

# **BIM DINÂMICO. UMA FERRAMENTA DE SIMULAÇÃO VISUAL DE MOVIMENTO PARA PROJETO DE FACHADA CINÉTICA EM PLATAFORMA .NET.**

**José Luis Menegotto**

Departamento de Expressão Gráfica, Escola Politécnica,  
Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil, e-mail: [jlmenegotto@poli.ufrj.br](mailto:jlmenegotto@poli.ufrj.br)

## **Resumo**

*O presente artigo relata a experiência de criação de um aplicativo de apoio ao projeto de fachadas cinéticas, implementado em linguagem C# em plataforma .NET e testado em ambiente gráfico Revit. O aplicativo tem como objetivo assistir os projetistas durante a concepção do projeto, especificamente nas etapas de simulação visual dos padrões de movimento que serão posteriormente transferidos aos componentes dinâmicos concretos do edifício. Neste caso, busca-se controlar esteticamente a variação dos vidros polarizados aplicados sobre as fachadas. O estudo apresenta uma técnica de criação de tramas geométricas a partir de séries numéricas controladas por operações de aritmética modular. O algoritmo programado permite obter tramas seriais de padrões geométricos controlados e variáveis. A pesquisa se dirige a formalizar uma biblioteca de padrões e tipos de movimentos possíveis, através de classificações e de uma representação simbólica abstrata que leva em conta a forma geométrica estática dos elementos associando-as às formas dinâmicas derivadas dos padrões de movimento programados.*

**Palavras-chave:** BIM, Revit, Fachadas cinéticas, API .Net.

## **Abstract**

*This article describes the experience of creating a support application for designing kinetic facades. The application was programmed and tested in Revit and implemented by algorithms in NET platform written in C#. The goal of this application is to assist architects during the design phase of the project; more specifically during the visual simulation of movement patterns that will be later transferred to actual dynamic components of the building. In this case, we try to control the variation of polarized glasses used on architectural facades. The study presents a modeling technique of geometrical patterns created and controlled by modular arithmetic operations. The programmed algorithm allows us to obtain serial frames of controlled and variable geometric patterns. The research aimed to formalize a library of patterns and types of possible movements through classification and also through an abstract symbolic representation that takes into account the static geometric shape of the elements linking them to dynamic forms derived from the programmed movement patterns.*

**Keywords:** BIM, Revit, kinetics facades, API .Net.

## 1. INTRODUÇÃO

### 1.1. Antecedentes.

A arquitetura contemporânea vem incorporando dispositivos mecânicos dinâmicos integrados aos elementos construtivos das edificações. O faz embasada em pesquisas e experimentos teóricos que começaram a surgir a partir da segunda metade do século XX. Como consequência dessa incorporação, a estética arquitetônica vê-se diante do problema de controlar e aprimorar a variável “movimento” dentre as suas preocupações de projeto.

Iniciadas entre as décadas de 1960 e 1970, pesquisas integradas a essa área de conhecimento vêm ganhando força, dando os seus primeiros resultados nas últimas duas décadas. Tanto a arquitetura produzida com comportamento dinâmico como a pesquisa associada nessa direção, denominam-se genericamente Arquitetura Cinética (*Kinetic Architecture*) e Arquitetura Responsiva (*Responsive Architecture*). Em ambos os casos está implícita a ideia da arquitetura como uma arte performática.

Este artigo volta a sua atenção para a experiência de criação de um aplicativo de apoio ao projeto de fachadas cinéticas programado e testado em ambiente gráfico Revit e implementado em plataforma .NET em linguagem C#. O aplicativo tem como objetivo assistir os projetistas durante a etapa de concepção do projeto, especificamente durante as etapas de simulação visual de padrões de movimento que serão posteriormente transferidos aos componentes dinâmicos concretos da edificação. Neste caso, busca-se controlar esteticamente a variação de vidros polarizados aplicados sobre fachadas. A tecnologia de vidros polarizados permite mudar a transparência do vidro pela aplicação de corrente elétrica. A passagem ou a obstrução da corrente pelo vidro, tem como efeito permitir que as moléculas do material se alinhem ou desalinhem, conferindo-lhe o aspecto de transparência ou opacidade. Partindo dessa possibilidade, a programação do aplicativo procura estabelecer uma forma de estudar a configuração de padrões de passagem ou obstrução da corrente elétrica para produzir o efeito visual. O estudo apresenta uma técnica de modelagem de tramas geométricas criadas a partir de séries numéricas controladas mediante operações de aritmética modular. O algoritmo programado permite obter tramas seriais de padrões geométricos controlados, tanto regulares como variáveis.

