

# **Desenho Generativo: uma nova fase no processo de projeto**

**Rita Margarida Serra Fernandes**

**Resumo Alargado**

**Júri**

Presidente: Prof. Dr. Ana Cristina dos Santos Tostões

Orientador: Prof. Dr. António Paulo Teles de Menezes Correia Leitão

Vogal: Prof. Dr. Francisco Manuel Caldeira Pinto Teixeira Bastos

**Junho 2013**

# 1. INTRODUÇÃO

Os problemas de projecto são pouco estruturados (Simon, 1973). Em arquitetura, projetar envolve uma resposta a um problema que muitas vezes não é claro desde o início e, portanto, o processo de projeto envolve a compreensão do problema (Hudson, 2010). Além disso, o projecto opera tanto no mundo natural como social, e ambos introduzem restrições (Gero, 1990). Estas restrições podem ser impostas por condições externas, como por exemplo, o local, o clima, os custos, as leis, ou pelas preocupações do arquiteto que estabelece as suas próprias metas e objectivos. Ainda que com estas inúmeras restrições, cada problema de projeto é caracterizado por uma ampla gama de possíveis soluções. Isto significa que projetar também implica um processo de seleção entre soluções que considere muitas alternativas. Neste contexto, projetar pode ser visto como um processo evolutivo (Alfaris, 2009) que evolui a partir de ideias abstratas para soluções mais complexas e concretas, atendendo às restrições e incorporando novas exigências. Este processo avança e recua tantas vezes quantas forem necessárias, a fim de alcançar uma boa solução para o problema de projeto.

As ferramentas CAD sempre tiveram um papel essencial nas atividades de projeto, contribuindo para aumentar a sua eficiência, em particular, na modelação de uma solução. Infelizmente, estas ferramentas estão atualmente a ser utilizadas apenas como uma versão mais eficiente da abordagem baseada em papel, uma abordagem que não ajuda os arquitetos a lidar com a mudança, em particular na exploração de diferentes soluções de projeto ou na adaptação à evolução dos seus requisitos.

Recentemente, novas abordagens têm sido introduzidas no processo de projeto, as quais são mais aptas a lidar com a mudança. O Desenho Generativo é uma delas e pode ser definido como a criação de formas determinada por algoritmos.

Esta dissertação defende um ponto principal: a integração do Desenho Generativo como uma nova fase no processo de projeto simplifica drasticamente a incorporação de mudanças. Em particular, é proposta uma abordagem algorítmica para o processo de projeto que supera as limitações da abordagem tradicional no uso das ferramentas CAD.

Esta abordagem requer a formalização das intenções de projeto, para a qual utilizámos programação. O uso desta abordagem tem um custo inicial. Contudo, argumenta-se que este custo é altamente recompensado quando os arquitectos necessitam de lidar com a mudança, permitindo-lhes explorar muitas alternativas diferentes, de forma rápida e sem esforço.

Para avaliar a abordagem proposta foi desenvolvido um modelo tridimensional de um caso de estudo, um edifício grande e complexo, usando uma representação puramente baseada em programação.

## 2. ABORDAGENS PARA A MODELAÇÃO: TRADICIONAL vs ALGORÍTMICA

Para lidar com a mudança, um arquiteto interage com uma ferramenta CAD assumindo diferentes papéis e dispendendo diferentes esforços.

Nesta secção são caracterizados os papéis do arquiteto e é avaliado o esforço necessário para lidar com a mudança em modelos desenvolvidos usando duas abordagens significativamente diferentes: a tradicional, baseada em operações de modelação explícitas, e uma moderna, baseada na implementação de algoritmos, a qual é proposta nesta dissertação.

### *2.1 Abordagem tradicional*

Na abordagem tradicional, que ainda rege a prática de hoje, o arquiteto interage com representações gráficas bi ou tridimensionais que ajudam essencialmente na produção de elementos de comunicação como desenhos ou *renderings*. Estas ferramentas CAD fornecem operações que permitem ao arquiteto criar, manipular, editar e relacionar partes de um modelo mas a sua utilização é principalmente descritiva: a ferramenta é usada como um substituto do meio de comunicação baseado em papel e dos modelos físicos convencionais.

Usando esta abordagem é possível lidar com projetos complexos, desde que a sua lógica seja claramente compreendida antes das tarefas de modelação. Contudo, é geralmente muito difícil lidar com mudanças, em particular, em fases avançadas do processo de projecto, pois elas tendem a afetar muitas partes do modelo, requerendo muito tempo e esforço. Em suma, embora estas ferramentas facilitem a produção e visualização de geometrias complexas, também limitam a exploração e restringem efetivamente o processo de projeto (Woodburry, 2010).

### *2.2 Abordagem algorítmica*

Contrariamente à abordagem tradicional, a abordagem baseada em algoritmos descreve o projeto como um programa escrito numa linguagem formal (textual, visual, ou ambas). Este programa pode então ser executado por um computador, gerando um modelo visível como resultado. Para escrever este programa, o arquiteto deve saber como decompor a lógica do projecto e deve seleccionar os elementos de programação apropriados para representar as partes fundamentais, bem como as relações entre essas partes.

Utilizando uma abordagem baseada em algoritmos, o arquiteto interage com o mecanismo computacional que gera a representação digital. Obviamente, as ferramentas de CAD convencionais também realizam computações para gerar a representação digital, a diferença é que na abordagem tradicional elas são escondidas do utilizador que interage diretamente com o CAD.

O pensamento algorítmico requer a descrição do projeto através de um processo passo-a-passo em que as ideias e intenções têm de ser claramente definidas, as relações entre elementos explícitas, os requisitos priorizados e os mecanismos de controlo desenvolvidos. Isto exige um óbvio esforço de formalização.

Um arquiteto descreve essencialmente projetos invés de processos e, usualmente, as representações são imprecisas, dependendo da capacidade humana para as interpretar de forma apropriada, uma capacidade que um computador não tem (Woodburry, 2010). Devido ao grande esforço inicial necessário para formalizar o projeto, escrever este algoritmo pode não ser uma tarefa trivial e pode ser até menos rentável do que a abordagem tradicional. Contudo, quando se torna necessário incorporar mudanças num projecto, este esforço inicial é rapidamente recuperado. Dependendo da estrutura do programa, uma mudança pode ser tão simples como atualizar o valor de um parâmetro, ou tão complexo como atualizar partes do programa. Contudo, em todos os casos, o programa modificado é usado para regenerar o modelo. Isto é possível porque o projeto é explicitamente representado no programa, e mudanças no projeto são convertidas em alterações no programa. A rápida regeneração do modelo permite ainda a exploração de espaços de solução muito maiores, o que é vantajoso na tomada de decisões.

