



SIGRADI2018
TECHNOPOLITICAS
xxii congresso da sociedade
iberoamericana de gráfica digital
22th conference of the
iberoamerican society
of digital graphics
07|08|09|novembro|2018
iau usp | são carlos | sp br

From the automated generation of layouts to fabrication with the use of BIM: a new agenda for Architecture in the 21st century

Verley Henry Côco Júnior

Universidade Estadual de Campinas | Brazil | verleyjr@gmail.com

Gabriela Celani

Universidade Estadual de Campinas | Brazil | celani@g.unicamp.br

Abstract

Scripting, BIM and Digital Fabrication are already recognized as important skills in education and practice in Architecture in the 21st century. However, they are not always applied together to generate innovative results for the industry. This paper starts from the observation of the difficulty that prefabricated bathroom factories have in meeting a demand for mass customization and proposes a workflow that goes from the generation of layouts to modeling in BIM and the automated production of documents for manufacturing. The preliminary results demonstrate the possibility of changing the mass production culture of an industry, by means of applying the proposed workflow.

Keywords: Building Information Modeling; Process algorithm; Automation; Modular bathrooms; Prefabrication.

INTRODUÇÃO

As demandas contemporâneas para a construção civil e o avanço da tecnologia abrem novos campos de atuação para arquitetos. Há uma expectativa socioeconômica por produtos personalizados, feitos com maior qualidade, mais rapidamente e com o menor custo possível – um feito que pode ser observado no desenvolvimento da indústria naval, automobilística e aeronáutica pela integração de expertises em busca de uma racionalização produtiva. As ferramentas digitais oferecem meios para responder às demandas atuais através de estruturas produtivas mais horizontais e dinâmicas por meio de redes colaborativas entre arquitetos, construtores, engenheiros de produção e de materiais (Kieran e Timberlake, 2004).

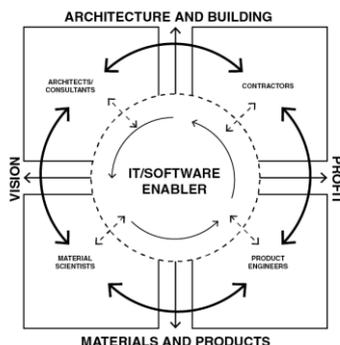


Figura 1: Esquema de possível integração entre profissionais com auxílio de software (Kieran e Timberlake, 2004).

O *Building Information Modeling* (BIM) potencializa essa integração ao centralizar as informações do ciclo de vida do edifício em um único modelo virtual cujo

armazenamento e fluxo de dados acontece em tempo real entre os envolvidos, como retrata a Figura 1 (Kieran e Timberlake, 2004). O BIM se apresenta como uma plataforma paramétrica própria para monitoramento de tempo e de custo, além de permitir a atualização e geração automatizada de documentos gráficos para a produção (Leach, 2012).

A correta identificação das variáveis envolvidas e o claro entendimento do problema permitem que determinados processos ou etapas de um projeto arquitetônico sejam automatizados. Isso porque algumas etapas do projeto arquitetônico são constituídas por uma sequência de passos repetitivos, que podem ser descritos em forma de algoritmos. Desde os anos 1960, autores como Eastman (1968) já afirmavam que o que é feito inconscientemente no projeto – pelo reconhecimento de padrões provenientes de unidades básicas (*Design Units*) – pode ser implementado em um sistema de computador. Neste caso, a estruturação dos dados e a declaração de parâmetros (*inputs*) criam horizontes para exploração de possibilidades dentro de intervalos previamente determinados como aceitáveis. Em seguida, um mesmo ou outro algoritmo pode convergi-las segundo uma sequência de regras específicas, encontrando o melhor arranjo que responda a um determinado problema (Caetano, Leitão e Bastos, 2016).

Discutimos neste artigo se o desenvolvimento de um sistema de automação de *layout* em BIM viabilizaria a produção em massa de produtos customizados. Para isso, buscamos entender quais os diálogos possíveis dos arquitetos com a indústria e como esses profissionais podem construir sistemas algorítmicos e implementá-los em BIM. Este recorte foi estabelecido pelo motivo desse ambiente ser ao mesmo tempo um sistema facilmente isolado, bem como se apresentar como um caminho

crítico no cronograma de construção (Kieran e Timberlake, 2004).

Como prova de conceito, encontra-se em desenvolvimento um modelo algorítmico para a geração de *layouts* de banheiros pré-fabricados. Para sua implementação, foram utilizados programação visual e textual em *Grasshopper* e posteriormente foram exploradas algumas possibilidades de implementação em BIM por meio do *Geometry Gym*, *VisualArq*, *Rhynamo* e *Live Connection*.

NOVAS AGENDAS

O PROCESSO ALGORÍTMICO: AUTOMATIZAÇÃO DO LAYOUT

Ao longo de um processo de projeto, os arquitetos trabalham sequencialmente identificando problemas, coletando dados, analisando-os, sintetizando-os e avaliando os resultados, ou, resumidamente, identificando problemas, gerando soluções por partes e, por último, integrando-as ao todo. Isso ocorre por meio dos elementos básicos de um sistema (*Design Units*) e sua reunião em padrões estabelecidos heurísticamente pela identificação dos mesmos elementos em estudos de caso que dão uma resposta similar – é uma memória semântica que após ser analisada é sintetizada em uma solução para um problema específico. Após a etapa de gerar possibilidades, os arquitetos avaliam se esta solução se integra harmonicamente com o todo (Eastman, 1968) – um ciclo de sucessivas divergências e convergências. Assim, os elementos básicos de um sistema podem ser combinados holisticamente para gerar novas possibilidades em um sistema generativo (Mitchell, 1975).

