

## 지오지브라를 이용한 3차원 구조물의 모델링 교육과 응용

정태은<sup>1†</sup> · 김태환<sup>2</sup>

<sup>1</sup>인덕대학교 메카트로닉스과, <sup>2</sup>주에듀프레스

### Education and Application of Modeling on 3D Structure using Geogebra

Tae-Eun Chung<sup>1†</sup> and Tae-Hwan Kim<sup>2</sup>

<sup>1</sup>Dept. of Mechatronics, Induk Univ.

<sup>2</sup>Edupresso, Co.

Received 12 February 2015; received in revised form 27 March 2015; accepted 27 March 2015

#### ABSTRACT

In this study, we organize and explain various ways to construct 3D models in the 2D plane using Geogebra, mathematical education software that enables us to visualize dynamically the interaction between algebra and geometry. In these ways, we construct three unit vectors for 3 dimensions at a point on the Cartesian coordinates, on the basis of which we can build up the 3D models by putting together basic mathematical objects like points, lines or planes. We can apply the ways of constructing the 3 dimensions on the Cartesian coordinates to modeling of various structures in the real world, and have chances to translate, rotate, zoom, and even animate the structures by means of slider, one of the very important functions in Geogebra features. This study suggests that the visualizing and dynamic features of Geogebra help for sure to make understood and maximize learning effectiveness on mechanical modeling or the 3D CAD.

**Key Words:** 3D models, Geogebra, Kinematic modeling, Mathematical software, Slider

### 1. 서 론

Geogebra는 기하(geometry)를 의미하는 Geo와 대수(algebra)를 의미하는 gebra의 합성어이다. Geogebra는 대수학(algebra), 기하학(geometry), 미적분학(calculus), 통계학(statistics)을 학습하는 영역에서 매우 유용한 수학 소프트웨어로 비영리적인 목적을 위해서는 무료로 사용할 수 있는 오픈 소스(open source)이면서 자바(Java)로 만들어진 프로그램이다. 객체지향언어인 자바로 만들어져

있어 자바언어가 가지고 있는 “운영체제에 종속되지 않고 웹과의 연동이 쉽다”는 특징을 가지고 있고, 직관적인 사용자 인터페이스로 상호작용이 가능한 동적인 웹 페이지를 빠르게 만들어 웹에서 손쉽게 공유할 수 있다. 또한 대수 창(algebra view)과 기하 창(graphics view) 외에 컴퓨터 대수 시스템(CAS: Computer Algebra System)을 구현하는 CAS 창(CAS view)과 스프레드 시트 창(spreadsheet view)까지 서로 연동이 되면서 초등 수학에서 대학 수준의 여러 종류의 다양한 수학적 주제들까지 활용이 가능한 동적인 수학 소프트웨어(DMS: Dynamic Mathematics Software)이다<sup>[1,2]</sup>.

교육 현장에서는 Geogebra를 시각적 표현에 기

<sup>†</sup>Corresponding Author, techung@induk.ac.kr  
©2015 Society of CAD/CAM Engineers

반한 수학 교육에 활용하고 있으며 이러한 추세는 점차 증가하고 있으나 아직까지는 주로 2차원 문제에 적용되고 있다<sup>13)</sup>.

본 연구에서는 Geogebra 4.2 버전을 활용해서 2차원 직교 좌표계에서 3차원 모델을 구성하는 대표적인 두 가지 방법을 제시한다. 2차원 직교 좌표계에서 3차원 모델을 표현하기 위해 3개의 2차원 표준기저벡터(standard basis vector)를 만든 후에 이러한 벡터들을 토대로 점(point), 선(line), 면(plane) 등의 Geogebra에서 제공하는 다양한 수학적 객체(mathematical object)들을 이용해서 3차원 벡터를 표현한다. 3차원 벡터의 표현원리를 여러 형태의 3차원 구조물에 적용해 기구학적 모델링을 진행한 후, Geogebra의 핵심 객체인 슬라이더(slidebar) 도구를 이용해 평행이동, 회전이동, 확대, 축소 등의 다양한 애니메이션을 구현해 본다.

2차원 직교 좌표계에서 3차원 모델을 표현하기 위해 3개의 2차원 표준기저벡터들을 만들어야 하는데, 본 연구에서 제시하는 두 가지 방법은 다음과 같다.

첫째는, 3차원 직교 좌표계에서 각 축을 중심으로 회전하는 오일러 각(Euler angle)을 사용한 회전행렬(rotation matrix)을 이용하는 방법으로 x, y, z 축을 중심으로 회전하는 각도를 나타내는 세 개의 매개변수  $\alpha, \beta, \gamma$ 를 슬라이더 객체로 만들어 값들을 변경하면서 3차원 보기(3D view)를 구현한다.