## 2. DESENVOLVIMENTO.

### 2.1. Variação sobre a trama de múltiplos.

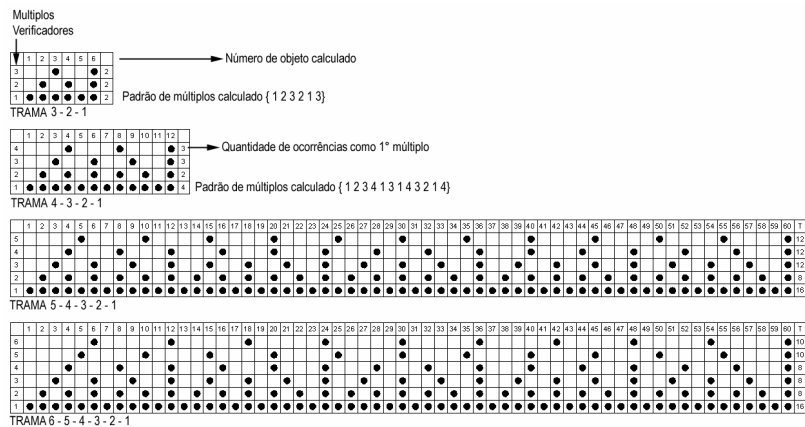
O método utilizado para projetar os estados dos vidros (ON-OFF) que outorguem à fachada uma característica dinâmica que permita defini-la como “cinética” se baseia na manipulação numérica de múltiplos. Podemos representar esse tipo de transformação por uma matriz de filas e colunas numeradas sequencialmente. Os números na linha horizontal da matriz representam a posição ordinal do elemento construtivo processado em determinado momento. A primeira coluna da esquerda da matriz, numerada de baixo para cima em sentido crescente, são os números denominados “múltiplos verificadores”. Cada uma das células da trama foi preenchida com pontos que representam uma congruência de múltiplo para o objeto processado.

Lendo cada coluna em sentido ascendente é colocado um ponto cada vez que o múltiplo verificador seja múltiplo do ordinal definido na linha horizontal da matriz. Assim, por exemplo, na primeira trama da Figura 1 o terceiro objeto processado terá dois múltiplos 3 e 1, já, na segunda trama, o número 4 que representaria o quarto objeto processado, possui como

múltiplo o 4, 2 e 1. O primeiro múltiplo a aparecer, começando a contar por cima representa o índice do ponto que acionaria o comando liga-desliga da corrente elétrica. Na trama de múltiplos de 1 a 3, do primeiro exemplo apresentado na figura, a ordem da sequência de alteração de estados seria {1 2 3 2 1 3}. Já, se fosse colocada a condição adicional, como ser múltiplo de 4 (segunda trama), a ordem seria {1 2 3 4 1 3 1 4 3 2 1 4}.

Observe-se que um período é fechado quando o número do objeto processado cumpre a condição de ser o Mínimo Múltiplo Comum de todas as linhas definidas na matriz, quer dizer, de todos os múltiplos verificadores, a partir do qual o ciclo se repete. Na trama de múltiplos de três, por exemplo, verifica-se que os múltiplos por ordem de aparição em sentido decrescente completam um período na sexto objeto processado, recomeçando o padrão geométrico no sétimo objeto.

Figura 1 – Tramas de múltiplos.



Fonte: (Menegotto, 2009).

Na medida em que a quantidade de múltiplos verificadores aumenta (coluna numérica da esquerda) o padrão geométrico que emerge da trama se torna mais longo, adquirindo uma característica visual mais rica e mais difusa. Observa-se também que as tramas contêm padrões intermediários dentro de cada ciclo ou período. Examinando com atenção podem ser distinguidos. Cada período possui no seu ponto médio um eixo de simetria axial. Na figura 2 mostram-se as distâncias relativas entre as tramas formadas pelos múltiplos de 3, 4, 5, 6, e 7. A trama 7 destaca-se das outras pela sua extensão. As tramas correspondentes a 5 e 6 apresentam a mesma longitude, embora os padrões gráficos sejam levemente diferentes pela adição de mais uma linha horizontal. Essa característica pode ser utilizada para distribuir equilibradamente variações de movimentos com a mesma periodicidade, mas com pequenas diferenças de percurso.

Figura 2 – Comparação das longitudes das tramas de múltiplos.