Em um artigo publicado nos anos 1960, e portanto muito antes do desenvolvimento, por ele próprio, do conceito de BIM, e da popularização dos softwares de modelagem paramétrica e da programação visual, Eastman (1968) testou a hipótese de estruturação da informação relacionada à resolução de problemas de projeto – identificação, geração e integração – aplicando protocolos para a resolução de *layout* de um banheiro. No experimento, que utilizou o método de análise de protocolo (*protocol analysis*¹), projetistas seguiram a sequência projetual que faziam normalmente, porém verbalizando o motivo das decisões. Como resultado, obteve-se trinta e três restrições, como por exemplo “a área principal de um cômodo possui uma área mínima livre [C1], (...) a banheira não deve ser posicionada de frente para a janela [C7]” (Eastman, 1968, p. 48, tradução nossa). Do mesmo modo, elencou-se vinte e quatro *Design Units (DU)*, dentre eles banheira (DU1), pia (DU4), janela (DU10) e luminárias (DU20). Por fim, determinou-se onze manipulações como “locar a unidade no canto [M1], (...) mover a unidade para o canto da mesma parede [M3], (...) alinhar [M10]” (Eastman, 1968, p.49, tradução nossa).

O Trabalho de Eastman (1968) infere que existe uma rotina implícita de passos para a resolução de problemas (Figura 2). Segundo Leach (2012), sequências de ações

¹ Para uma revisão da literatura sobre *protocol analysis* ver, por exemplo, Hao e Ching-Chiuan (2009).

quando são lidas como instruções formam um algoritmo. A consequência direta disso é a possibilidade de automatização de determinados processos por meio de ferramentas de programação. Eastman (1968) conclui em relação a um dos protocolos que “apenas uma máquina seria paciente o suficiente para realizar simulações recursivas”, analisando todas as combinações possíveis a partir das regras estabelecidas (Eastman, 1968, p. 78, tradução nossa). Assim, o que é feito intuitivamente pode dar lugar a uma clara e racional sistematização, elevando a eficiência do processo projetual e reduzindo o tempo gasto com trabalhos repetitivos.



Figura 2: "Simulação número um. Sequência usando busca não direta. (...)" (Eastman, 1968, p.70).

Dentro deste contexto, os arquitetos podem olhar para o projeto, ou ao menos para algumas partes dele, como um sistema lógico divisível em vários subsistemas. Cada uma das partes possui elementos, restrições e manipulações a serem identificadas, listadas e combinadas para ao final comporem o todo novamente. Uma visão lógica e holística do projeto - desde a declaração de variáveis ao entendimento das suas interdependências - possibilita que os arquitetos avaliem quais processos podem ser otimizados e quais os mecanismos necessários para isto ocorrer.

BIM: INFORMAÇÕES EM TEMPO REAL

Stanley Davis (1990) apresenta cenários socioeconômicos emergentes nas últimas décadas do século XX que viriam a reestruturar empresas, negócios, sistemas de produção e prestação de serviços. O conceito de tempo seria resignificado no encurtamento ou na inexistência de intervalos para o fluxo de informações, como diria Davis (1990), “se A e B estão ligados, então as informações específicas para A deveriam alterar B quase ao mesmo tempo” (p. 33). Na construção civil isso é observado no desenvolvimento de BIM, onde um único modelo virtual tridimensional é constantemente atualizado e concentra todas as informações ao longo do ciclo da vida do edifício em um banco de dados com interdependência das informações. A resposta às modificações acontece em tempo real, isto é, as informações armazenadas e interconectadas resultam na visualização das alterações, de quantitativos de materiais,

estimativas de custos, tabelas, sequência de construção e desenhos para fabricação (Garber, 2014; Eastman et al., 2013).

O modelo virtual com semântica construtiva proporciona que arquitetos e engenheiros, ainda nas etapas iniciais de projeto, analisem diferentes configurações, simulem alternativas, identifiquem conflitos e transmitam intenções projetuais em três dimensões (Garber, 2014). Davis (1990) afirma que “na nova economia, a visão da máquina permite a inspeção de todas as peças em tempo real, enquanto estão sendo produzidas [...]” (p. 29). Por meio do BIM, o controle da qualidade de produção está presente ainda na fase de projeto e cada parte pode ser verificada antes da construção. O BIM abre novos caminhos para o projeto por ser uma ferramenta de gestão e controle de dados. Davis (1990) destaca a relevância da informação: “geralmente os recursos são finitos, mas a informação é infinita e é o maior valor agregado na nova economia de hoje” (p. 107).

Nessa plataforma, os arquitetos se mantêm como autores do projeto, porém em uma relação muito mais próxima de outros profissionais. Como descreve Garber (2014), os arquitetos são aqueles que criam, selecionam as equipes de trabalho e acompanham todo o andamento do projeto. Os profissionais necessários ao longo do projeto acrescentam informações em tempo real ao modelo, o que agiliza o trabalho integrado e a compatibilização de sistemas. Assim, mediante a potencialização do fluxo de informações, a indústria se torna um campo propício para que estas trocas aconteçam e resultem em produtos com maior qualidade para a construção (Kieran e Timberlake, 2004).

OS ARQUITETOS E A INDÚSTRIA: INTEGRAÇÃO EM FAVOR DA PRÉ-FABRICAÇÃO

Até o Renascimento, os arquitetos atuavam como *master builders*, reunindo as funções de planejadores do espaço, construtores, engenheiros de produção e engenheiros de materiais. Este foi o caso de Brunelleschi que, para a construção da cúpula da Catedral de Santa Maria em Florença, inventou um novo sistema de elevadores e uma nova forma de assentar tijolos. Entretanto, o aumento da complexidade construtiva ao longo dos séculos seguintes demandou uma especialização de conhecimento e uma divisão de responsabilidades com outros profissionais para que se pudesse lidar com o projeto (Kieran e Timberlake, 2004).

