한국 CAD/CAM학회 논문집
제15권 제6호 2010년 12월 pp. 440-448

# 도형문법 기반의 생성디자인 활용에 관한 연구

김언용*, 전한종**

# A Study on Using Generative Design based on Shape Grammars

Eonyong Kim* and Hanjong Jun**

**ABSTRACT**

At the early design stage, a designer would develop a design through reviewing various alternatives. However, the limitation of human information handling capacity restricts what makes various alternatives. A way to overcome the limitation is to create the alternatives with computer power and the generative design concept with computer has been used. For the generative design, even if a great deal of methodologies has been presented, the core of methodologies would be the process of shapes which examine interrelationship among shapes and how shape can be changed with the interrelationship. The interrelationship can be represented as rules. To represent the rule, the shape grammar is suggested by Stiny in 1972 with the article of "Shape Grammars and the Generative Specification of Painting and Sculpture". The aim of this paper is to present a way of using generative design based on shape grammar for generating alternatives in 3D manner.

***Key words*** : Generative Design, Shape Grammars, Rule based Design

## 1. 서 론

초기디자인 단계에서, 디자이너는 다양한 대안(Alternatives)의 검토를 통해 디자인을 발전시킬 수 있을 것이다. 하지만 인간의 정보처리 능력의 한계는 다양한 대안을 만들어 내는 것을 구속하고 있다. 이러한 한계를 극복하는 방법중의 하나로, 컴퓨터를 이용한 대안의 생성이 가능할 것이며 생성디자인(Generative Design)이라는 개념이 이를 위하여 사용되고 있다. 현재까지 생성디자인을 위한 다양한 방법론들이 제시되고 보고되고 있으나, 가장 핵심적인 하부요소는 도형(Shape)의 처리라고 볼 수 있으며, 또 다른 요소로써 각각의 도형문법의 변형과 상호연관 관계에 대한 규명이라고 볼 수 있다. 또한 이러한 상호연관 관계들은 규칙(Rule)으로 표현될 수 있다. 이를 위하여 Stiny는 1972년에 "Shape Grammars and the Generative Specification of Painting and Sculpture" (Stiny and Gips 1972)라는 논문을 통해서 이러한 규칙을 표현할 수 있는 도형문법(Shape Grammar)를 제안하였고 이 연구에서 도형기반의 디자인 생성을 제시하였다. 또한 도형문법은 컴퓨터를 기반으로 하는 디자인에서 컴퓨터가 디자인을 인지하고 계산할 수 있게 하는 방법으로 주목받고 있다.

그러나 기존의 도형문법 관련 연구들은 주로 디자인 보다는 기존의 디자인 결과물을 분석하는 용도로 연구 되어 왔으며, 그 대상 객체가 주로 2차원 도형이나 3차원 정형 메스의 수준에 머물러 있는 채로 존재해왔다. 이는 도형문법이 실제적인 디자인에 사용되지 못하게 하는 현상을 초래했다고 볼 수 있다. 이러한 문제는 도형문법에 한정을 둔 용어상의 문제로 볼 수도 있으므로 도형문법을 확장한 디자인 문법(Design Grammar)라는 개념(Chase 2002)도 제안되고 있다. 본 연구에서는 3차원 비정형에서의 도형문법의 적용 가능성과, 이를 바탕으로 한 실험을 통해 도형문법 기반의 생성디자인 방법을 고찰한다.

## 2. 생성 디자인

생성시스템을 기본으로 하는 생성 디자인이 디자인

*학생회원, 한양대학교 공과대학 건축학부 건축환경공학과
**교신저자, 정회원, 한양대학교 공과대학 건축학부
- 논문투고일: 2010. 04. 15
- 논문수정일: 2010. 10. 19
- 심사완료일: 2010. 11. 09

프로세스와 프로세스를 표출하는 기존의 패러다임을 변경할 수 있다는 주장은(Khun 1996) 생성시스템이 역동적인 프로세스와 이를 통한 결과물의 관점으로 세계를 보는 철학과 더불어 이를 위한 방법론을 제공하기 때문에 가능한 것이다. 생성 시스템은 디자인 과정중의 설계 대안에 대한 디자이너의 추론 과정에 개입을 하여 도움을 줄 수 있고, 이를 처리하는 행위에도 개입을 하여 도움을 줄 수 있다. 또한 각각의 디자인 요소, 프로세스, 시스템 등의 상호작용을 통해 생성 시스템이 생성해 내는 결과물을 통하여 디자인 개념화의 과정에 도움을 줄 수 있다. 이 생성과정은 디자이너가 규정하는 독특한 속성에 기반을 두고 있으며, 생성 시스템은 디자이너가 자신의 설계 개념의 출발점이 되는 초기의 디자인 형태를 기반으로 여러 가지 주변 상황을 기반으로한 다양한 대안을 생성하여 디자이너의 디자인 행위에 도움을 준다.

생성 디자인은 기존의 디자인 개념화 작업 방법을 탈피한 새롭고 혁신적인 방법을 제시하고 있으며, 이러한 연구들은 일반적으로 자연계에서 볼 수 있는 합성(Synthesis)이라는 개념에 기반을 두고 있다고 볼 수 있다(McCormack 2004).