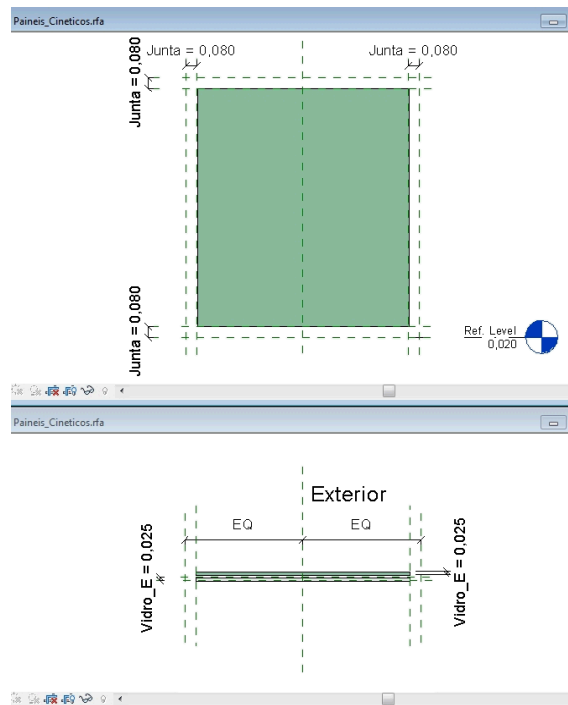


Fonte: (Menegotto, 2009).

Se for reduzido o número de linhas, limitando a escolha de múltiplos verificadores aos números pares dentro do domínio {1 2 3 4 5 6 7 8}, observa-se que a trama formada apresenta a seguinte característica:



Figura 4 – Família de Painel\_cinético.



Fonte: (o autor).

## 1.2. O algoritmo codificado em C# em ambiente .NET

A seguir se apresenta o código fonte da seção principal da função. Nas linhas 1 a 9 são preparados os objetos de interface e criados os filtros de seleção e as listas com os objetos que serão processados. Neste caso foram usados objetos da categoria “*CurtainWallPanels*”. Na linha 10 se apresenta uma lista com a série numérica de múltiplos verificadores com a qual se procede a verificar se o número correspondente da posição ordinal que ocupa o painel no *curtain-wall* é múltiplo desse verificador. Cada número dessa lista corresponde a um ciclo e a um estado da fachada. Na linha 32 é setada a variável “ruído” que estabelece um fator de variabilidade nas tramas modulares regulares modificando o valor do múltiplo verificador calculado quando o valor ordinal do painel é múltiplo desse fator.

Quadro 1 – Código.

```

1. public class FachadaCinetica : IExternalCommand
2. { public Result Execute(ExternalCommandData commandData, ref string message, ElementSet elements)
3. { UIApplication uiApp = commandData.Application;
3. Document doc = uiApp.ActiveUIDocument.Document;
4. UIDocument uidoc = uiApp.ActiveUIDocument;
5. ElementCategoryFilter filtro = new ElementCategoryFilter(BuiltInCategory.OST_CurtainWallPanels);
6. FilteredElementCollector colecao = new FilteredElementCollector(doc);
7. ICollection<ElementId> Paneis = colecao.WherePasses(filtro).WhereElementIsNotElementType().ToElementIds();
8. IList<Element> L_Paneis = colecao.WherePasses(filtro).WhereElementIsNotElementType().ToList();
9.
10. List<int> Matriz = new List<int>(new int[] {1,2,3,4,5,4,3,2,1,7,9,7,5,6,5,6,9,1,2,4,8,10,5,10,8,4,8} );
11. using (TransactionGroup group = new TransactionGroup(doc))
12. { group.Start("Estado fachada");
13. for (int ind = 0; ind < Matriz.Count; ind++)
14. { int modulo = Matriz.ElementAt(ind);