Apesar de todas as suas vantagens, existe uma desvantagem no uso de uma representação puramente baseada em programação: se foram efetuadas alterações manuais ao modelo gerado estas serão perdidas quando o programa voltar a ser executado e o modelo regenerado. A solução mais óbvia para este problema é simplesmente proibir quaisquer alterações manuais ao modelo e forçar sempre a sua implementação no programa. Outra possibilidade é a de permitir alterações manuais ao modelo, por exemplo, para o estudo de diferentes soluções, que são depois incorporadas no programa para o sincronizar com o modelo.

### **3. CASO DE ESTUDO**

Dado que a abordagem proposta nesta dissertação introduz um passo adicional de formalização algorítmica no processo de projeto, é importante avaliar se o tempo gasto nesta etapa permite uma significativa redução no tempo e esforço necessário para lidar com mudanças, em particular nas fases mais avançadas do projeto.

De forma a avaliar a abordagem algorítmica para o projecto, foi feita uma experiência: foi formalizado o projecto de um edifício complexo, o edifício Market Hall dos arquitetos MVRDV, cujo processo de modelação foi executado até ser obtido um modelo muito detalhado do edifício.



**Figura 3.1** *Rendering* do edifício Market Hall (fonte: <http://www.mvrdv.nl>).

Foi delineada uma estratégia de modelação genérica que se inicia com a análise das restrições impostas externamente e do processo, conceitos e intenções dos arquitetos. Em seguida, o processo de modelação foi decomposto segundo duas perspectivas: (1) para coincidir com a decomposição do processo de projeto de forma a aumentar progressivamente o detalhe e a definição do modelo, e (2) para coincidir com a decomposição do próprio projeto, dividindo o modelo nos elementos relevantes que ele representa. De seguida, esta estratégia foi aplicada ao caso de estudo, a qual é descrita de seguida.

O edifício Market Hall está localizado numa praça e não se encontra adjacente a outros edifícios o que permitiu uma modelação livre de contexto, sem impor restrições relacionadas com o terreno, a forma, ou a dimensão do edifício. Como pode ser visto na Figura 3.1, o edifício é caracterizado por uma forma não convencional que cria um grande *hall*, o qual é fechado por duas fachadas de vidro. Esta forma curvada foi o ponto de partida do processo de projeto uma vez que ela delimita tanto os elementos do edifício como os seus espaços interiores.

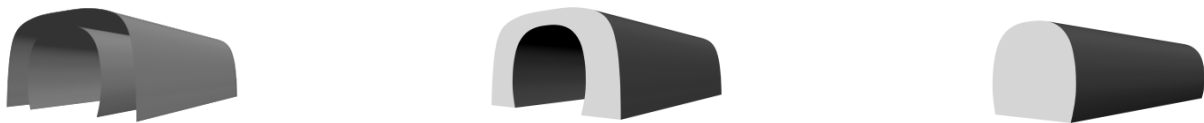
No entanto, em vez de capturar a exata geometria do edifício, foram capturadas as ideias subjacentes ao mesmo, de modo a ser possível gerar vários modelos diferentes, incluindo o Market Hall.

Para este fim, cada ideia presente no projeto foi formalizada e abstraída, e foi implementada numa linguagem de programação. Isto significa que foram formalizadas todas as formas, relações, repetições, variações, e regras identificadas no Market Hall. Ao mesmo tempo, todas estas formalizações foram generalizadas e parametrizadas para aumentar a sua aplicabilidade.

Quando se analisa o edifício torna-se evidente que há fortes dependências entre os seus elementos. Por exemplo, a laje da cobertura apenas pode ser representada depois de se decidir a forma global do edifício. Como outro exemplo, pode-se considerar as posições das janelas que apenas podem ser

decididas depois de se ter conhecimento acerca da localização das lajes e das paredes verticais do edifício. Consequentemente, o processo de modelação utilizado deve estabelecer uma ordem entre a modelação dos diferentes elementos.

O processo de projeto utilizado para construir o modelo pode ser dividido em quatro fases (Figura 3.2 e 3.3): (1) formalização da forma global do edifício, (2) modelação dos elementos principais do edifício, por exemplo, paredes e lajes, (3) definição das aberturas e modelação dos caixilhos, e por último, (4) modelação dos elementos de detalhe (prumos das guardas das varandas e detalhe das fachadas de vidro).



**Figura 3.2** Fase 1 do modelo. Formalização da forma global do edifício: geometria subjacente.



**Figura 3.3** Restantes fases do modelo. Esquerda: fase 2. Centro: fase 3. Direita: fase 4.

Na primeira fase foi definida a geometria subjacente que captura os principais conceitos da forma global do edifício. Esta geometria foi usada como *input* nas restantes fases do processo.

A decomposição do processo em quatro fases, cada uma aumentando o detalhe do modelo, levou à existência de dependências entre elementos, permitindo a propagação automática de mudanças. Na Figura 3.4 (página 6), é apresentada a hierarquia do modelo de acordo com as quatro fases do processo e as dependências entre elementos. Como pode ser visto, todos os elementos do edifício dependem da sua forma global, a qual foi formalizada na primeira fase. Adicionalmente, esta figura também mostra as duas decomposições, a do processo de modelação que coincide com a do processo de projeto, de forma a aumentar progressivamente o detalhe e definição do modelo, e para coincidir com a decomposição do projeto, decompondo o modelo nos elementos relevantes que ele representa.