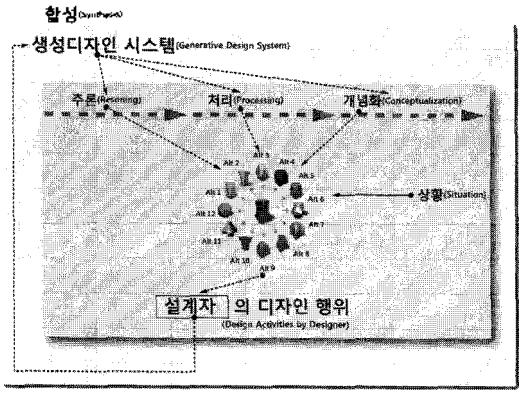



**Fig. 1.** 생성디자인 프로세스.

Fig. 1은 이러한 생성디자인의 프로세스를 도식화하여 보여주고 있으며, 디자이너는 자신의 디자인 의도가 반영된 생성디자인 시스템이 생성한 디자인 대안을 검토하고 주변상황, 즉 디자인 발주자의 요구상황 등과 비교하여 대안을 선택하며 필요 시에는 생성디자인 시스템의 설정을 변경하여 새로운 대안이 생성되게 하기도 한다. 생성 디자인 시스템은 다음과 속성을 가지고 있다(McCormack, 2004).

1. 디자인의 복합성을 생성: 주어진 요소들과의 상호관계를 이용하여, 가능한 조합들을 생성할 수 있는 능력을 의미한다. 즉, 데이터의 조합을 의미한다.
2. 디자인 환경과 디자인 요소들 간의 상호연결 관계와 그 복합성의 표현: 디자인 요소들과의 거미줄과 같은 상호연결관계의 표현을 통하여, 디자인 히스토리를 처리한다. 즉, 순환적인 디자인 피드백을 통한 디자인 과정에 기여를 할 수 있어야 한다는 의미이다.
3. 자가 관리 및 복구: 인간이 디자인 하는 구조는 물리적 기능적 관점에서 일반적으로 다루기가 어려운 경향을 가지고 있다. 즉 이러한 부분을 해결하기 위한 능력을 가지고 있어야 한다. 즉, 새로운 디자인 개념의 추가, 변경의 요구 등 디자인 환경 변화에 적응할 수 있어야 한다.
4. 창의적인 구조, 속성, 결과물 또는 관계를 생성: 새롭고 창의적인 결과물을 생산할 수 있어야 한다.

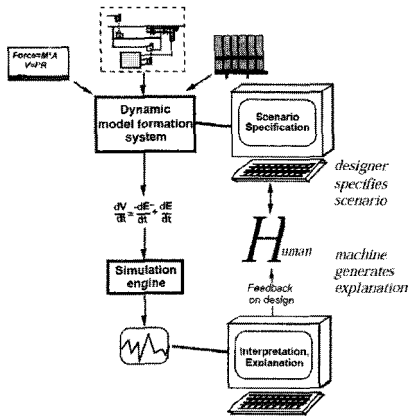



**Fig. 2.** 인간과 생성디자인 시스템간의 상호작용 프로세스 (Iwaski and Low 1991).

상기의 속성을 가지고 있는 생성디자인 시스템의 프로세스는 먼저 매우 간단한 단계에서부터 시작한다.

이 단계를 시작으로 계속적인 추가 작업을 통하여 점진적으로 복잡한 형태로 발전하고, 각각의 단계 또는 객체의 위계 질서를 구축하는 단계를 거치며, 디자이너는 생성시스템을 통해 생성된 대안들의 검토를 통하여 최적의 디자인 안을 찾게 된다. 이 프로세스에서 기본적인 디자이너의 아이디어는 디자인 알고리즘으로 표현되며, 생성시스템은 이 알고리즘을 기반으

로 대안을 생성하게 된다. Fig. 2는 디자이너와 생성 디자인 시스템 사이의 상호작용 프로세스를 보여 주고 있다.

이 프로세스에서 디자이너는 자신의 디자인 의도를 정의하여 생성디자인 시스템에 입력하고 생성디자인 시스템은 이를 바탕으로 가능한 대안들을 생성하여 디자이너에게 제시하면 디자이너가 이의 검토를 한 후, 만약 대안에 대해 만족하지 않으면 생성디자인 시스템에 입력한 내용을 수정하는 과정을 수행한다.

생성디자인을 위해서는 디자인의 요소를 표현하고 조정할 수 있는 방법이 필요하게 되며 이러한 방법으로는 파라메트릭 디자인 방법, 알고리즘 기반 디자인 방법, 도형문법 등이 필요하다.

## 3. 생성디자인을 위한 기술

생성디자인을 컴퓨터 상에서 구현하기 위해서는 각각의 도형의 관계를 설정하기 위하여 파라메트릭 디자인 기술과 알고리즘 기반 디자인 기법이 사용된다. 추가적으로 도형문법이 규칙을 정의하는데 사용되며, 도형문법이 이들 기술에 비하여 상위의 기술적 개념이라고 할 수 있다.

### 3.1 파라메트릭 디자인(Parametric Design)

파라메트릭 디자인(Parametric Design)은 컴퓨터상으로 표현되는 2차원 또는 3차원 객체를 용이하며 효과적으로 통제하기 위한 기반 기술로서, 파라메터디자인은 파라메터 컴포넌트, 파라메터 어셈블, 파라메터 컨트롤의 3가지 요소로 정의될 수 있다.