```

```

15.         string nometran = "Ciclo" + ind.ToString();
16.         EstadoFachada(doc, uidoc, Paneis, nometran, modulo);
17.     }
18.     group.Assimilate();}
19.     return Result.Succeeded;
20. }
21. private void EstadoFachada(Document doc, UIDocument uidoc, ICollection<ElementId> Paneis, string cic, int mod)
22.     {using (Transaction tr = new Transaction(doc))
23.         {tr.Start(cic);
24.           Calcula_Padrao(doc, uidoc, Paneis, mod);
25.           tr.Commit();}
26.         uidoc.RefreshActiveView();
27.         Thread.Sleep(500);
28.     }
29. public Result Calcula_Padrao(Document doc, UIDocument uidoc, ICollection<ElementId> Paneis, int mod)
30.     {View vista = doc.ActiveView;
31.     int modul = mod;
32.     int ruido = 5;
33.     for (int ind = 0; ind < Totais; ind++)
34.         {ElementId Panel_id = Paneis.ElementAt(ind);
35.         FamilyInstance panel = doc.get_Element(Panel_id) as FamilyInstance;
36.         if (Modulo(ind, ruido)) modul = ruido; else modul = mod;
37.         double abre = (0.01 * modul);
38.         Parameter junta = panel.get_Parameter("Junta");
39.         Parameter ligar = panel.get_Parameter("Vidro_Ligado");
40.         Parameter desli = panel.get_Parameter("Vidro_Desligado");
41.         junta.Set(Modulo(ind, modul) ? 0.25 - abre : abre);
42.         ligar.Set(Modulo(ind, modul) ? 1 : 0);
43.         desli.Set(Modulo(ind, modul) ? 0 : 1);
44.     }
45.     return Result.Succeeded;
46. }
47. public bool Modulo(int num, int mod) { return num % mod == 0; } // Verificação modular do painel.
48. }

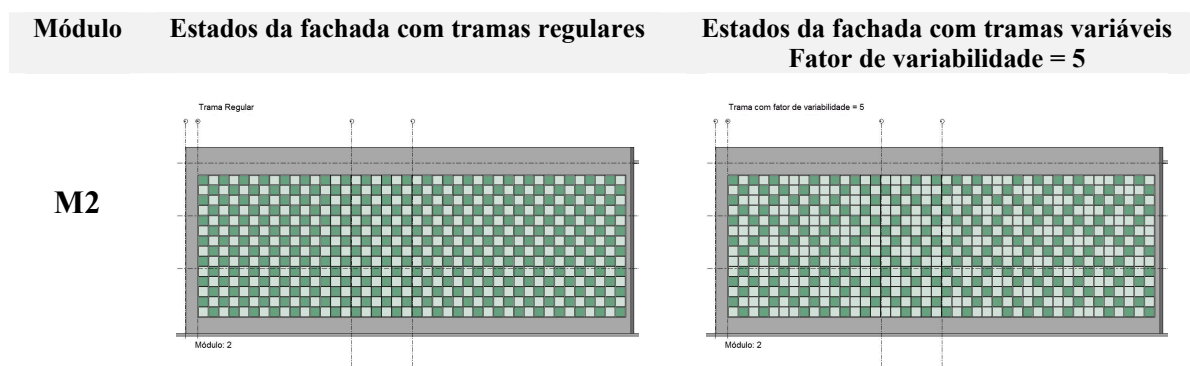
```

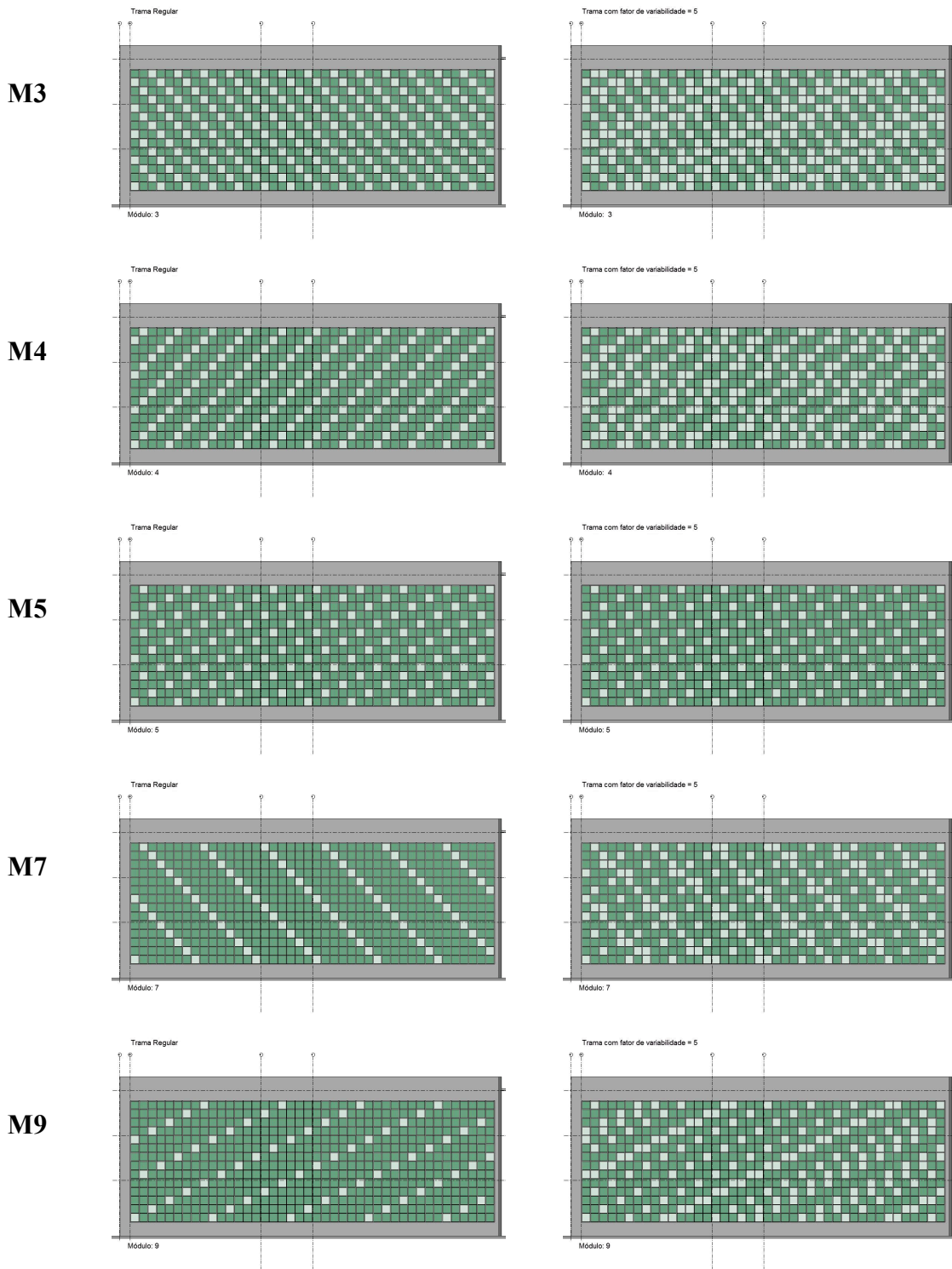
Fonte: (o autor).