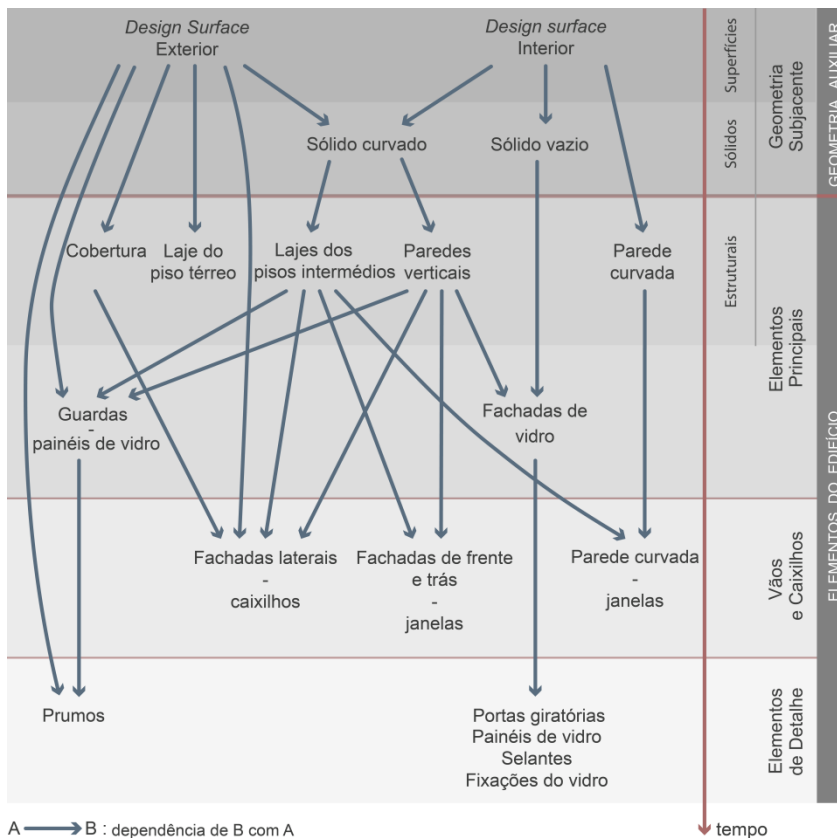


Figura 3.4 Fases do processo de projeto e hierarquia do modelo.

## 4. AVALIAÇÃO

Para avaliar a capacidade de resposta da abordagem algorítmica à mudança e ao crescimento dos requisitos inerentes ao processo de projeto, foram simulados diferentes tipos de mudança e medido o seu impacto, em particular no tempo e esforço necessário para a sua implementação comparativamente com a abordagem tradicional.

Foram testados cinco cenários principais. Partindo do Market Hall completo, foi simulada a introdução das seguintes mudanças: (1) mudando apenas a forma global do edifício, (2) mudando apenas a dimensão ou posição de elementos do edifício, (3) mudando simultaneamente a forma global do edifício e as dimensões ou posições dos seus elementos, (4) mudando a forma de alguns elementos do edifício, e (5) mudando a ordem de elementos do edifício. Por último, (6) foi também simulada a reutilização do programa em outros projectos, mudando a formalização da forma global do edifício.

### 4.1 Cenário 1: mudando apenas a forma global do edifício

Uma mudança na forma do edifício causa muitas e diferentes mudanças nos seus elementos, nomeadamente na sua geometria, posição, direção e número de subelementos.

O efeito desta mudança é claramente visível na Figura 4.1 (à direita) quando comparado com o Market Hall (Figure 4.1, à esquerda), onde um edifício diferente foi gerado mudando a forma global do edifício e a sua secção ao longo do eixo longitudinal. Estes resultados foram produzidos mudando apenas o valor de quatro parâmetros, o que foi feito em minutos.

Este é o tipo de mudança que é bastante difícil de introduzir usando uma abordagem tradicional à modelação uma vez que a maioria dos elementos teria de ser refeita ou as suas posições e direções recalculadas. Este processo seria certamente muito demorado e laborioso e, para cada mudança na forma global do edifício, o esforço teria de ser repetido.



**Figura 4.1** Esquerda: Market Hall dos MVRDV gerado a partir do programa desenvolvido. Direita: Um edifício diferente gerado mudando a forma global do edifício e a sua secção ao longo do eixo longitudinal.

#### ***4.2 Cenário 2: Mantendo a forma do edifício e mudando a dimensão ou posição dos seus elementos***

Outra possível alteração ao projeto pode requerer uma mudança nas dimensões ou posições de certos elementos. Para dar um exemplo concreto, qualquer mudança na posição das lajes dos pisos intermédios do edifício teria repercussões em vários outros elementos, por exemplo, na dimensão e posição dos caixilhos das fachadas laterais ou na geometria e posição dos painéis de vidro das guardas das varandas.

O efeito desta mudança é visível na Figura 4.2 (à direita). Esse edifício foi gerado mudando apenas o parâmetro que contem as cotas dos pisos intermédios, uma mudança que foi implementada em minutos. Observando essa figura e comparando com o Market Hall (Figura 4.2, à esquerda), torna-se claro que esta mudança não tem muito impacto visual, contudo, vários elementos do edifício foram afetados por essa mudança, por muito pequena que ela seja.



**Figura 4.2** Esquerda: Market Hall gerado a partir do programa desenvolvido. Direita: Edifício gerado após a mudança da posição das lajes.

Considerando uma abordagem tradicional, o arquiteto teria de refazer as lajes dos pisos intermédios e muitos outros elementos que necessitariam de ser ajustados.

Dado este exemplo, é possível concluir que uma mudança que pode até parecer simples de resolver usando uma abordagem tradicional, continuaria a requerer um enorme esforço para modificar todos



os elementos afetados, de facto, semelhante ao esforço necessário para produzir os elementos pela primeira vez.

#### **4.3 Cenário 3: Mudando a forma do edifício, a sua dimensão e as posições ou dimensões dos seus elementos**

Outro efeito da introdução de uma mudança na forma global do edifício é visível na Figura 4.3. Nestes casos, os diferentes edifícios foram gerados mudando mais drasticamente as curvas de secção e usando diferentes secções ao longo do eixo longitudinal. Além da mudança na forma do edifício outros parâmetros foram alterados, por exemplo, o número e a posição dos pisos e paredes, o número das janelas das fachadas de frente e trás, o número e a posição das portas giratórias ou a dimensão de elementos tais como as janelas ou painéis de vidro. As relações entre os elementos foram mantidos e adaptadas a uma diferente escala.



**Figura 4.3** Dois edifícios gerados introduzindo várias alterações.

De forma semelhante ao primeiro e segundo cenários, estes resultados foram produzidos mudando um pequeno número de parâmetros, uma tarefa que foi executada em minutos. Estas mudanças são muito mais difíceis de introduzir usando uma abordagem tradicional do que as dos dois primeiros cenários dado que elas são uma combinação desses dois tipos de mudança.

Usando uma abordagem tradicional, não seria possível reutilizar nenhum dos elementos de modelo para modelo. O número de alterações seria tão grande que provavelmente seria mais fácil começar a construir um modelo do início, perdendo, assim, o esforço já gasto a modelar o modelo anterior.

#### **4.4 Cenário 4: Mudando a forma de elementos do edifício**

Neste cenário são avaliadas mudanças mais drásticas que implicam modificações não apenas nos parâmetros mas também nos algoritmos desenvolvidos. Esta mudança foi simulada para dois elementos diferentes, nomeadamente para as janelas das fachadas de frente e trás e para os prumos das guardas das varandas.