1. 파라메터 컴포넌트: 디자인 오브젝트들의 속성, 제약사항, 관계 등을 정의할 수 있게 한다. 즉, 벽돌, 바닥재, 유리 등의 요소들의 사이즈, 무게, 가격, 색상, 재질, 드로잉 특성, 방음 특성, 방화특성, 단열 특성 등 다양한 매개 변수들을 가질 수 있다.
2. 파라메터 어셈블: 이렇게 정의된 파라메터 컴포넌트들의 상호관계를 정의할 수 있게 하여, 파라메터 컴포넌트의 조합을 가능하게 한다.
3. 파라메터 컨트롤: 조합된 파라메터 컴포넌트들은 에 의해 디자인 규범 및 상호관계 수식에 기반을 두고 처리가 가능하다(Huw 2004).

파라메트릭 디자인의 모델링 방법은 변식기반(Propagation-based)과 구속조건기반(Constraint-based)의 두 가지로 구분될 수 있으며, 일반적으로 컴퓨터기반 건축디자인 도구에서 사용되는 모델링 방법은 후자인 구속조건기반이 사용되고 있다. 구속조건기반의 파라메트릭 모델링은 주어진 조건 하에서 파라메터를 조절하여 모델링하는 방법으로, 정형의 디자인에는 그 효율성이 극대화되지만 자유로운 형상, 특히 비정형의 디자인에는 적합하지 않은 방법이다. 이를 개선하기 위해 개발된 파라메트릭 모델링 기법이 변식기반이며 이 방법이 파라메트릭 디자인을 생성디자인 시스템으로 확장 가능하게 해주는 개념이다(Javier 2006).

### 3.2 알고리즘 기반 디자인(Algorithmic Design)

알고리즘의 기본개념은 기계적으로 실행가능하고 유한한 지시사항의 목록이며, 입력된 내용의 처리를 거쳐 출력을 하는 과정 중에서 처리를 담당하는 부분이다. 즉, 어떻게 입력 내용을 처리할 것인가를 지시하는 것이다. 디자인 알고리즘이란 디자인 요구에 대응하기 위해 디자인의도에 의해 디자인 결과물을 생성해내는 과정을 정리한 것이라고 볼 수 있다(Stiny and Gips 1978). 디자인 알고리즘은 디자인 과정을 의미한다고도 볼 수 있으며, 알고리즘 기반 디자인은 이를 모호한 형태가 아닌 논리적이며 수학적인 형태로 정의 하는 것을 의미한다고 볼 수 있다.

변식기반 파라메트릭 모델링 방법을 구현하기 위한 방법으로서 파라메터간의 상관관계를 표현하고, 전체 디자인의 논리적 구조의 표현을 수학적으로 표현하는 알고리즘 기반 디자인은, 생성디자인시스템에서 중요한 기술적 요소라고 할 수 있다. 디자이너가 컴퓨터에게 자신의 디자인 의도를 전달하는 방법인 알고리즘 기반 디자인을 통해서 컴퓨터는 디자인을 계산할 수 있게 되며, 복잡하고 이형의 디자인 형태를 논리적이며 체계적으로 처리할 수 있게 만든다.

### 3.3 도형문법

컴퓨터를 기반으로 형상을 생성하는 방법을 제안하기 위한 도형문법은 1972년 Stiny와 Gips가 "Shape Grammars and the Generative Specification of Painting and Sculpture: 회화와 조각의 생성 규정과 도형문법"이라는 논문을 통해서 최초로 제안을 했다.

이 논문에서 Stiny와 Gips는 도형을 기반으로 추상파 화가나 조각가가 작품을 창작할 때에 구조와 재료의 관계를 선택하고 난 후, 이를 그들만의 생성 규정에 의해 논리적으로 작품을 창작한다고 파악하고 이 활동에는 가장 근본이 되는 도형이 있으며 이 도형들이 어떠한 규칙을 가지고 조합이 되어 생성이 된다는

가정하에, 이를 논리적으로 표현하기 위해 Chomsky의 구절구조문법(Phrase Structure Grammar) 차용하여 도형문법을 제안하였다(Stiny and Gips 1972).

도형문법은 표지를 가지고 있는 초기도형 I에 정의된 도형 규칙들 중 선택된 규칙이 적용된 결과 도형을 만들어 내며, 다시 결과 도형의 요소에 표지를 부여하고 이 부분에 규칙을 적용하여 또다시 결과 도형을 생성해 낸다. 규칙 적용전의 도형은 계속 유지되며, 결과가 적용된 도형은 기존도형과 기하학적으로 유사한 종속도형을 갖게 된다

Stiny와 Gips가 도형문법을 제안한 초기의 도형문법은 유한한 규칙 집합에 의해 단지 형상 치환만하여 도형을 생성하는 비 파라메트릭 방법을 사용하였으며, 이후 컴퓨터에서의 구현을 위하여 도형과 규칙에 파라메터를 적용하여 유한한 규칙집합을 적용하여 도형의 다양성을 표현하는 파라메트릭 도형문법을 제안하였다. Fig. 3은 이의 예를 보여주고 있다. 그림에서 (A)의 초기 도형은 네 끝점의 좌표를 파라메터로 가지고 있으며, (B)는 규칙에 의해 생성된 도형이다.