Para Kieran e Timberlake (2004), nas últimas décadas do século XX, os arquitetos passaram a se voltar para um produto formal onde a aparência se tornou mais importante que a substância – uma separação entre *art* e *craft* quando, na verdade, a arquitetura deveria tratar das duas vocações. A indivisibilidade de *art* e *craft*, por sua vez, deve passar pela industrialização. Os autores destacam a visão de Le Corbusier:

Para Le Corbusier, engenheiros, diferentemente de arquitetos, não são guiados por preconceção de aparência. Em vez disso, eles possuem um único foco no propósito e na economia. Massa poética, superfície e plano resultam dos filtros da economia e do processo por meio do qual seus projetos devem passar (Kieran e Timberlake, 2004, p.7, tradução nossa).

Kieran e Timberlake (2004) defendem que a arquitetura deve rever seus processos construtivos observando a metodologia produtiva da indústria aeronáutica, naval e automobilística, cuja produção é feita em módulos, partes e conjuntos, por meio de linhas de montagem independentes. Kieran e Timberlake (2004), ao falar sobre a indústria aeronáutica, escrevem que “o propósito [dessa abordagem] é alcançar elevada qualidade, melhores recursos, menos tempo de fabricação e menor custo: mais *art* e *craft*, não menos” (p. 81, tradução nossa).

A industrialização vista nos séculos passados permitiu a mudança da produção artesanal para uma produção em larga escala com redução de custo e, mais do que isso, conduziu a uma personalização em massa. Ao invés de se produzir mais do mesmo, há a diversificação de produtos sem o aumento do custo e do tempo de produção.

Uma personalização em massa efetiva só ocorre mediante uma maior atenção aos sistemas produtivos e ao controle da informação nas etapas de projeto, implicando na automação de processos, com fabricação por meio de máquinas de controle numérico, impressoras 3D ou braços robóticos. Esta nova visão sobre arquitetura é um campo fértil para novas sementes criativas cujas raízes estarão em sistemas racionalizados que respondem a questões que estão além da forma, mas falam de processos. A pré-fabricação permite que o edifício seja pensado em conjuntos e partes fabricadas fora do canteiro de obras e depois encaixadas no local, como ocorre em uma montadora de veículos. O resultado desta abordagem arquitetônica é o aumento exponencial da qualidade e das possibilidades projetuais com redução de custo e de tempo (Kieran e Timberlake, 2004).

O *master builder* no presente não é uma pessoa como o profissional renascentista; assemelha-se mais a uma inteligência coletiva na qual os arquitetos estão inclusos ao lado de outros arquitetos, construtores, engenheiros de materiais e de produção (Kieran e Timberlake, 2004).

Ao longo do desenvolvimento desta pesquisa, foram realizadas visitas a indústrias ligadas à construção modular, o que facilitou o entendimento de como acontece o diálogo entre as áreas, de seus desafios e de suas potencialidades. A primeira visita foi a uma indústria de banheiros pré-fabricados no estado de São Paulo, onde foi possível ver a linha de montagem e posteriormente validar resultados preliminares, por meio de conversa com o gerente da indústria; a segunda visita foi a uma indústria do mesmo segmento na região de Trentino-Alto Ádige, Itália, na qual tivemos contato com diferentes processos construtivos. Uma terceira visita foi realizada a uma empresa de inovação tecnológica de concreto e sua parceira de fabricação - ambas na Lombardia, Itália - nas quais foi possível observar uma interação consistente entre arquitetura, engenharia de materiais, engenharia de produção e clientes.

PROPOSTA DE WORKFLOW

Com os objetivos de testar os conceitos apresentados acima e de tentar viabilizar a personalização em massa nas indústrias de banheiros pré-fabricados, foi desenvolvido um *workflow* que começa com a geração

automatizada de layouts, passa pela modelagem em BIM e termina com a produção de documentos para a fabricação dos módulos.

Neste artigo são relatadas apenas as duas primeiras etapas do *workflow*, uma vez que o trabalho ainda se encontra em desenvolvimento. É importante salientar que a etapa I, de automatização de *layouts*, não teve a intenção de reproduzir, e muito menos de aprimorar, o brilhante trabalho de Eastman desenvolvido nos anos 1960. Esse trabalho serviu apenas de inspiração para a incorporação de uma etapa a mais de automação no processo, a qual é descrita aqui de maneira simplificada por motivos de restrição de espaço.

ETAPA IA: COMBINAÇÃO HOLÍSTICA

O processo de desenvolvimento do algoritmo de geração do *layout* dos banheiros teve início com a identificação dos *Design Units* que poderiam compor um banheiro pré-fabricado (Figura 3). São eles: cuba, sanitário, bidê, chuveiro, banheira, shaft e porta. Escolhemos combinar quatro elementos básicos na tipologia mais simples possível. Assim, separamos porta, sanitário, cuba e chuveiro, combinando-os holisticamente em uma malha quadrada 2x2.

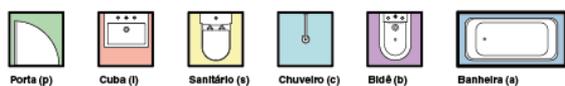


Figura 3: *Design Units* identificados. Fonte: Autores.

Estabelecemos uma lógica para a nomenclatura que viria a facilitar a organização do trabalho. Primeiramente, cada uma das unidades da malha recebeu uma numeração de um a quatro, contando em sentido horário e iniciando a contagem na casa inferior esquerda. Em segundo lugar, cada *Design Unit* recebeu uma letra: porta (p), cuba (l), sanitário (s) e chuveiro (c). Assim, a cada letra seria associado um número independentemente do caso. Por exemplo: o código p1 – l2 – s3 – c4 se refere a uma porta na casa 1, uma cuba na 2, um sanitário na 3 e um chuveiro na 4. Na primeira geração, combinamos quatro equipamentos distintos que não podem se repetir em uma malha quadrada 2x2. Foram geradas vinte e quatro possibilidades no total (sem levar em conta a rotação dos equipamentos). A maioria dos casos, porém, poderia ser obtida por rotação e espelhamento de outros; esses casos foram descartados, restando apenas seis casos diferentes.