Em ambos os casos a lógica subjacente ao processo de modelação continuou válida e a modificação de ambos os procedimentos foi semelhante. Estes procedimentos, que lidavam com formas específicas, foram generalizados para aceitar um parâmetro adicional para um procedimento que produz a forma desejada. Para dar um exemplo concreto, na Figura 4.4 estão ilustrados os resultados da generalização efetuada para os prumos. Neste caso, a secção dos prumos foi abstraída

(inicialmente rectangular para produzir prumos iguais aos do Market Hall) para permitir qualquer outra secção o que se encontra ilustrado para uma secção circular e para um perfil em forma de I. As modificações efetuadas no programa foram concluídas em menos de uma hora, o que é muito pouco tempo para uma tarefa que, na abordagem tradicional, requer alterações a muitos elementos.



**Figura 4.4** Mudando a secção dos prumos. Esquerda: rectangular. Centro: circular. Direita: perfil em forma de I.

#### **4.5 Cenário 5: Mudando a ordem de elementos do edifício**

O Market Hall é caracterizado por caixilhos nas fachadas laterais que seguem uma sequência em cada piso. Neste cenário foi avaliado o custo de mudar a ordem dos elementos, sem mudar qualquer outra coisa.

Para mudar a ordem dos tipos de caixilhos foi apenas alterada uma sequência de procedimentos, argumentos em outros procedimentos. Este efeito é visível na Figura 4.5 e foi conseguido em segundos.



**Figura 4.5** Mudando a ordem dos caixilhos das fachadas laterais. Esquerda: Caixilhos seguindo a sequência do Market Hall. Direita: Caixilhos com uma ordem diferente.

Nem a forma dos caixilhos nem a sua posição foi modificada. Desta forma, numa abordagem tradicional, o arquitecto poderia reutilizar os elementos anteriormente definidos e usar as operações “cortar”, “copiar” e “colar” para os reordenar, o que resolveria facilmente o problema. Contudo, em edifícios com um número tão grande de elementos como o Market Hall esta tarefa seria certamente laboriosa, demorada e repetitiva, o que a torna muito susceptível à introdução de erros.

#### **4.6 Cenário 6: Mudando a formalização da forma do edifício**

A experiencia mais drástica foi feita simulando a mudança da formalização do edifício de uma forma curvada e longitudinal para uma forma paralelepípedica. Na Figura 4.6 (página 10) é possível observar o efeito desta mudança.

Durante o processo de modelação, em particular, na fase de revisão, foi considerada a reutilização do programa em futuros projetos. Desta forma, os procedimentos foram abstraídos para receber entidades abstratas, por exemplo, sólidos, superfícies ou curvas, e apenas o último procedimento que

produz um elemento foi particularizado para receber os parâmetros que foram utilizados para formalizar a forma global do Market Hall, isto é, para ser aplicado à geometria subjacente definida na primeira fase do modelo. Para obter os resultados ilustrados foi apenas necessário definir um novo conjunto de procedimentos, a geometria subjacente, para formalizar a nova forma. Esta geometria auxiliar foi usada como *input* nos procedimentos já definidos, aos quais não foram feitas quaisquer modificações. A definição deste novo conjunto de procedimentos foi feita em aproximadamente meia hora.

Caso exista a intenção de uma equivalente reutilização na abordagem tradicional à modelação, o arquiteto pode usar alguns elementos como os caixilhos das fachadas laterais de outros modelos quando a sua dimensão global é igual, mas, no geral, ele teria de começar o modelo do início



Figura 4.6 Mudança na formalização da forma do edifício: duas formas paralelepípedicas.

## 5. CONCLUSÕES

Os processos de projeto são caracterizados por mudança. Infelizmente, as ferramentas CAD estão atualmente a ser utilizadas apenas como uma versão mais eficiente da abordagem baseada em papel, uma abordagem que não ajuda os arquitectos a lidar com a mudança, em particular, na exploração de diferentes soluções de projeto ou na adaptação à evolução dos seus requisitos.

Recentemente, novas abordagens têm sido introduzidas no processo de projeto, as quais são mais aptas a lidar com a mudança. O Desenho Generativo é uma delas e pode ser definido como a criação de formas determinada por algoritmos.

Esta dissertação defende um ponto principal: a integração do Desenho Generativo como uma nova fase no processo de projeto simplifica drasticamente a incorporação de mudanças. Em particular, é proposta uma abordagem algorítmica para o processo de projeto que supera as limitações da abordagem tradicional no uso das ferramentas CAD.

Esta proposta introduz um passo de formalização do projeto usando uma representação baseada em programação. É defendido que este passo acelera a incorporação das inevitáveis mudanças que ocorrem com a evolução do projeto. Para avaliar a abordagem proposta, foi formalizado o projeto de um edifício, o edifício Market Hall dos MVRDV e concluiu-se que os ganhos em tempo e esforço são consideráveis.

Para avaliar a formalização algorítmica desenvolvida foram simulados vários tipos de mudança e medido o seu impacto, em particular no tempo e esforço necessário para a sua implementação comparativamente com a abordagem tradicional no uso das ferramentas CAD. Estas simulações permitiram verificar a relevância do uso de uma abordagem algorítmica ao projeto desde as fases iniciais do processo de projeto, e também explorar os limites da representação baseada em programação.

Com esta avaliação ficou provado que a abordagem proposta é suficientemente flexível, não apenas para incorporar mudanças previamente antecipadas, mas também mudanças que não foram planeadas. Esta conclusão pode ser estendida à prática arquitetónica na qual as mudanças surgem frequentemente sem serem previstas. Apesar do passo adicional de formalização do projeto requerer um óbvio esforço inicial, é defendido que não é apenas possível mas, efetivamente, rentável o uso de uma abordagem algorítmica ao projeto como uma nova fase no processo de projeto. Na verdade, o custo inicial, nomeadamente o tempo e esforço, é rapidamente recuperado quando são necessárias mudanças.

A generalização efectuada do projeto Market Hall permitiu produzir edifícios diferentes do Market Hall, o que foi possível devido à propagação automática de mudanças entre os elementos do edifício. Desta forma, uma abordagem algorítmica ao projeto provou ser alinhada com as necessidades do processo de projeto permitindo não só a introdução de mudanças, sem esforço, mas também a produção de um número infinito de modelos diferentes explorando diferentes soluções para o projeto e, desta forma, assistindo verdadeiramente os arquitetos na tomada de decisões.

A experiência efectuada provou também que esta abordagem pode ser integrada logo após o desenvolvimento do conceito que estrutura o projeto. Esta integração na fase inicial do processo de projeto permite aos arquitetos trabalhar com uma ferramenta poderosa e eficaz ao longo de todo o processo, desde a fase conceptual até à construção. A avaliação efectuada revelou ainda que a abordagem algorítmica permite e facilita o uso de estratégias de customização em massa, nas quais o esforço necessário para produzir um programa é recuperado através da sua utilização para modelar vários edifícios semelhantes.