## 2. RESULTADOS.

Nesta seção se apresentam algumas imagens extraídas das sequências de estados da fachada. No quadro 2 pode-se diferenciar dois tipos de tramas: as regulares e as variadas. As primeiras, correspondentes à coluna da esquerda do quadro, foram obtidas verificando apenas os múltiplos fornecidos em forma de lista. Já as segundas incorporam o fator de variabilidade (variável ruído) que confere à trama maior nível de movimento visual.

Quadro 2 – Resultados.





Fonte: (o autor).

### 3. CONCLUSÕES.

A aritmética modular mostrou ser uma ferramenta matemática versátil para obter as tramas de movimento com características de aleatoriedade assim como tramas que se apresentam perfeitamente simétricas. O presente estudo de fachadas cinéticas em ambiente Revit, mostrou que é possível obter simulações dinâmicas para aprimorar esteticamente projetos de arquitetura que pretendam apresentar variação ou transformação dos seus elementos ao longo

do tempo. As tramas obtidas mostram um potencial promissor para o desenvolvimento de sistemas de controle de movimento de padrões geométricos integrados com as etapas de concepção do projeto. Em outras palavras, o controle estético das tramas de movimento tornou-se um aspecto que pode ser controlado intencionalmente pelo projetista dentro de um aplicativo específico articulado à tecnologia BIM.

A pesquisa encaminha-se para a formalização de bibliotecas de padrões e tipos de movimentos possíveis, através de representações simbólicas de nível mais abstrato que levem em conta tanto a forma geométrica estática dos elementos construtivos quanto às formas dinâmicas derivadas dos padrões de movimento programados, associando-as.

Como contribuição para o campo da arquitetura, a pesquisa intenciona continuar a experimentar o controle dinâmico sobre outros tipos de elementos arquitetônicos em fachadas (brises, esquadrias, fachadas multicamadas), visando inter-relacionar parâmetros ambientais de eficiência energética, estes de natureza objetiva, com os subjetivos parâmetros estéticos vinculados com o livre jogo das distribuições geométricas desses componentes.

## REFERÊNCIAS

AUTODESK, INC. *Revit 2011. API Developer's guide*. Autodesk, 2010.

COATES, PAUL. *Programming Architecture*. Routledge, 2010.

MENEGOTTO, JOSÉ LUIS. A caixa de música. Contraponto formal entre a arquitetura e a música. Tese. Universidade Federal do Rio de Janeiro, Faculdade de Arquitetura e Urbanismo, Rio de Janeiro: UFRJ/ FAU, 2009.

MOLONEY, JULES. *Designing kinetics for Architectural Facades: State Change*. Routledge. London, New York, 2009.

XENAKIS, IANNIS. *Formalized Music. Thought and Mathematics in Composition*. New York: Pendragon Press, 1992.

## AGRADECIMENTOS

Agradecimentos a Autodesk ADN, Autodesk Student Program e a Jeremy Tammik pelo suporte durante a criação da aplicação.