Cada resultado da primeira geração foi trabalhado separadamente na segunda, introduzindo neste momento a rotação dos equipamentos - cada um poderia estar em duas posições diferentes para que continuasse ligado a uma parede. Logo, tivemos dezesseis possibilidades para cada uma das seis combinações da primeira geração. Nesta fase também foram introduzidas regras a fim de convergir as soluções. São elas: uma porta nunca deve abrir diretamente para uma bacia sanitária e todos os equipamentos devem estar conectados a no máximo duas paredes adjacentes. O código de cada caso recebeu um segundo número indicando a orientação do equipamento; se igual a 1 estaria alinhado com o eixo y e se igual a 2 alinhado com o eixo x. Por exemplo: o código

p1.1 – l2.1 – s3.1 – c4.1 indica que todos os elementos estão alinhados com o eixo y.

Após as dezesseis combinações para os seis casos, obtivemos noventa e seis tipologias diferentes, porém ao aplicarmos as duas regras este número caiu para dezesseis novamente. Na terceira e última geração (Figura 4), foi aplicada uma terceira regra segundo a qual a cuba jamais estaria em diagonal com a porta, para melhor circulação interna. Conseqüentemente, o número aceitável de soluções caiu para doze. Destes, novamente, seis eram tipologias rotacionadas e espelhadas, assim obtivemos ao final seis soluções diferentes para quatro equipamentos combinados em uma malha de 2x2 aplicando três restrições e uma manipulação (rotação) por equipamento.

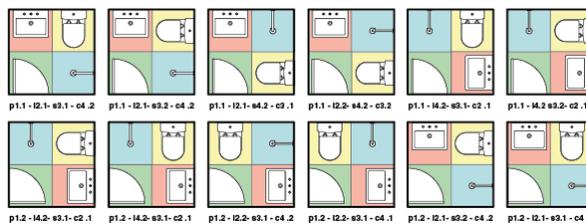


Figura 4: Resultados da análise combinatória. Fonte: Autores.

ETAPA IB: ALGORITMO EM CAD

Na etapa seguinte, o número de variáveis envolvidas foi ampliado, deixando de ser um caso holístico para ser um caso estruturado heurísticamente. Nesta etapa, foram estabelecidas as tipologias de banheiro quadrado e retangular. Para o primeiro caso escolhemos a tipologia p1.1 – l2.1 – s4.2 – c3.1. Já o segundo permitiria uma maior flexibilidade de equipamentos e de disposições, como possibilidade de inserção de uma segunda pia ou de um bidê, ou de dispor a porta em diferentes posições. A tipologia retangular foi separada em duas: uma com porta no canto e outra com porta central. Assim, estabelecemos três tipologias: quadrada, retangular com porta no canto e retangular com porta central.

O algoritmo foi desenvolvido incrementalmente por meio de programação visual e textual em *Grasshopper*, plataforma que permite a interoperabilidade entre CAD e BIM, potencializando a criação e aplicação de algoritmos. Em um primeiro momento, parametrizamos esquematicamente a planta do banheiro quadrado feito na Etapa IA. A partir da valoração das variáveis de largura do banheiro, do box e da porta, as posições da cuba e do sanitário eram reajustadas para se posicionarem equidistantes do box e das paredes.

Desde o início, houve a preocupação em tornar o algoritmo acessível para um usuário que não está familiarizado com programação. Assim foi utilizado um add-on para *Grasshopper*, o *Human UI*, que permite o desenvolvimento de interfaces. Seus componentes recebem as variáveis e em certo momento bifurcam os *inputs* de modo que elas são inseridas em um menu de interface e posteriormente devolvidas ao sistema. O uso do *Human UI* trouxe bons resultados em um curto espaço de tempo.

Nesse momento, foi iniciada a parametrização do layout a partir de relações obtidas na literatura. Foram consultadas referências como Ching (2010), Gurgel (2002), Neufert (2009) e Panero e Zelnik (2002), além do catálogo da Deca para obter as dimensões médias para cada equipamento, bem como seus afastamentos com relação às paredes e demais equipamentos.

Houve a introdução das variáveis para o *layout* em um componente de VB.net. Ainda neste momento, o componente retornava apenas valores de largura e comprimento. A estrutura de interface do *Human UI* foi retirada provisoriamente, com o propósito de ganharmos maior agilidade para experimentação e desenvolvimento inicial.