O desenvolvimento do modelo permitiu fornecer boas-práticas para o desenvolvimento de modelos semelhantes. Para reduzir drasticamente o esforço para lidar com a mudança no processo de projeto, pode ser destacada a importância da definição dos parâmetros e a estratégia utilizada para construir o modelo. Esta estratégia pode ser resumida em quatro passos principais: (1) analisar o projeto e formalizar as suas intenções, (2) abstrair e generalizar essas intenções, prevendo as mudanças que podem surgir durante processo de projeto, (3) decompor a definição dos elementos em subtarefas, e (4) definir essas subtarefas de modo a que elas funcionem com entidades abstratas. A abstração e decomposição são de grande importância para incorporar mudanças que não tenham sido previamente antecipadas.

Em síntese, uma abordagem algorítmica pode ser integrada no processo de projeto para o tornar mais rentável a lidar com mudanças, beneficiando o arquiteto com uma vantagem competitiva. A abordagem proposta não exclui outras abordagens para o processo de projeto, é uma fase adicional que não substitui o trabalho criativo do arquiteto. Em vez disso, permite-lhe ir mais longe na exploração de diferentes soluções de projeto.

No início deste trabalho foi proposto determinar os limites da representação baseada em programação. Considerando o elevado grau de detalhe dos modelos desenvolvidos é possível concluir que é viável ir bastante longe utilizando esta representação.

## 6. BIBLIOGRAFIA

Alfaris, A. (2009) "Emergence through conflict: the Multi-Disciplinary Design System (MDDS)", Tese de Doutoramento, Massachusetts Institute of Technology.

El-Khaldi, M. (2007) "Mapping boundaries of generative systems for design synthesis", Tese de Mestrado, Massachusetts Institute of Technology.

Gero, J. (1990) "Design Prototypes: A Knowledge Representation Schema for Design", *AI Magazine*, 11, nº4, pp. 26-36.

Hudson, R. (2010) "Strategies for parametric design in architecture: an application of practice led research", Tese de Doutoramento, University of Bath.

Simon, H. (1973) "The Structure of Ill Structured Problems", *Artificial Intelligence*, 4, nº3, pp. 181-201.

Woodburry, R. (2010) *Elements of Parametric Design*, New York: Routledge.



# **Generative Design: a new stage in the design process**

**Rita Margarida Serra Fernandes**

**Extended Abstract**

## **Examination Committee**

Chairperson: Prof. Dr. Ana Cristina dos Santos Tostões

Supervisor: Prof. Dr. António Paulo Teles de Menezes Correia Leitão

Members of the Committee: Prof. Dr. Francisco Manuel Caldeira Pinto Teixeira Bastos

**June 2013**

# 1. INTRODUCTION

Design problems are ill-structured (Simon, 1973). In architecture, design involves a response to a problem which is often not clear at the outset and therefore the design process involves developing an understanding of the problem (Hudson, 2010). Moreover, the designed artifact operates in natural and social worlds, and they both introduce constraints (Gero, 1990). These constraints can be imposed by external conditions, e.g., site, climate, cost, laws, or by the concerns of the designer who establishes his own goals. Even with these countless constraints each design problem is characterized by a large range of possible solutions. This means that design also implies a process of selection amongst solutions, considering many tradeoffs. In this context, the design activity can be seen as an evolutionary process (Alfaris, 2009) that evolves from abstract ideas to more complex and concrete solutions, attending to the constraints and embodying new design requirements. This process advances and backtracks as many times as necessary in order to achieve a good solution to the design problem.

CAD tools always had an essential role in the design activities, contributing to increase their efficiency, particularly, in the modeling of the design solution. Unfortunately, CAD tools are currently being used just as a more efficient version of the traditional paper-based approach, an approach that does not help the designers at handling change, particularly for the exploration of different solutions or to adapt the design to evolving requirements.

Recently, new approaches have been introduced in the design process, which are better tailored for handling change. Generative Design is one of them and can be defined as the creation of shapes determined by algorithms.

This dissertation argues for one main point: integrating Generative Design as a new stage in the design process dramatically simplifies the handling of changes. In particular, we propose an algorithmic approach to design that overcomes the limitations of the traditional approach for using CAD tools.

Our approach requires the formalization of the design intents, for which we used programming. This approach has an initial cost. However, we argue that this cost is highly rewarding when the designers need to handle change, by allowing them to explore many different alternatives, quickly and effortlessly.

To evaluate the proposed approach we developed a three-dimensional model of a case study, a large and complex building, using a purely programming-based representation.



## 2. APPROACHES TO MODELING: TRADITIONAL vs ALGORITHMIC

To handle change, a designer interacts with a CAD tool requiring different roles and efforts.

In this section we characterize the role of the designer and we evaluate the effort needed to handle change in models developed using two significantly different approaches: the traditional one, based on explicit modeling operations, and a modern one, based on the implementation of algorithms, which is proposed in this thesis.

### *2.1 Traditional Approach*

In the traditional approach, that still governs today's practice, the designer interacts with two or three-dimensional graphic representations that mainly support the production of communication elements such as drawings or renderings. These CAD tools provide operations that allow the designer to easily create, manipulate, edit, and relate the parts of a model but their use is mainly descriptive: the tool is used as a replacement of the paper-based media and conventional physical models.

Using this approach, it is possible to handle complex designs as long as the design logic is clearly understood prior to the modeling tasks. However, it is usually very difficult to handle changes, particularly, in advanced phases of the design process, as they tend to affect many parts of the model, thus requiring too much time and effort to modify all affected parts. In short, although these tools facilitate the production and visualization of complex geometries, they also "(...) limit exploration and effectively restrict design" (Woodburry, 2010: 23).

### *2.2 Algorithmic Approach*

Contrary to the traditional approach, the algorithmic-based approach describes the design as a program written in a formal language (textual, visual, or both). This program can then be executed by a computer, generating a visible model as result. In order to write this program, the designer must know how to decompose the design logic and must select the appropriate programming elements to represent the fundamental parts as well as the relations between these parts.

Using an algorithmic-based approach, the designer interacts with the computational mechanism that generates the digital representation. Obviously, conventional CAD tools also perform computations to generate the digital representation; the difference is that in the traditional approach they are hidden from the user that interacts directly with the CAD.

Algorithmic thinking requires the description of the design through a step-by-step process in which the ideas and intents have to be clearly defined, the relationships between elements are explicit, the requirements prioritized and the control mechanisms developed. This requires an obvious formalization effort.