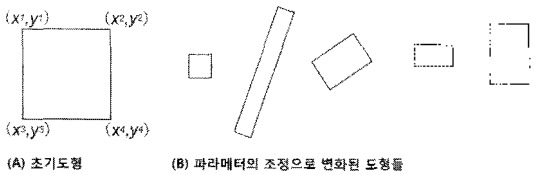



**Fig. 3.** 파라메터 도형문법-파라메터가 적용된 사각형(Tapia 1996).

### 3.4 CGA 도형문법

컴퓨터상에서 구현되는 상용화된 도형문법의 예로는 최근에 Müller가 제안한 CGA(Computer Graphic Architecture) 도형문법이 있다(Müller 등, 2006A). 절차적 건물 모델링(Procedural Building Modeling)을 위한 도형문법인 CGA 도형문법은 표현의 수준이 높은 건물의 외피를 작성하고, 저비용으로 컴퓨터 게임과 영화에 사용하기 위한 건축 모델을 만들기 위해 개발되었다. 스크립트 방식의 CGA도형문법의 규칙은 사용자가 도형의 위계적 관계 안에서 요소들간의 상호작용을 정의할 수가 있다.

CGA 도형문법은 도형을 정의하기 위한 도형(Fig. 4), 규칙을 정의하는 Notation으로 구성되어 있다. 규칙의 정의는 다음의 문법을 사용한다.

Id: predecessor: condition → successor

Id는 규칙의 번호를 의미하고, predecessor 규칙이 적용될 도형, condition은 규칙이 적용될 조건, successor는 규칙이 적용된 도형을 의미한다. 사용 예는 다음과 같다.

1: fac(h): h > 9→floor(h/3) floor(h/3) floor(h/3)

H의 높이를 가지고 있는 건물의 파사드가 9보다 크면 파사드 높이의 1/3 높이의 건물 바닥을 3개 만든다는 의미이다.

현재 CGA 도형문법은 CityEngine이라는 프로그램을 통해서 구현되고 있으며, 주로 도시계획, 고고학적 도시 복원에 사용되고 있다. Fig. 5는 멕시코 유카탄 반도의 Pucc 지역의 Xkipche이라는 고도시의 건물을 CGA 도형문법을 이용해 복원한 예이다.

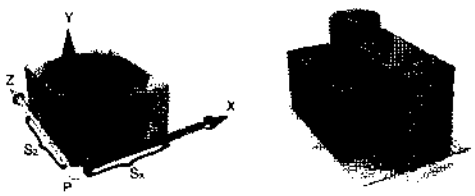



**Fig. 4.** CGA 도형문법의 3차원 도형정의, 오른쪽 그림은 3개의 정의된 도형을 조합하여 표현된 매스형태의 건물.

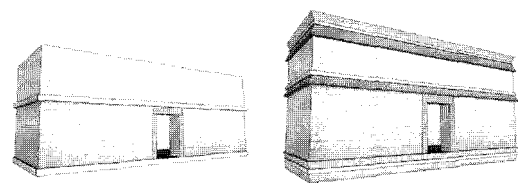


**Fig. 5.** CGA 도형문법을 이용해 복원한 Xkipche 건물.

CGA 도형문법은 진보적이며 구체적인 적용 방법을 제시하고 있지만, 아직까지는 정형건물의 표피 부분만을 처리 하고 있으며, 곡선이나 비정형적인 요소는 외부의 3차원 모델링 툴을 이용하여 작성한 객체를 분할한 표피에 배열하는 수준으로 적용하고 있다. 하지만 도시계획이나 도시모델링에서는 상당히 유용하게 사용되고 있다.

## 4. 도형문법 기반 생성디자인

도형문법은 3장에서 설명한 생성디자인을 위한 기술인 파라메트릭과 알고리즘기반 디자인을 포함할 수 있으므로, 본 연구에서는 생성디자인을 도형문법 기반으로 제시한다. 또한 CGA도형문법은 절차적 모델

링을 기반으로 하는 구조를 제시하고, 기존의 도형문법이 도형만으로 표시함으로 인해 전체적인 관계를 표현하기에 한계를 보임으로 이를 도입하였다. 도형문법 기반 생성디자인을 위해 본 연구에서는 비정형 건물의 외피디자인과 정형의 다리 구조체 디자인 사례를 이용하였다. 외피디자인은 형태의 도출(Form Finding)을 위한 사례이고, 구조체 디자인은 좀더 구체적인 디자인을 위한 사례이다.

이 사례를 위해서 사용된 환경은 McNeel의 Rhino 3D와 3차원 형태 생성을 위한 VPL(Visual Programming Language)의 일종인 Grasshopper를 사용하였다. 이 시스템환경의 특징은 기존의 프로그램 환경과는 다르게 프로그램을 코딩하는 것이 아니라 컴포넌트라 불리는 명령어 블록의 입출력 단을 서로 연결하여 완성하는 환경으로 매우 직관적이다. 규칙의 적용과 변경은 파라메터의 변경을 통하여 운영된다. 본 사례의 구현방법은 Fig. 6과 같다.

