o código da geometria adaptado e conectado no *template* geral, os alunos poderiam alterar os parâmetros de número de pavimentos, dimensão de pé-direito e dimensões que definem a geometria e analisar, checando de maneira automática, se a solução formal atendia aos três requisitos pré-

definidos (Fig. 4). A solução formal que melhor atendesse a questões plásticas, à legislação e à viabilidade do empreendimento (maior potencial dentro do orçamento disponível) era escolhida pelos alunos como resposta final à questão projetual. Por fim, os alunos apresentaram o código e as justificativas que embasaram tal decisão.

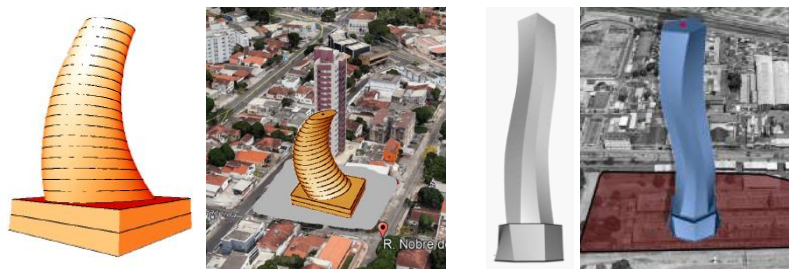


Figura 4. Exemplos de dois trabalhos mostrando duas soluções formais resultantes das variações dos *inputs* do código. Fonte: Trabalhos dos estudantes.

Em síntese, todos os estudantes conseguiram desenvolver tanto um código que correspondesse a sua forma inicial desejada, considerando uma lógica de construção geométrica paramétrica e utilizando uma ferramenta de programação visual. Das diversas explorações formais eram computadas, automaticamente, todas as informações construtivas (área por pavimento e total), o orçamento, bem como eram verificadas as limitações em relação aos parâmetros urbanísticos, sem a necessidade de refazer cálculos em função da forma geométrica. Além disso, foi percebido que pensar o projeto dentro desse novo paradigma proporciona maior liberdade criativa, permitindo, também, o desenvolvimento de formas geométricas mais complexas, ao mesmo tempo em que facilita o gerenciamento (e manipulação) de informações não geométricas.

4 Considerações finais

O experimento propõe uma maneira de se ensinar o tema da MP, sem que o foco fosse voltado apenas para a manipulação de softwares. Isso porque o objetivo era buscar a construção do conhecimento do tema, através da metodologia PBL, agregando aos conceitos do edifício o uso de ferramenta paramétrica e algorítmica que permita a rápida manipulação da forma baseada em dados. Ou seja, por meio da MP apoiada em dados é possível explorar rapidamente opções de geometria e avaliar os impactos na estética da forma, na área construída, no atendimento aos parâmetros urbanísticos e no custo do empreendimento. Assim, com o auxílio da MP e da programação visual, os

estudantes conseguem otimizar o processo de resolução de questões projetuais através da codificação de parâmetros de projeto com o uso de meios computacionais e a aplicação da MP.

O principal objetivo deste artigo foi apresentar sinteticamente uma experiência didática realizada no curso de AU da UFPE, a qual mostrou resultados quanto à integração do PBL e da MP. Foi importante observar que, a partir do entendimento de atividades práticas simples da MP, os estudantes conseguiram desenvolver rapidamente soluções de projeto para problemas de projeto dado. Com base na apresentação dos trabalhos, acredita-se que os estudantes começam a entender os princípios básicos do uso de um código de programação na exploração formal e nos processos criativos, alternando entre o form-making e o form-finding (Polonini, 2014).

Em se tratando da introdução gradual dos processos paramétricos para estudantes em início de curso, sugere-se atividades que exercitem a análise de padrões e sistemas existentes, com posterior aplicação dentro de um contexto projetual, ou a disponibilização de modelos pré-definidos para a investigação de variações, a partir da manipulação de seus parâmetros.

Sabe-se que, apesar da intensa discussão sobre o impacto de novas tecnologias dentro do campo da AU, ainda não são vistas suas aplicações amplamente difundidas nos ateliês de projeto no Brasil. Algumas disciplinas estão sendo inseridas em currículos, como a apresentada aqui, mas ainda não são suficientes para provocar uma mudança significativa na maneira de ensinar a projetar. Segundo Romcy (2017) é preciso ter uma visão curricular estratégica do curso de AU, de maneira que as áreas de projeto e representação estejam em sintonia, para que os estudantes possam adquirir habilidades em atividades que ultrapassem os âmbitos de disciplinas individuais. É preciso ainda uma maior discussão didática-pedagógica acerca do uso dos computadores no ensino de projeto, uma vez que estes têm presença constante dentro do ateliê.