O código em VB.net foi estruturado em seguida por meio do comando *select case* (Figura 5). Isso trouxe uma otimização do código. Todo o desenvolvimento seguinte foi embasado nesta função.

```

327 ' Posicao Sanitario e bidet = A
328 If ps = 0 And bi = 1 Then:
329   Faucet_position_y = 2 * (y2 - lb) / 3 + 0.3
330   Faucet2_position_y = 1 * (y2 - lb) / 3 + 0.4
331   Toilet_position_x = 2 * (x2) / 3 + 0.2
332   Toilet_position_y = 0
333   Bidet_position_x = 1 * (x2) / 3 + 0.2
334   Bidet_position_y = 0
335   Toilet_bidet_orientation_x = 0
336   Toilet_bidet_orientation_y = 0.62
337   ' Posicao Sanitario e bidet = B
338 ElseIf ps = 1 And bi = 1:
339   Faucet_position_y = 2 * (y2 - lb) / 3
340   Faucet2_position_y = 1 * (y2 - lb) / 3
341   Toilet_position_x = x2
342   Toilet_position_y = (y2 - lb - tp - 0.4) / 3 + 0.05
343   Bidet_position_x = x2
344   Bidet_position_y = 2 * (y2 - lb - tp - 0.30) / 3 + 0.25
345   Toilet_bidet_orientation_x = -0.62
346   Toilet_bidet_orientation_y = 0
347   ' Posicao Sanitario B sem bidet
348 ElseIf ps = 1 And bi = 0:
349   Faucet_position_y = 2 * (y2 - lb) / 3
350   Faucet2_position_y = 1 * (y2 - lb) / 3
351   Toilet_position_x = x2
352   Toilet_position_y = (y2 - lb - tp - 0.4) / 3 + 0.3
353   Toilet_bidet_orientation_x = -0.62
354   Toilet_bidet_orientation_y = 0
355   ' Posicao Sanitario A sem bidet
356 Else ps = 0 And bi = 0:
357   Faucet_position_y = 2 * (y2 - lb) / 3
358   Faucet2_position_y = 1 * (y2 - lb) / 3
359   Toilet_position_x = 2 * (x2) / 3 + 0.05
360   Toilet_position_y = 0
361   Toilet_bidet_orientation_x = 0
362   Toilet_bidet_orientation_y = 0.62
363 End If
364 End If
365 End If

```

Figura 5: Trecho final do algoritmo desenvolvido em VB.net para geração do layout para os banheiros estudados. Fonte: Autores.

Mediante a consolidação das relações entre as variáveis de *layout*, foi possível dar continuidade a outros trechos do código. Foi criada uma interface com o *Human UI* em duas janelas: a primeira compunha o menu principal na qual estavam contidas as variáveis relativas a todas as tipologias, enquanto que na segunda estavam as variáveis específicas de cada uma, isto é, conforme o usuário altera as tipologias, as janelas da interface são abertas ou fechadas. Além disso, nesse momento foi introduzida a criação do piso, paredes e cobertura simplificados, bem como as primeiras análises numéricas, como de área e perímetro.

As versões seguintes buscaram resolver a estrutura feita em *light steel frame* com vedação em *drywall*. Foi necessário realizar diversas análises e mecanismos para adequar sua aplicação a diferentes situações como, por exemplo, ajustes de largura e profundidade em relação às modificações nos painéis de *drywall* ou a retirada de perfis para dar lugar à porta. O algoritmo da estrutura retira os perfis de onde a porta foi inserida, bem como

inverte um deles para a colocação do batente, esteja a porta em uma das paredes alinhadas com o eixo x ou y.

Na sequência, o trecho do código para a inserção dos perfis em planta foi atualizado, mediante a consulta dos catálogos da Gypson, Knauf Drywall, Ananda e Placo. Foi acrescentada a estrutura nas paredes, pisos e cobertura, o que também demandou uma reestruturação dos menus.

A interface (Figura 6) continuou a lógica dos códigos anteriores - um menu principal para os parâmetros comuns e três menus secundários para cada uma das tipologias. No menu principal, o usuário pode ativar as análises, que aparecerão em outra janela, acessar as configurações para o “3D simplificado” e para o “Drywall”, além de acessar o menu “Finalização”, onde as opções *Layout*, *Drywall 2D*, *Drywall 3D* (estrutura), *Drywall 3D* (placas) e volume simplificado podem ser convertidas em desenhos vetoriais dentro do *Rhinoceros*, inclusive com uma divisão automática por *layers*. É importante ressaltar que as ativações ou desativações presentes no menu vão além de uma visualização, proporcionando o completo desligamento do trecho de código, ou seja, quando o usuário desativa uma certa funcionalidade está causando um “erro proposital” em um determinado conjunto de componentes, conduzindo ao desligamento daquela seção e deixando o processamento mais rápido.



Figura 6: Conjunto de menus de interface. Fonte: Autores.

A Figura 7 apresenta exemplos de layouts gerados automaticamente pelo algoritmo a partir de três tipologias de banheiros.

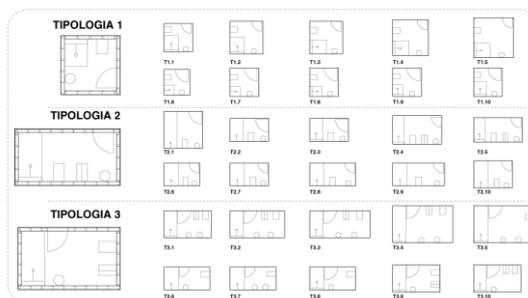


Figura 7: Trinta resultados possíveis de layout gerados pelo código agrupados segundo as três tipologias iniciais. Fonte: Autores.

ETAPA II: ALGORITMO EM BIM

Iniciamos a implementação do código em BIM explorando diferentes caminhos. Inicialmente, trabalhamos em um modelo simplificado e autônomo em relação ao que foi realizado anteriormente com a finalidade de entender as características de cada situação. O modelo consistia em um ambiente retangular, com paredes, laje para o piso e

outra para a cobertura, estrutura metálica externa, uma porta e uma janela (Figura 8). O projeto do módulo utilizou as soluções construtivas observadas na indústria de banheiros pré-fabricados, de modo que será possível, futuramente, que o algoritmo venha a ser efetivamente empregado.

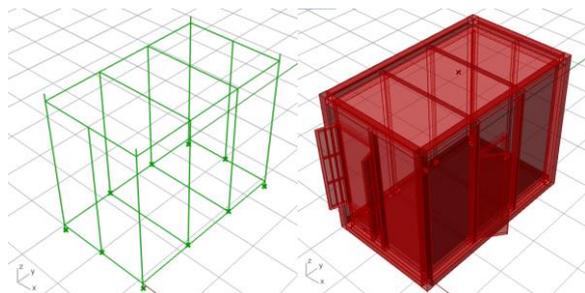


Figura 8: Modelo base em CAD e aplicação de elementos construtivos em BIM. Fonte: Autores.

