Grammar based design tools: Issues of representation and interaction

Scott Chase
Architecture, Design & Media Technology
Aalborg University

scha@create.aau.dk
http://homes.create.aau.dk/scha/

Designing With Vision Workshop
The Open University, 8 June 2010
http://design.open.ac.uk/DV/
Challenge

We want conceptual design tools that support designers’ ways of thinking and working and enhance creativity, e.g. offering design alternatives difficult or not possible without the use of such tools.
Shape grammars

Shape grammar for greek cross church plans

Initial shape

Rule

Derivation

Designs
Grammar applications

- Analysis
  - Grammar construction from a corpus of designs
- Synthesis
  - New grammars of design styles
- Transformation
  - From one design style to another
  - Change of design brief
  - Hybrid, crossover designs
Emergence

A: 4 maximal lines
S: 8 maximal lines

τ(A) ≤ S
Unexpected results

Rule

\[ \square \rightarrow \Diamond \]

Initial shape

Derivation
Emergent forms

Rule R

Initial shape S

derivation

emergent forms
Grammar computational complexity

Darker areas indicate more computational complexity

Other factors:
• Symbolic/semantic information
• Composite representations

String grammars
Set grammars (graph grammars)
Shape grammars
Parametric shape grammars (shape schema grammars)
Grammar use & interaction

Grammar system development

Grammar definition
Interpretive mechanism

Grammar application (derivation)

Determination of rule
Determination of object
Determination of matching condition

Design evaluation
Grammar evaluation
Grammar transformation

Scenario 1
Designer: Full control

Scenario 2
Developer
Designer: Full control over grammar application

Scenario 3
Developer
Designer: Rule and object selection
Computer control

Scenario 4
Developer
Designer: Rule selection only
Computer control

Scenario 5
Developer
Computer control

Scenario 6
Computer: Full control
Issues for computer implementation

• Maximal element representations create huge computational issues (e.g. combinatorial explosion)

• Designer interaction with such systems is a challenge
  • e.g. how to present all possibilities to the designer
Common implementation restrictions

• ‘Toy’ systems, e.g. proof of concept
• Single design application, hard coded
• Representation restrictions, e.g. set grammars, raster representations
• Design restrictions, e.g. orthogonal designs only

Do these restrictions keep us from moving forward?
/* "shapegrammar.pro" Consulted. */
  yes
?- initialize.
  yes
?- start.
  initial shape file: initial3.dbda
  rules file: rules5.dbba
/* Consulting "initial3.dbda" */
/* "initial3.dbda" Consulted. */
/* Consulting "rules5.dbba" */
/* "rules5.dbba" Consulted. */
  yes
?- redraw.
  yes
?- go.
Enter rule #: 5
Enter rule point 1: Dig = (24,48)  Found
Enter rule point 2: Dig = (48,48)  Found
Enter rule point 3: Dig = (49,24)  Found
Pick shape point 1: Dig = (101,150) Found
Pick shape point 2: Dig = (200,150) Found
Pick shape point 3:
GEdit
EifForm

Planar truss grammar

Dome

Canopy/landscape
DSSG project overview
... we anticipate three intertwined cycles

http://www.engineering.leeds.ac.uk/dssg/
Classification of shape rules

participant 1

bend

substitute

change angles

add

change length/width

cut

substitute
Recent grammar implementations

To be demonstrated at DCC 2010, 11 July

1. Grammar development environment
2. General interpreter for rectilinear forms
3. 3D parameterised primitives
4. Curved shapes
5. Machining planning
6. Subshape detection w/computer vision
7. Mass customised housing
An outline is not available.

**Property** | **Value**
---|---
ID | 2
Name | AddSquare
Rule Type | Addition
So, while there is ongoing research activity looking at issues of representation, interaction and use, how close are we to having truly useful grammar based design aids?