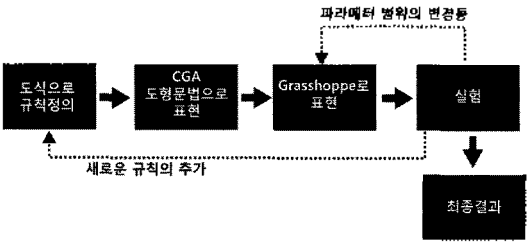



**Fig. 6.** 사례의 구현방법.

### 4.2 건물 외피 디자인 사례

외피 디자인을 위한 도형, 상호연관관계, 파라메터, 도형문법 규칙은 Fig. 7과 같다. 3개의 원이 Z축을 기준으로 서로 떨어져 있고, 각각의 원의 끝점을 연결하여 호를 만든다. 즉 호는 3개의 원의 크기와 Z 방향의 높이에 영향을 받는다. 이 설정에서 파라메터는 3개의 원의 중심점(Center 1.. 3), 3개의 원의 Z 축 위치(Level1.. 3), 그림에는 표시 안되어 있지만 3개의 원의 반지름(Radius1.. 3)이다. 호를 구성하는 3개의 원은 파라메터가 아니라 원들의 공간상 위치와 반지름에 의해서 결정되는 값이다. A에서 F 표지가 붙은 그림은 규칙을 보여 주고 있다. A는 초기 도형 B에서 F까지는 초기 도형의 변형 규칙이다. 이 규칙들은 파라메터를 가지고 있어 이의 조정으로 규칙 적용 이후 세부적인 조정이 가능하다.

E와 F는 3개의 원에 의해서 결정되는 호의 형태를 도식으로 보여 주고 있다. 이는 파라메터의 변경으로 인하여 다양한 형태 패턴이 생성될 수 있음을 보여준다.

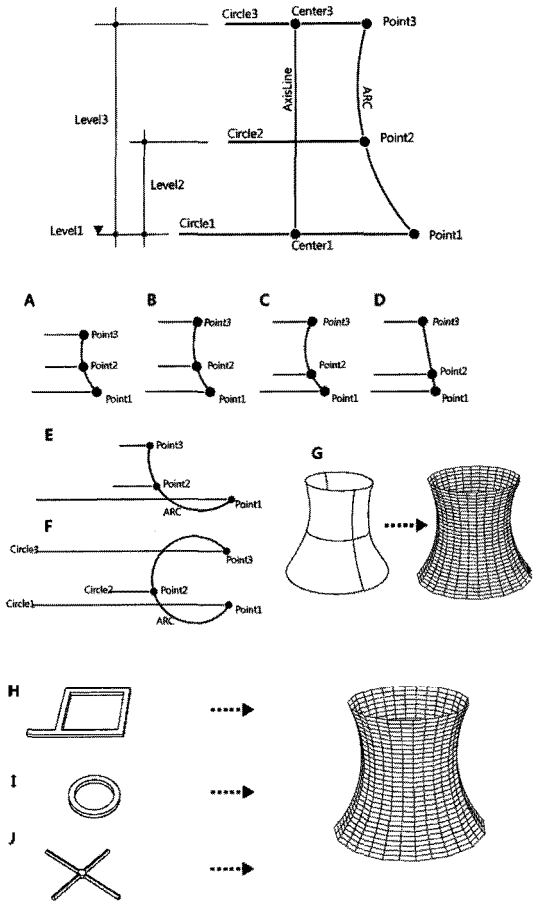



**Fig. 7.** 도형문법 작성을 위한 건물외피디자인 정의.

Fig. 7의 설정을 바탕으로 아래와 같은 16개의 규칙이 만들어 졌다. 이 표기방법은 CGA 도형문법의 기술 방법을 이용하여 표기했으나, CGA 도형문법 플랫폼인 CityEngine에서 이용될 수는 없다. 그 이유는 CGA가 3차원 비정형 형상을 지원하지 못하고 이를 위한 기능이 존재하지 않기 때문이다. 하기의 정의에서 CityEngine이 지원하지 않은 기능은 밑줄로 표시를 해 놨다. 이 내용의 구현은 Grasshopper를 이용하였으며, 이의 해석을 위하여 구문해석기는 개발되지 않은 관계로 사용되지 못했다. Grasshopper로 정의된 내용은 Fig. 8과 같다.

*1: Circle1(Center1, Level1, Radius1)*
*2: Circle2(Center2, Level2, Radius2)*
*3: Circle3(Center3, Level3, Radius3)*
*4: Point1(endpoint(Circle1))*
*5: Point2(endpoint(Circle2))*

*6: Point3(endpoint(Circle3))*
*7: AxisLine(Center1, Center3)*
*8: Arc(Point1, Point2, Point3)*
*9: FacadeObject("..\object1.3dm")*
*10: Arc → RevolutionSurface(Arc, AxisLine) Surface*
*11: Surface → Isotrim(U1, V1) {SubSurfaces}*
*12: SubSurfaces → Brepcomponents( ) {Vertices}*
*13: Vertices → Line(Vertives0, Vertices1) {Line1} ; Line(Vertices1, Vertices2) {Line2} ; Line(Vertices2, Vertices3) {Line3}*
*14: Line1 ; Line2 ; Line3 → Pipe(RadiusPi) {pipes}*
*15: Vertices → Surface4point(Vertices0, Vertices1, Vertices2, Vertices3) {faces}*
*16: {Vertices} →Morph(FacadeObject1) {FacadeObject1}*

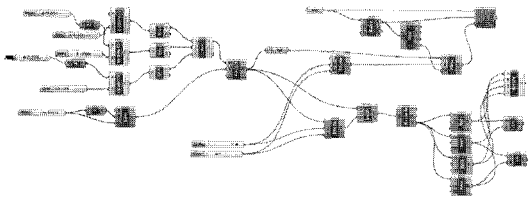


**Fig. 8.** 정의된 규칙을 Grasshopper로 정의한 내용.

이 모든 규칙은 순차적으로 이루어 지며, 상호 결합이 되어 있어 어느 한 파라메터가 수정이 되면 연관된 요소들이 조정이 되어 새로운 형태를 생성하게 된다. 이 파라메터의 변경은 새로운 규칙의 적용 방법으로 고려 되었으며, Fig. 7의 B, C, D, E, F 규칙의 적용은 기본 도형 요소의 파라메터의 변경으로 구현되어 진다.