Por fim, é possível observar determinadas estratégias e desafios para a introdução da abordagem paramétrica no ensino de projeto. Para além da capacitação em determinados softwares, é necessária uma mudança de olhar sobre o objeto projetado, a partir da compreensão da forma como expressão de uma lógica pré-definida e a visão do processo como sistêmico, onde a definição de relações e interações repercutem na geração do produto final.

Referências

Bernstein, P. (2018). Architecture, design, data: Practice competency in the era of computation. De Gruyter.

- Bashir, F. M., Ahmad, M. H., & Hamid, M. (2013). Design studio as problem based learning in architectural education in Universiti Teknologi Malaysia. In *Proceedings of the 4th International Research Symposium on Problem-Based Learning (IRSPBL)* (pp. 373-379).
- Carmo, M. (2010). The Alphabet and the algorithm. <http://ci.nii.ac.jp/ncid/BB12516318>
- Davis, D. P., Burry, J., & Burry, M. (2011). Understanding Visual Scripts: Improving Collaboration through Modular Programming. *International Journal of Architectural Computing*, 9(4), 361–375. <https://doi.org/10.1260/1478-0771.9.4.361>
- Fischer, T., & Herr, C. M. (2001). Teaching Generative Design. 4th International Generative Art Conference, Generative Design. <http://papers.cumincad.org/data/works/att/c78f.content.pdf>
- Ingram, J. (2020). *Understanding BIM: the past, present and future*. Routledge Taylor & Francis Group.
- Jones, C. Informe sobre la situación de la metodología del diseño. In: BROADBENT, G. (Org.). *Metodología del diseño arquitectónico*. Gustavo Gili, Barcelona, 1971. p. 385-395.
- Leach, N. (2014). *Parametrics Explained*. <https://doi.org/10.7480/ngb.1.1.1530>
- Nardelli, E. (2019). BIM training in Brasil. Preparing professionals for BIM adoption by public administration [Review of BIM training in Brasil. Preparing professionals for BIM adoption by public administration]. In *Congresso Da Sociedade Ibero-Americana De Gráfica Digital* (Vol. 17). Buenos Aires.
- Oxman, R. (2006). Theory and design in the first digital age. *Design Studies*, 27(3), 229–265. <https://doi.org/10.1016/j.destud.2005.11.002>
- Oxman, R. (2008). Digital architecture as a challenge for design pedagogy: theory, knowledge, models and medium. *Design Studies*, 29(2), 99–120. <https://doi.org/10.1016/j.destud.2007.12.003>
- Pereira, N. S., & Vaz, C. E. V. (2013). Parametrismo e ensino de geometria—as superfícies de Felix Candela. In *Simpósio Nacional de Geometria Descritiva e Desenho Técnico e International Conference on Graphics for Arts and Design* (pp. 01-09).
- POLOLINI, F. *A Modelagem Paramétrica na concepção de formas curvilíneas da Arquitetura Contemporânea*. Dissertação (Mestrado em Arquitetura e Urbanismo da UFBA), 2014, 288p.
- ROMCY, N. *Abordagem paramétrica e ensino de projeto. Proposição de diretrizes metodológicas, considerando estratégias curriculares e o atelier de projeto*. Tese (Programa de pós-graduação em Arquitetura e Urbanismo). Universidade Federal do Rio Grande do Norte: Natal, 2017. 317f.
- SACKS, R.; EASTMAN, C.; LEE, G; TEICHOTZ, P. *BIM Handbook: a guide to Building Information Modeling for Owners, Desiners, Engineers, Contractors, and Facility Managers*. New Jersey: John Wiley & Sons, Inc. Hoboken, 2018.
- Schumacher, P. (2009). Parametricism: A New Global Style for Architecture and Urban Design. *Architectural Design*, 79(4), 14–23. <https://doi.org/10.1002/ad.912>
- Sprecher, A., & Ahrens, C. (2016). Adaptive knowledge in architecture: A few notes on the nature of transdisciplinarity. *Architectural Design*, 86(5), 26-35.
- Tedeschi, A. (2014). *AAD Algorithms-Aided Design. Parametric strategies using Grasshopper*. Le penseur publisher.
- Voltolini, G. (2016). *Design paramétrico e modelagem algorítmica: os efeitos de seus conceitos e técnicas em acadêmicos de arquitetura*. <https://repositorio.ufsc.br/xmlui/handle/123456789/173283>