Os testes em BIM foram iniciados partindo de uma revisão da estruturação dos dados, na qual o algoritmo foi simplificado e dividido em subsistemas que recebem uma única entrada de valores e devolvem um único conjunto de informações.

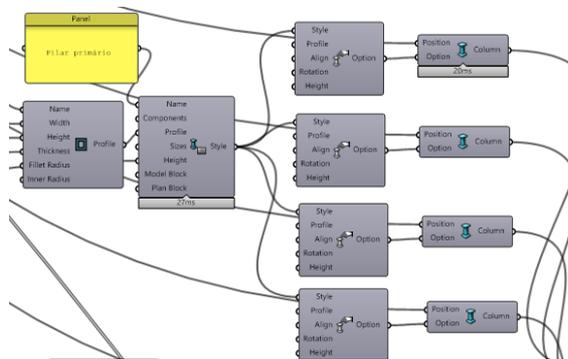


Figura 9: Trecho do código do *VisualArq*. Neste caso, há a aplicação de um mesmo estilo em vários componentes de pilares. Fonte: Autores.

Com o objetivo de estabelecer a conexão entre o algoritmo de geração de layout e o ambiente BIM, foi realizado um teste com o *Geometry Gym*, um *add-on* de *download* gratuito. Ele permite que o trabalho seja feito com componentes dentro de uma semântica mais geral ou com uma estrutura próxima do *Revit*, um dos principais programas BIM do mercado. Em ambos os casos, os arquivos são exportados no formato IFC. As geometrias básicas de CAD funcionaram como guias para os elementos construtivos BIM. Para os pilares, por exemplo, são requisitados linhas; para as vigas, curvas; para as lajes, curvas fechadas ou superfícies e para as paredes, curvas fechadas ou abertas como referência. Em todos os casos, os elementos possuem parâmetros específicos, bem como são agrupados dentro de tipologias onde recebem parâmetros comuns. Esses parâmetros referem-se a questões geométricas - como altura, espessura, largura e comprimento; de transformação - rotação e alinhamento; bem como de identidade - como nome, descrição, ID e materiais.

Em seguida, o mesmo procedimento foi repetido no *VisualArq* (Figura 9), que não é apenas um *add-on* do

Grasshopper, mas um plugin comercializado pela McNeel para se trabalhar com BIM no *Rhino*, permitindo que alguns procedimentos sejam feitos ora em um, ora no outro. Alguns desses, por exemplo, são as propriedades de materiais somente atribuídos no *Rhino* e a necessidade de passar pelo mesmo para exportar em IFC. No geral, seguiu-se a mesma estrutura desenvolvida no *Geometry Gym* com algumas alterações de nomenclaturas e parâmetros.

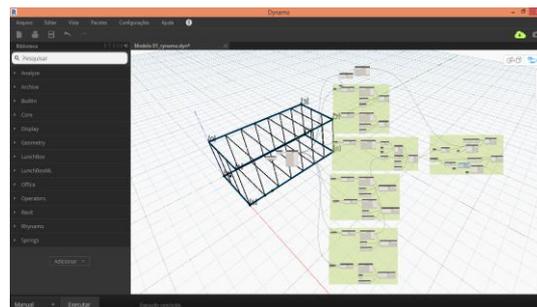


Figura 10: Modelo desenvolvido no *Revit/Dynamo* por meio do *Rhynamo*. Fonte: Autores.

Na sequência, foi desenvolvida uma conexão entre *Grasshopper* e *Revit*, por meio do *Dynamo*, plataforma pertencente ao *Revit* e de programação visual (Figura 10). Para tanto, foi usado o pacote *Rhynamo*, disponível gratuitamente para *download*. Neste caso, foi necessário transformar as geometrias do *Grasshopper* para o *Rhino*, separando-as em *layers* específicas (por meio do comando *bake*). Em seguida, dentro do *Rhynamo*, selecionou-se o caminho do arquivo *Rhino*, seus dados foram ramificados segundo as *layers* e finalmente as famílias do *Revit* foram atribuídas a cada um deles.

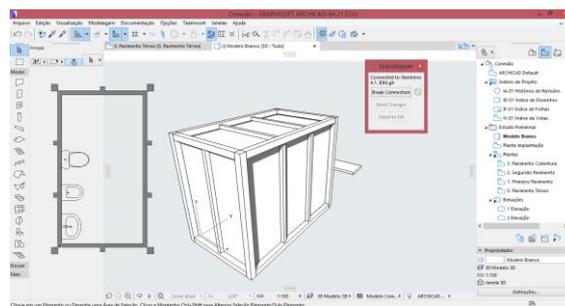


Figura 11: Resultado da *Live Connection* entre *Grasshopper* e *Archicad*. Fonte: Autores.

Por último, testou-se o *Live Connection* (Figura 11), que consiste em um mecanismo de conexão em tempo real entre o processo algorítmico (*Grasshopper*), CAD (*Rhino*) e BIM (*Archicad*). Seu funcionamento exige que ambos os programas estejam abertos e que a conexão seja iniciada no *Archicad*. A partir de então, componentes no *Grasshopper* com a semântica do *Archicad* passam a operar. Todas as operações realizadas no *Grasshopper* passam quase que instantaneamente para o *Archicad*. O algoritmo também pode fazer uso de composições e propriedades de elementos pertencentes ao projeto do *Archicad*. Iniciamos seu estudo repetindo o teste feito nos demais casos. Em essência, os *inputs* e estrutura dos componentes possuem grande similaridade com os do *VisualArq*.

O teste com o *Live Connection* foi mais adiante, iniciando uma união com o algoritmo de *layout* desenvolvido anteriormente. Utilizamos objetos, composições e propriedades do *Archicad* e atribuímos sanitário, pias e bidê aos pontos de posicionamento resultantes do código. A Tabela 1 apresenta uma síntese das vantagens e desvantagens dos diferentes sistemas testados para a conversão do layout esquemático para o BIM.