A designer largely describes designs rather than processes, and usually representations are imprecise, relying on the human capacity to interpret them appropriately, a capability that a computer does not have: one misplaced character means that an algorithm will not work (Woodburry, 2010). Due to the large initial effort needed to formalize design, writing this algorithm might not be a trivial task and it might even be less cost-effective than the traditional approach. However, when it becomes necessary to incorporate changes in the design, this initial effort is quickly recovered. Depending on the structure of the program, a change might be as simple as updating the value of a parameter, or as complex as updating parts of the program. In all cases, however, the modified program is then used to regenerate the model. This is possible because the design is explicitly represented in the program, and changes in the design are translated to changes in the program. The quick model regeneration also allows the exploration of much larger solution spaces, which is advantageous for decision-making.

In spite of all its advantages, there is one drawback in the use of a purely programming-based representation: if manual changes are made to the generated model, these changes will be lost when the program is re-executed and the model regenerated. The most obvious solution to this drawback is simply to forbid any manual changes to the model and always force its implementation in the program. Another possibility is to allow manual changes to the model, e.g., for studying different solutions, that are later incorporated in the program to synchronize it with the model.

### **3. CASE STUDY**

Given that our proposal introduces an additional algorithmic formalization step in the design process, it is important to evaluate if the time spent in this step allows a significant reduction in the time and effort needed for handling changes, particularly in the more advanced phases of the project.

In order to evaluate our proposed approach to design, we made an experiment: we formalized the design of a sufficiently complex design, the Market Hall building, and we executed the modeling process until we obtained a very detailed model of the building.

We outlined a generic modeling strategy that starts with the analysis of the externally imposed constraints and the designers' process, concepts and intentions. Then, we decomposed our modeling process according to two perspectives: (1) to match the design process decomposition in order to progressively increase the detail and definition of the model, and (2) to match the design artefact decomposition, dividing the model into the relevant elements that it represents. Then, we applied this strategy to our case study which we describe below.

The Market Hall is located in a square without adjacent buildings, enabling a context-free modeling without imposing site, shape or dimension constraints. As can be seen in Figure 3.1, the building is characterized by an unconventional shape that creates a large hall, closed by two glass facades.



**Figure 3.1** Photo-realistic image of the Market Hall (source: <http://www.mvrdv.nl>).

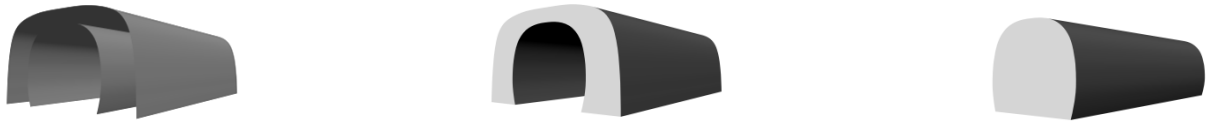
This bent shape was the starting point for the design process, as it constraints both the building elements and the interior spaces.

However, instead of capturing the exact geometry of the building, we captured the underlying ideas of the design, so that it would be possible to generate several different models, including the actual Market Hall.

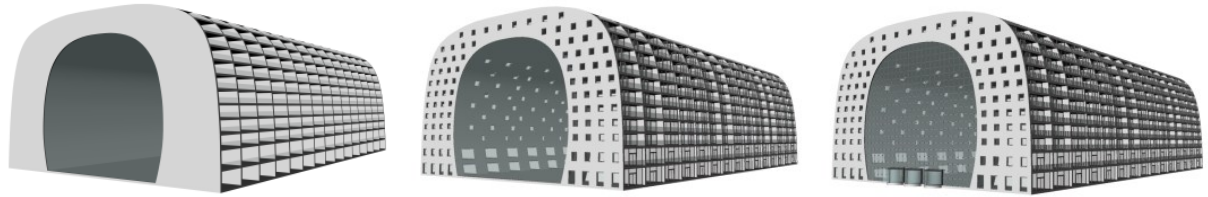
To this end, we formalized and abstracted each design idea, and we implemented it in a programming language. This means that all shapes, relations, repetitions and variations, and rules identified in the Market Hall were formalized. At the same time, all these formalizations were generalized and parameterized to increase its applicability.

When we analyze the building, it becomes obvious that there are strong dependencies between its elements. For example, the cover slab can only be represented after we decide on the overall shape of the building. As another example, we can consider the location of the windows that can only be decided after we know the location of the slabs and vertical walls. Similar dependencies can be established between many other elements. As a result, our design process must establish an order between the modeling of different elements.

The design process we used to construct the model can be divided into four phases (Figure 3.2 and 3.3, page 5): (1) formalization of the overall shape of the building, (2) modeling of the main elements of the building, e.g., walls and slabs, (3) definition of the openings and modeling of the frames, and lastly (4) the detail elements (fencing posts and detail of the glass facades).



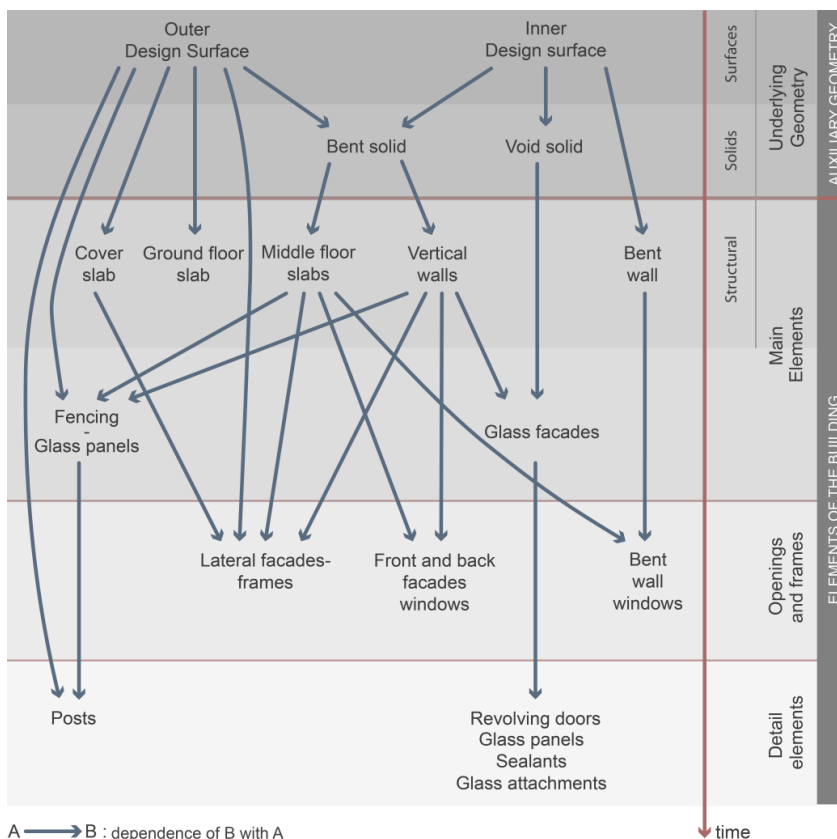
**Figure 3.2** Phase 1 of the model. Formalization of the overall shape of the building: underlying geometry.