규칙 1에서 9까지는 기본도형요소에 대한 정의이며, 괄호 안의 내용은 이 도형들이 가지고 있는 파라메터이다. 규칙 10은 Arc가 RevolutionSurface 명령을 이용하여 Surface로 변환되는 규칙이다.

Fig. 9의 A는 Circle 1의 반지름의 변경을 통하여 생성되어진 대안들이며, B는 Circle 2의 반지름의 변경을 통하여 생성되어진 대안들이며, C는 Circle 3의 반지름의 변경을 통하여 생성되어진 대안들이다. 파라메터의 범위는 3개의 반지름 모두 0에서 100까지이며 10씩 증가시키면서 생성해낸 결과이다. 이 생성된 대안들은 각각의 방법의 조합으로 또 다른 형태를 생성할 수 있으며, 각 단계별 적용된 파라메터의 값은 각 결과물의 하단에 기록 되므로 쉽게 조합의 생성이 가능하다.

이러한 생성과정을 거쳐서 찾아낸 4개의 대표적 결과물은 Fig. 10의 A, B, C, D이다. B는 Circle2의 반지름을 Circle1과 Circle2보다 크게 조정하여 생성된 결과이고, C는 Circle1과 Circle2의 반지름을 동시에 크게 조정하여 생성된 결과이다.

D는 C의 경우보다 반지름의 값을 크게 조정하여 생성된 결과로 Circle2가 위치한 부분의 변형된 형태를 보여주기 위해 Face 집합은 표시되지 않게 하고 랜더링한 모습이다. D는 상당히 의외적인 형상이 생성됨

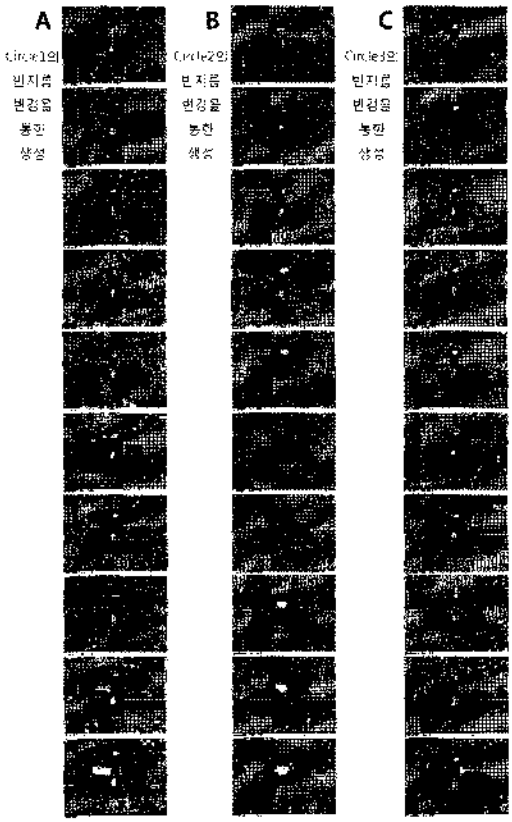



**Fig. 9.** 파라메터의 변경으로 생성결과들.

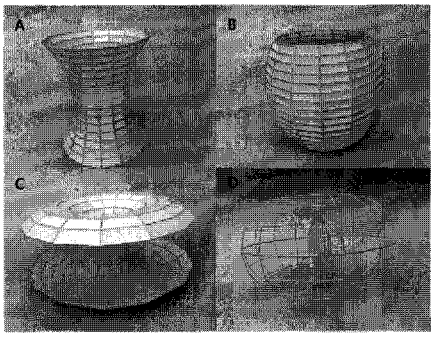



**Fig. 10.** 생성된 결과들.

을 보여줌으로써, 이 테스트를 통해 파라메터의 변경을 통해 다양하며, 의도하지 않은 발견인 UXD (Unexpected Discovery)의 결과이며, 생성디자인 시스템을 통해서 취할 수 있는 장점이라고 볼 수 있다.

규칙11에서 16은 형태 결정 후 외피의 디자인을 위한 규칙이다. 다양한 외피의 형태를 실험하기 위해 규칙11에서 Surface는 Isotream 명령으로 분할된 SubSurface 집합으로 생성된다. 여기서 {}의 의미는 다수의 요소가 생성됨을 의미한다. 즉 {SubSurfaces}는 유한한 분할된 SubSurface의 집합을 의미한다. 이 집합의 개수는 U와 V 파라메터로 결정된다. 이 규칙은 Fig. 6의 G 규칙을 사용한다.