Tabela 1: pontos positivos e negativos entre as conexões estudadas entre *Grasshopper* (modelagem paramétrica e algorítmica) e BIM.

Software	Pontos positivos	Pontos negativos
Geometry Gym	- Gratuito para estudantes; - Possibilidade de trabalhar com uma linguagem genérica BIM ou próxima da sintaxe do <i>Revit</i> ; - Trabalho independente do <i>Rhinoceros</i> ;	- Interface; - Demanda inicial de trabalho maior que em outros softwares BIM, como <i>Revit</i> e <i>Archicad</i> ;
VisualArq	- Interface; - Bom fluxo de trabalho de elementos CAD para BIM e vice-versa;	- Demanda inicial de trabalho maior que em outros softwares BIM, como <i>Revit</i> e <i>Archicad</i> ;
Live Connection	- Boa interoperabilidade entre a sintaxe e elementos do <i>Archicad</i> com os componentes do <i>Grasshopper</i> ; - Independência das plataformas;	- Maior demanda de processamento que os outros softwares estudados; - Necessidade de conhecimento da linguagem do <i>Archicad</i> e do <i>Grasshopper</i> .
Rhynamo	- Gratuito; - Rápida transposição de objetos para a plataforma do <i>Revit</i> ;	- Necessidade de conhecimento das sintaxes do <i>Dynamo</i> e do <i>Grasshopper</i>

RESULTADOS PRELIMINARES

A etapa inicial de identificação de *Design Units*, entendimento de padrões, estabelecimento de regras e manipulações resultou em três tipologias de *layouts* de banheiros que foram trabalhadas nas etapas seguintes da pesquisa.

No segundo momento, a metodologia incremental para a construção do algoritmo - partindo de parametrizações elementares e algoritmos simples até chegar a uma maior complexidade - viabilizou um progressivo amadurecimento da lógica algorítmica para a resolução de *layouts*, resultando em um completo sistema de geração de *layout* para banheiros pré-fabricados que comporta inúmeras combinações. Pode-se escolher, simplesmente alterando valores de variáveis, por exemplo, se o banheiro conterà *shaft* ou não, uma ou duas pias, inserir um bidê, definir em qual parede será inserida a porta e quais serão as suas dimensões. Além disso, o algoritmo devolve análises das manipulações, como área, perímetro, comprimento dividido por largura, área dividida por perímetro, área do box, área do banheiro dividida pela área do box, área do banheiro subtraindo a área do box e quantidade de montantes por parede, bem como sugere possíveis mudanças caso algumas relações fujam de intervalos aceitáveis.

O código em *Grasshopper* conferiu uma maior liberdade para exploração de possibilidades nas etapas iniciais de desenvolvimento com um foco no entendimento das relações entre os parâmetros antes da atribuição de elementos construtivos. Na sua última versão, chegou-se a um alto grau de detalhamento. Para a estrutura, por exemplo, foram levadas em conta as espessuras das chapas, folgas para encaixe das vigas horizontais com os perfis verticais, e três casos de espaçamento que redistribuem os perfis da estrutura e encaixes das placas de *drywall*.

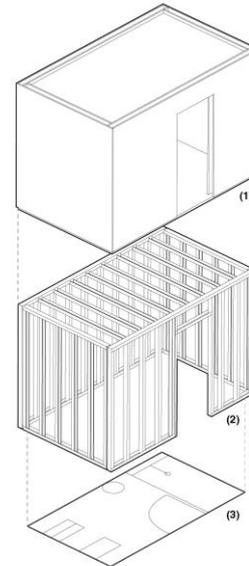


Figura 12: Resultado da etapa 1B – (1) vedação, (2) estrutura e (3) layout. Fonte: Autores.

A separação em cinco subsistemas (*layout*, modelo 3D simplificado, *drywall* 2D, *drywall* 3D e estrutura 3D) proporcionou uma melhor organização do trabalho e abriu possibilidade para manutenção futura do algoritmo (Figura 12).

Na terceira etapa de desenvolvimento, a sistematização dos dados em subsistemas com apenas uma entrada e uma saída de dados imprimiu uma maior clareza e flexibilidade ao fluxo de informações. Por exemplo, a partir de uma única saída do sistema que continha curvas para vigas e pilares, foi possível testar diferentes componentes que trabalham com BIM.

Em todos os casos, trabalhar com BIM é trabalhar com semântica construtiva, ou seja, o algoritmo faz uso de uma estrutura pré-existente e de suas propriedades e devolve elementos que são enquadrados dentro de classificações pré-estabelecidas. Isso faz com que, em geral, os *add-ons* de BIM contidos no *Grasshopper* possuam estruturas lógicas parecidas, permitindo uma certa interoperabilidade para o usuário. Tanto o *Rhynamo* quanto o *Dynamo*, por exemplo, seguem a mesma estrutura lógica que se baseia em geometrias (curvas, pontos, superfícies) e recebem elementos construtivos (pilar, laje, parede). Entretanto, há também divergências entre as linguagens das plataformas de criação de algoritmos em BIM, como ocorre na criação de um pilar no *Geometry Gym* que requer uma linha como referência, enquanto que os componentes do *Archicad* exigem um

ponto inicial e outro final. Essa e outras especificidades exigem diferentes formas de estruturação do código, mas a lógica permanece inalterada.