**Figure 3.3** Remaining phases of the model. Left: Phase 2. Middle: Phase 3. Right: Phase 4.

In the first phase, we defined the underlying geometry that captures the main concepts of the overall shape of the building. This geometry was then used as input to the remaining phases of the design process.

The decomposition of the process into four phases, each one increasing the detail of the model, led to the existence of dependencies between elements, allowing the automatic propagation of changes. In Figure 3.4, we present the hierarchy of the model according to the four phases of the process, and the dependencies between elements. As can be seen, all the elements of the building depend on its overall shape, which we formalized in the first phase of the modeling process. Additionally, it also shows two decompositions, of the modeling process that matches the one of the design process, in order to progressively increase the detail and definition of the model, and to match the one of the design artifact, decomposing the model into the relevant elements that it represents.



**Figure 3.4** Phases of the design process and hierarchy of the model.

## 4. EVALUATION

To evaluate the ability of the algorithmic approach to respond to change and growth in the requirements, we simulated several types of changes and measured their impact, particularly, in the time and effort required for their implementation compared to the traditional approach.

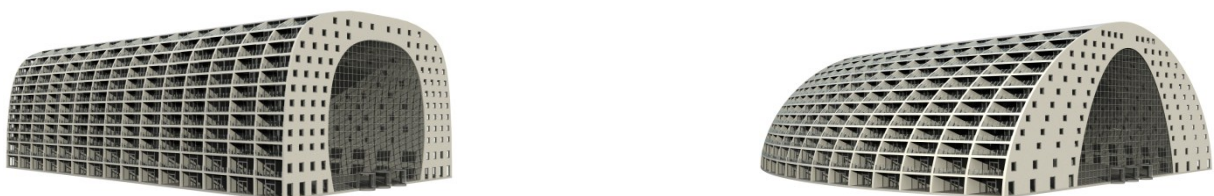
We tested five main scenarios. Starting from the complete Market Hall, we simulated the introduction of different changes, namely: (1) changing just the overall shape of the building, (2) changing just the dimensions or positions of the elements of the building, (3) simultaneously changing the overall shape of the building and the dimensions or positions of its elements, (4) changing the shape of some elements of the building, and (5) changing the order of elements of the building. Lastly (6), we also simulated the reuse of the program in other projects by changing the formalization of the overall shape of the building.

### **4.1 Scenario 1: Changing just the overall shape of the building**

A change in the shape of the building causes many different changes in its elements, namely in their geometry, position, direction, and number of subelements.

The effect of this change is clearly visible in Figure 4.1 (on the right) when comparing with the Market Hall (Figure 4.1, on the left), where a different building is generated by changing the overall shape of the building and its section along the longitudinal axis. These results were produced by changing four arguments, which we made in minutes.

This is the kind of change that is very difficult to accommodate using a traditional approach to modeling because the majority of the elements would have to be remade or their positions and directions recalculated. This process would certainly be very time-consuming and laborious and, for each change in the shape of the building, the effort would have to be repeated.



**Figure 4.1** On the left: MVRDV's Market Hall as generated from our program. On the right: A different building generated by changing the overall shape of the building.

### **4.2 Scenario 2: Maintaining the shape of the building and changing the dimensions or positions of the elements**

Another possible change may require a change in the dimensions or positions of certain elements. To give a concrete example, any change in the position of the middle slabs would have repercussions in

several other elements, e.g., dimension and position of the frames of the lateral facades, geometry and position of the fencing panels, etc.

The effect of this change is visible in Figure 4.2 (on the right). That building was generated by simply changing the parameter containing the z-coordinates of the middle floors, a change that was implemented in minutes. Observing that figure, and comparing with the Market Hall (Figure 4.2, on the left), it is clear that this change do not have much visual impact, however several elements of the building are affected by that change, however small it may be.



**Figure 4.2** On the left: Market Hall as generated from our program. On the right: Building generated after changing the positions and dimensions of the elements.

Considering a traditional approach the designer would have to remake the middle slabs and many other elements that need to be adjusted.

Given this example, we concluded that a change that might even seem simple to solve using a traditional approach, would continue to require an enormous effort to modify all affected elements, in fact, similar to the effort needed to produce the elements in the first place.

#### ***4.3 Scenario 3: Changing the shape of the building, its dimension and the positions and dimensions of its elements***

Another effect of the introduction of a change in the overall shape of the building is visible in Figure 4.3. In these cases, different buildings were generated by more drastically changing the section curves and by using different sections along the longitudinal axis. Besides the change of the shape of the building, other parameters were changed, e.g., the number and position of the floors and walls, the number of windows of the front and back facades, the number and positions of the revolving doors, and the dimensions of elements such as the windows or glass panels. The relationships between elements were kept and adapted to a different scale.



**Figure 4.3** Buildings generated by introducing several changes: variant 1 and 2.

Similarly to the first and second scenarios, these results were produced by changing a small set of parameters, a task that was made in minutes. It should be clear that these changes are much more

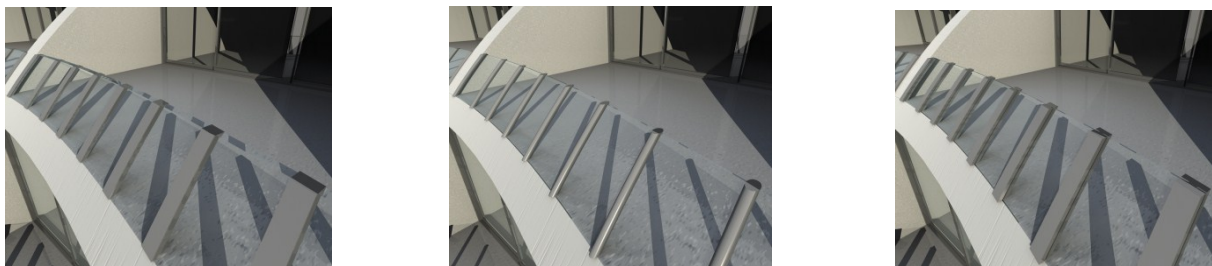
difficult to accommodate using a traditional approach than what it already was in the first two scenarios as they are, in fact, a combination of those two sets of changes.

Using a traditional approach, it would not be possible to reuse any element from one model to the next. The number of changes would be so large that, in fact, it would probably be easier to just start building the model from scratch, thus wasting the effort already spent in the previous model.

#### **4.4 Scenario 4: Changing the shape of the elements of the building**

In this scenario we evaluate more drastic changes that imply modifications not only in the parameters but also in the developed algorithms. We simulated this change for two different elements, namely the windows of the front and back facades, and the fencing posts.