규칙 12는 생성된 SubSurfaces에서 절점들의 좌표를 생성해내는 규칙이며, 규칙 13은 생성된 절점의 좌표를 이용하여 3쌍의 선의 집합을 생성하며, 규칙 14는 이렇게 생성해낸 라인들을 이용하여 프레임을 생성하는 규칙이다. 이렇게 생성된 프레임은 단순 사각 또는 삼각의 외피형태만 가능하다. 그 예는 Fig. 9에 보여진 예이다. 단일한 곡면외피는 외피에 설치되는 외피디자인 요소와 충돌하는 문제점을 가지고 있다. 이를 해결하기 위해서는 곡면이 평면사각형으로 분할되어야 한다. 규칙 15는 이를 위해 4쌍의 절점 좌표를 이용하여 평면 사각형으로 분할된 유한한 Face 집합을 만들어 낸다. 마지막으로 다양한 외피디자인을 위하여 규칙 16이 적용되는데 규칙9번에서 가져온 외부객체를 생성된 Surface에 Morph명령어를 이용하여 결합을 시키다. 이때 객체의 개수는 Subsurface의 개수와 동일하다. 이 규칙은 Fig. 7의 규칙 H, I, J 규칙을 사용한다. 이 규칙의 수는 실험하고자 하는 외피요소의 개수에 의해서 증감할 수 있다. Fig. 11은 이를 이용하여 생성된 결과이다. 각각의 결과는 좌측 상단의 외피 구성요소에 따라 각기 다른 형태의 외피 형상을 보여 주고 있다. 이 실험에서는 하나의 형상에 3가지 외피구성요소를 사용한 결과를 보여 주고 있으나, Fig. 9의 결과와 조합을 할 경우 기존의 수작업으로는 불가능한 수의 다양한 대안을 실험할 수 있다.

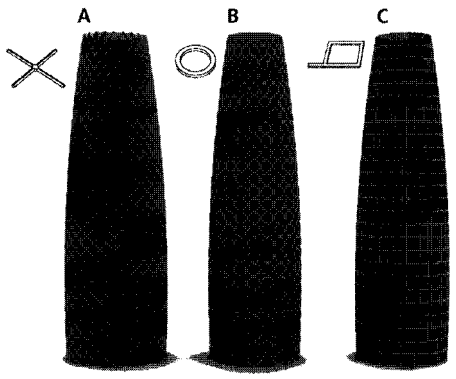



**Fig. 11.** 생성된 결과들.

### 4.2 구조체 디자인 사례

구조체 디자인사례는 형태도출 사례인 외피디자인이 개념 디자인 단계의 실험이므로, 그 이후 디자인 단계인 디자인의 구체화 단계를 위한 사례이다. 이 사례의 초기 목적은 다리를 위한 구조체를 디자인하는 것이나, 본 연구에서는 기본 골격만 대상으로 작업을 하였다. Fig. 12는 이 사례에서 사용된 초기 도형과 규칙, 파라메터를 보여주고 있다. 이 사례는 건물외피 디자인 사례에 비해 각각의 규칙간의 구속 조건이 강하기 때문에 의도하지 않은 발견과 같은 사례는 나타나지 않는다. 만약 인위적으로 파라메터 값의 범위를 넓게 하여 조정을 하게 되면 형태자체가 구성되지 않는다. 이 사례를 실험하게 된 중요한 이유는 강한 구속조건을 가지고 있는 사례에 대한 실험이다. 이 사례는 기본적인 형태는 유지하지만 세부적인 요소를 생성 하는 것에 관한 사례이다.

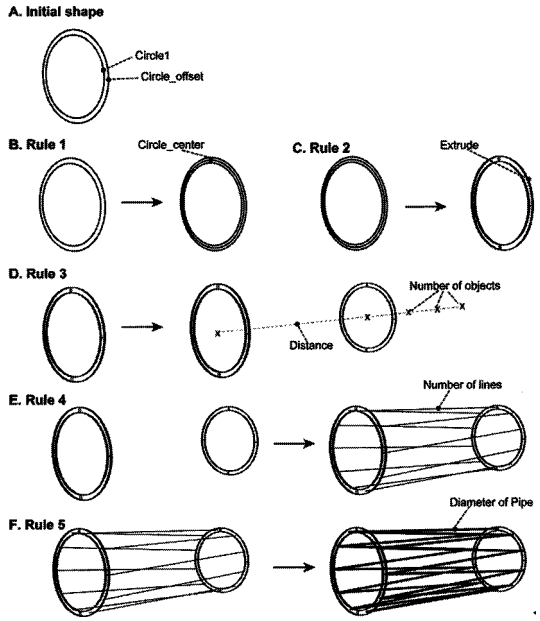



**Fig. 12.** 구조체 설계 사례를 위한 정의들.

사용된 파라메터의 내용은 다음과 같다.

1. 원형 구조체의 반지름(Circle1)
2. 원형 구조체의 두께(Circle_offset)
3. 원형 구조체의 깊이(Extrude)

4. 파이프의 길이(Length of Pipe)
5. 파이프의 개수(Number of Pipe)
6. 구조체의 개수(Number of Objects)
7. 파이프의 크기(Diameter of Pipe)

이 파라메터를 이용하여 생성한 결과는 Fig. 13과 같다. 여기서 파라메터의 값은 변경된 예를 쉽게 보여주기 위하여 범위를 넓게 잡아 주었으며 모두 5개씩만 보여 주고 있다. A는 원형 구조체의 두께(Circle_offset)의 변경, B는 원형 구조체의 깊이(Extrude) 변경, C는 파이프의 크기(Diameter of Pipe) 변경, D는 구조체의 개수(Number of Objects) 변경에 의한 생성을 보여 주고 있다.