Os resultados preliminares da etapa dois foram discutidos com a indústria de banheiros pré-fabricados paulista. Segundo o gerente e um arquiteto da indústria, em princípio a hipótese de que o desenvolvimento de um sistema de automação de layout em BIM viabilizaria a produção em massa de produtos únicos (ao invés de pequenas séries) foi confirmada. Como foi exposto pelos representantes dessa indústria (bem como pela outra indústria italiana do mesmo segmento), atualmente exige-se um número mínimo de unidades iguais para viabilizar a produção por causa, entre outras coisas, do custo do projeto. Com o processo de geração automatizada dos projetos, a indústria reconheceu a viabilidade de produção de banheiros personalizados um a um.

ETAPAS FUTURAS

Esta pesquisa terá continuidade, nos próximos meses, com a consolidação de um sistema completo da geração automatizada dos *layouts* ao detalhamento estrutural, elétrico e hidráulico de banheiros, utilizando as pontes aqui apresentadas entre o processo algorítmico e BIM, continuando a aplicação em pré-fabricados e alimentando o modelo com mais informações. Além disso, pretende-se também extrair automaticamente do modelo os desenhos necessários para a fabricação de componentes por meio de equipamentos operados por controle numérico. Por exemplo, serão gerados perfis para corte das placas de gesso acartonado em fresadora CNC. Como resultado ao final, pretende-se propor uma parceria com uma indústria de banheiros pré-fabricados, o que nos permitirá, efetivamente, confirmar a viabilidade da solução proposta.

DISCUSSÃO

As transformações socioeconômicas demandam dos arquitetos uma revisão dos processos produtivos e da profissão.

Como os resultados preliminares apontaram, parece ser possível viabilizar uma maior personalização em massa ao fornecer um sistema que automatiza o processo de projeto, permitindo extrair informações para a documentação e a fabricação. Por consequência, os arquitetos podem fazer uso de processos industrializados subdividindo o edifício (ou outro produto) em módulos, partes e conjuntos (como ocorre na produção de veículos, navios e aviões). Desse modo, cada parte pode ser avaliada, automatizada, desenvolvida, produzida independentemente das outras e todas são reunidas ao final. O resultado dessa revisão de processo produtivo e perspectiva é a redução de tempo e custo do empreendimento, bem como o aumento da qualidade da construção e das possibilidades de projeto, em comparação aos atuais sistemas pré-fabricados em massa.

O ciclo de produção, construção e gestão é fechado quando inserimos algoritmos em BIM, onde pode ocorrer a documentação e a comunicação entre os profissionais envolvidos na construção. Dentro deste contexto, os arquitetos têm a possibilidade de serem integradores de conhecimentos, uma nova versão do *master builder*. O

processo algorítmico, o BIM e a industrialização podem fazer com que a arquitetura ofereça uma nova alternativa projetual, além da produção artesanal ou da produção em massa: a personalização em massa. As demandas deste século criam horizontes a serem percorridos por arquitetos e novas formas de atuação desses profissionais. Mas é importante ter em mente que essas demandas também exigem novos tipos de conhecimento, dentre os quais a lógica de programação é provavelmente o principal deles.

AGRADECIMENTOS

Agradecemos à Fundação de Amparo à Pesquisa do Estado de São Paulo pelo financiamento da pesquisa (processos 2017/22105-0 e 2017/09702-9); à prof. Ingrid Paoletti e ao dr. Roberto Naboni e demais pesquisadores do ACTLAB do Politécnico de Milão pelo acolhimento e pela colaboração; e às indústrias TecnoBagno, Sanika, Italcementi e Stylcomp pelas informações disponibilizadas.

REFERÊNCIAS

- Caetano, I., Leitão, A., Bastos, F. (2016, novembro). Aplicação do design generativo nas tecnologias BIM. In: 1º Congresso Português de Building Information Modelling, Universidade do Minho, Guimarães, 239 – 249.
- Ching, F. D. K. (2010). Técnicas de construção ilustradas (4ª ed.). Porto Alegre: Bookman, 9.26-9.30.
- Davis, S. M. (1990). Futuro perfeito. São Paulo: Nobel.
- Eastman, C. (1968). Explorations of the cognitive processes in design. Pittsburgh: Dept. of Computer Science, Carnegie-Mellon University. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.49.8.6603&rep=rep1&type=pdf>.
- Eastman, C., Liston, K., Sacks & R., Teicholz, P. (2013). Manual de BIM: um guia de modelagem da informação da construção para arquitetos, engenheiros, gerentes, construtores e incorporadores (1ª ed.). Porto Alegre: Bookman.
- Garber, R. (2014). BIM Design: Realising the Creative Potential of Building Information Modelling (1ª ed.). Nova York: John Wiley & Sons, Ltd.
- Gurgel, M. (2002). Projetando espaços: guia de arquitetura de interiores para áreas residenciais (6ª ed.). São Paulo: Senac.
- Jiang, H. & Yen, C.-C. (2009, outubro). Protocol analysis in design research: a review. International Association of Societies of Design Research (IASDR) 2009, Rigor and Relevance in Design, Seoul, Korea, 147-156. Retrieved from: <http://www.iasdr2009.or.kr/Papers/Orally%20Presented%20Papers/Design%20Method/Protocol%20Analysis%20in%20Design%20Research%20-%20a%20Review.pdf>.
- Kieram, S., Timberlake, J. (2004). Refabricating Architecture: How Manufacturing Methodologies Are Poised to Transform Building Construction (2ª Ed). Nova York: McGraw-Hill.
- Leach, N. (2012). 'Parametrics Explained', in Leach, Yuan (Eds.), Scripting the Future (pp. 9-13). Shan-ghai: Tongji University Press.
- Mitchell, W. J. (1975). The theoretical foundation of computer-aided architectural design. Environment and Planning B, V.2. 127-150.
- Neufert, P. (2009). Arte de projetar em arquitetura (17ª ed.). São Paulo: Gustavo Gili, 246-251.
- Panero, J., Zelnik, M. (2002). Dimensionamento humano para espaços de interiores. Barcelona: Editorial Gustavo Gili.