In both cases the underlying logic of the modeling process continued valid and the modification of both procedures was similar. We generalized the procedures dealing with specific shapes to accept an additional parameter for a procedure that produces the desired shape. To give a concrete example, in Figure 4.4, we illustrate the results of the generalization we made for the fencing posts. In this case we abstracted the section of the posts (initially rectangular to be identical to those of the Market Hall) for allowing any other section which we illustrate for a circular section and an I-beam profile. The modifications we made to our script were completed in less than an hour, which is a very small amount of time for a task that, in the traditional approach, requires changes to many elements.



**Figure 4.4** Changing the section of the posts. Left: rectangular. Middle: circular. Right: I-beam profile.

#### **4.5 Scenario 5: Changing the order of elements of the building**

The Market Hall is characterized by frames in the lateral facades that follow a sequence on each floor. In this scenario we evaluate the costs of changing the order of the elements, without changing anything else.

To change the order of types of frames we just changed a sequence of procedures, arguments in other procedures. This is visible in Figure 4.5 and was made in seconds.



**Figure 4.5** Changing the order of the frames of the lateral frames. Left: Frames following the sequence of the Market Hall. Right: Frames with a different order.

Neither the shape of the frames nor their position was modified. Thus, in a traditional approach, the designer could reuse the previous elements and use “cut” “copy” and “paste” to rearrange their order, which would solve the problem easily. However, in buildings with so many elements as the Market Hall, this task is certainly laborious, time consuming, and repetitive, which makes it very error-prone.

#### **4.6 Scenario 6: Changing the formalization of the shape of the building**

The more drastic experiment was made by simulating a change in the formalization of the building from a bent and longitudinal shape to a parallelepiped shape. In Figure 4.6 it is possible to observe the effect of this change.

During the modeling process and, particularly, in the review phase, we considered the use of the script for future design problems. Thus, we abstracted the procedures to receive abstract entities, e.g. solids, surfaces or curves, and just the last procedure that produces one element was particularized to receive the parameters that were used to formalize the overall shape of the Market Hall, that is to be applied to the underlying geometry that was defined in the first phase of the model. To obtain these results it was just required to define a set of procedures, the underlying geometry, to formalize the new shape. This auxiliary geometry was used as input in the procedures already defined, to which we did not made any change. The definition of this new set of procedures was made in approximately half an hour.

If an equivalent reuse is intended in a traditional approach to modeling, the designer might use some elements such as the frames of the lateral facades of other models when their overall dimensions are identical, but, in general, he would have to start the model from scratch.



**Figure 4.6** Change in the formalization of the building: two parallelepiped shapes.

## **5. CONCLUSIONS**

Design processes are characterized by change. Unfortunately, CAD tools are currently being used just as a more efficient version of the traditional paper-based approach, an approach that does not help the designers in handling change, particularly for the exploration of different solutions or to adapt the design to evolving requirements.



Recently, new approaches have been introduced in the design process, which are better tailored for handling change. Generative Design is one of them and can be defined as the creation of shapes determined by algorithms.

This dissertation argues for one main point: integrating Generative Design techniques as a new stage into the design process dramatically simplifies the handling of changes. In particular, we propose an algorithmic approach to design that overcomes the limitations of the traditional approach.

This proposal introduces a formalization step using a programming-based representation of the intended design. We claim that this step speeds up the handling of changes that always occur as the design evolves. To evaluate our proposal, we formalized the design of a building, the MVRDV's Market Hall.

To evaluate our algorithmic formalization, we simulated several types of changes and measured their impact, particularly, in the time and effort required for their implementation compared with the traditional use of CAD tools. These simulations allowed us to verify the relevance of using an algorithmic approach to design starting from the early stages of the design process, and also to explore the limits of the programming-based representation.

With this evaluation we proved that our approach is sufficiently flexible, not only to accommodate changes previously anticipated, but also changes that were not planned. This conclusion can be extended to the architectural practice in which changes arise, frequently without being anticipated. Although the additional step of design formalization requires an obvious initial investment, we believe that it is not only possible but actually cost-effective to use an algorithmic approach to design as a new stage in the design process. In fact, the initial cost, namely in time and effort, is quickly recovered when changes are needed.

Our generalization of the Market Hall design allowed us to also produce buildings that are different from the Market Hall. This was made possible due to the automatic propagation of changes between the elements of the building. Thus, an algorithmic approach to design proved to be aligned with the design process' needs by allowing not only an effortless introduction of changes but also the production of an infinite number of different models exploring different designs solutions, and, thus truly assisting designers in decision-making activities.

Our experiment proved also that this approach can be integrated right after the development of the concept that structures the design. This early integration allows designers to work with a powerful tool along the design process, from conceptual design until construction. In fact, our evaluation showed that an algorithmic approach to design allows mass-customization strategies, in which the effort required to produce one program is recovered by its use to model several similar buildings.

The development of our model allowed us to provide lessons to the development of similar models. For dramatically reducing the effort for handling change in the design process, we can highlight the importance of the definition of the parameters and the strategy we used to construct the model. This

strategy can be summarized in four main steps: (1) to analyse the design and formalize its intentions, (2) to abstract and generalize these intentions, predicting the changes that can arise in the design process, (3) to decompose the definition of the elements in subtasks, and (4) to define these subtasks so that they operate with abstract entities. The abstraction and decomposition are of great importance for incorporating changes that were not previously anticipated.

In summary, an algorithmic approach can be integrated into the design process to make it more cost-effective at handling changes, giving the designer a competitive advantage. Our proposal does not exclude other approaches for the design process. It is an additional stage that does not replace the creative work of the designer. Instead, it allows him to go farther in the exploration of different design solutions.

In the beginning of this work we proposed to identify the limits of the programming-based representation. Given the level of detail that was implemented in our models, we can now safely conclude that it is possible to go very far indeed.

## 6. BIBLIOGRAPHY

Alfaris, A. (2009) "Emergence through conflict: the Multi-Disciplinary Design System (MDDS)", PhD thesis, Massachusetts Institute of Technology.

El-Khaldi, M. (2007) "Mapping boundaries of generative systems for design synthesis", SM thesis, Massachusetts Institute of Technology.

Gero, J. (1990) "Design Prototypes: A Knowledge Representation Schema for Design", *AI Magazine*, 11, n°4, pp. 26-36.

Hudson, R. (2010) "Strategies for parametric design in architecture: an application of practice led research", PhD thesis, University of Bath.

Simon, H. (1973) "The Structure of Ill Structured Problems", *Artificial Intelligence*, 4, n°3, pp. 181-201.

Woodburry, R. (2010) *Elements of Parametric Design*, New York: Routledge.