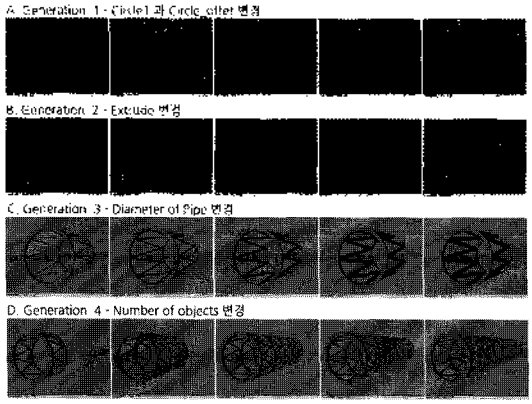



**Fig. 13.** 생성된 다리 구조물 결과.

## 4. 결 론

도형문법 기반의 생성디자인은 디자인 초기단계에서 다양한 대안을 생성하여, 디자이너의 디자인을 발전시킬 수 있게 해주며, 구체화 단계에서는 검토를 위한 다양한 대안을 제공해주며, 디자인의 논리적 구조를 확보할 수 있게 해준다. 특히 2차원 또는 3차원 정형 형태보다는 3차원 비정형 형태에서 그 효과가 크다고 할 수 있다. 3차원 비정형 형태는 그 복잡성과 형상의 좌표의 추적이 어려운 점이 이를 뒷받침해준다.

본 연구에서는 생성의 방법을 파라메터의 변경을 이용하며 적용 하였으나, 이는 완벽한 생성 방법이라고 할 수는 없다. 이를 위하여 추가 연구가 필요하며, 이는 사용된 도형문법이 컴퓨터가 해석을 할 수 있는 단계가 아니고 일종의 생성 디자인 시스템을 디자인 하는 계획서로서의 역할만 수행하고 있기 때문이다.

이를 해석해서 컴퓨터가 수행할 수 있는 해석 시스템이 필요하며, 이에 대한 추가 연구와 이를 위한 도형문법을 구체화 시키는 방법이 향후 연구에서 필요하다.

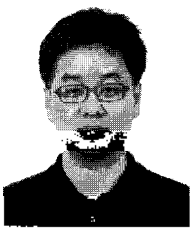
## 감사의 글

이 연구는 한양대학교 지속가능 건설기술 전문인력 양성 사업단 및 2008년도 선정 이공계대학원 연구장학생(유형1) 지원 결과의 일부임.

## 참고문헌

1. 김봉한, 김홍곤, 형상문법을 이용한 건축디자인 논리적 측면-기하학 평면 생성을 중심으로, *대한건축학회 학술발표논문집*, 제20권, 제1호, pp. 213-216, 2000.
2. Chase, S., "A Model for User Interaction In Grammar-based Design Systems", *Automation in Construction*, Vol. 11, pp. 161-172, 2002.
3. Huw, W. R., "The Importance of Parametrics in Building Information Modeling", *AECbytes Viewpoint #5*, May 13, 2004.
4. Iwasaki, Y. and Low, C. M., "Model Generation and Simulation of Device Behavior with Continuous and Discrete Changes", *Technical Report KSL-91-69*, Knowledge Systems Laboratory, Stanford University, 1991.
5. Javier Monedero, "Parametric Design: A Review and Some Experiences", *15th eCAADe-Conference Proceedings*, Vienna University of Technology, September, 2006.
6. Kuhn, T. S., *The Structure of Scientific Revolutions*, (Third Edition), University of Chicago Press, Chicago, Ill. 1996.
7. McCormack, J., Dorin, A. and Innocent, T., "Generative Design: A Paradigm for Design Research", *Proceedings of Futureground*, Design Research Society, 2004. Melbourne.
8. Müller, P., Vereenooghe, T., Wonka, P., Paap, I. and Gool, L. V., "Procedural 3D Reconstruction of Puuc Buildings in Xkipché", *The 7th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST*, 2006A.
9. Müller, P., Wonka, P., Haegler, S., Ulmer, A. and Gool, V. L., "Procedural Modeling of Buildings", *Proceedings of ACM SIGGRAPH 2006/ACM Transactions on Graphics 25*, 3, pp. 614-623, 2006B.
10. Stiny, G. and Gips, J., "Shape Grammars and the Generative Specification of Painting and Sculpture", *The Best Computer Papers of 1971*, pp. 125-135, 1972.
11. Stiny, G. and Gips, J., *Algorithmic Aesthetics*

*Computer Models for Criticism and Design in the Arts*, University of California Press, 1978.
12. Tapia, M. A., *From Shape to Style : Shape Grammars: Issues in Representation and Computation*, *Presentation and Selection*, Ph.D. Thesis, Department of Computer Science, University of Toronto, 1996.



**김 언 용**

1994년 명지대학교 건축학과 학사
1997년 명지대학교 건축공학과 석사
2001년 University of Sydney 박사 중퇴
2009년 한양대학교 건축환경공학과 박사수료
관심분야: Design Computing, BIM, City Information Modeling

**전 한 종**

1985년 한양대학교 건축공학과 학사
1996년 University of Sydney 공학박사
2007년 한양대학교 건축공학과 석사
현재 한양대학교 건축학부 부교수
관심분야: Architectural Design Process, Sustainable Architecture, BIM, City Information Model, 신한옥