둘째는, 구면 좌표계(spherical coordinate system)의 좌표성분으로 사용하는 편각(polar angle)  $\theta$ 와 방위각(azimuthal angle)  $\phi$ , 반지름 r을 슬라이더 객체로 만들어 이용하는 방법이다. 2차원 직교 좌표계의 x, y 좌표를 이러한 매개변수들의 식으로 표현하고 매개변수 값들을 변경하면서 3개의 기저벡터를 만들어 3차원 보기를 구현한다. 이 기법을 확장해서 응용하면 슬라이더 객체 대신 임의의 위치에 시점 객체(view point object)를 생성한 후, 시점 객체의 x 좌표를  $\phi$  값으로, y 좌표를  $\theta$  값으로 사용하면 매 순간 시점 객체의 이동에 따라 3차원 보기를 다양하게 구현할 수 있다.

## 2. 기저 벡터의 구성 방법

### 2.1 직교 좌표계의 오일러 각을 이용한 방법

#### 2.1.1 오일러 각

오일러 각(Euler angle)은 강체(rigid body)가 놓

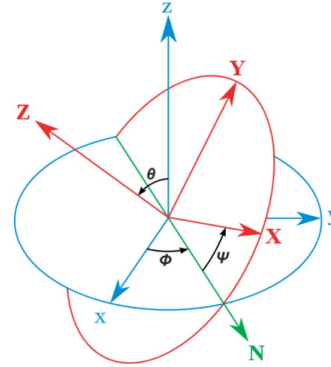


Fig. 1 Euler angles representing rotations about z, N, and Z axis

인 위치와 방향을 3차원 공간에 표시하기 도입한 세 개의 각도이다<sup>14,15)</sup>. 3차원 공간에 놓인 강체의 위치와 방향은 오일러 각도를 사용하여 세 번의 회전을 통해 얻을 수 있다. 오일러 각을 이용한 회전변환(rotation transform)은 x, y, z 축 중 하나의 축을 선택하여 회전하는 과정을 3번 반복하는데, 회전 축의 조합이 다양하게 만들어질 수 있다. 오일러 각을 이용한 회전변환은 회전 순서에 따라 결과가 다르게 나타난다.

오일러 각은  $\phi$ (phi),  $\theta$ (theta),  $\psi$ (psi)로 부르는 세 각의 조합으로 표시하는데, 일반적으로 회전하는 축에 상관없이 첫 번째 회전한 각을  $\phi$ , 두 번째 회전한 각을  $\theta$ , 세 번째 회전한 각을  $\psi$ 로 정의한다. Fig. 1은 z 축을 중심으로  $\phi$ , N 축을 중심으로  $\theta$ , Z 축을 중심으로  $\psi$ 만큼 회전하는 오일러 각 회전변환을 나타낸다<sup>15)</sup>. 여기서 x-y-z는 원래의 좌표축(original system)이고 X-Y-Z는 회전된 좌표축(rotated system)이다. N 축은 원래의 좌표축에서 z 축을 중심으로  $\phi$  만큼 회전한 후의 x축의 위치를 나타내며 교선(line of nodes)이라고 한다.

#### 2.1.2 내부 회전과 외부 회전

초기의 고정된 좌표계를 x-y-z로, 매 순간의 회전하는 좌표계를 X-Y-Z로 표기하기로 한다.

내부회전(intrinsic rotation)은 회전하는 좌표계 X-Y-Z의 각 좌표축을 중심으로 회전하는 경우에 발생하고, 외부회전(extrinsic rotation)은 고정된 좌표계 x-y-z의 각 좌표축을 중심으로 회전하는 경우에 발생한다. 3차원 공간상의 동일한 위치와 방향을 나타내는 강체를 내부회전과 외부회전으로 표현할 수 있다. x-y-z 좌표계나 X-Y-Z 좌표계에

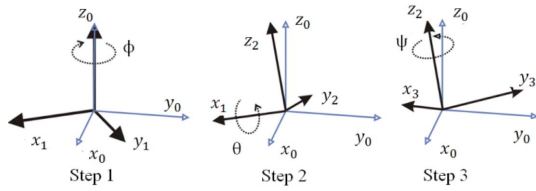


Fig. 2 A rotation represented by Euler angles  $(\varphi, \theta, \psi) = (-60^\circ, 30^\circ, 45^\circ)$ , using Z-X-Z intrinsic rotations

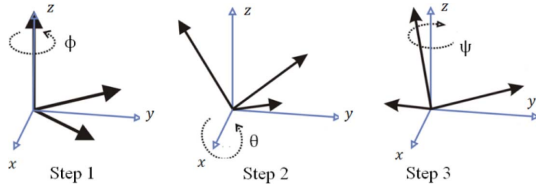


Fig. 3 A rotation represented by Euler angles  $(\varphi, \theta, \psi) = (45^\circ, 30^\circ, -60^\circ)$ , using Z-X-Z extrinsic rotations

서 각 축을 중심으로 회전하는 각도를 각각  $\alpha, \beta, \gamma$ 라 한다면, X 축으로  $\alpha$ , Y 축으로  $\beta$ , Z 축으로  $\gamma$  만큼 회전하는 내부회전은 z 축으로  $\gamma$ , y 축으로  $\beta$ , x 축으로  $\alpha$  만큼 회전하는 외부회전과 같다. 각 축을 중심으로 회전하는 각도는 같지만 회전의 순서는 반대가 된다. Fig. 2는  $(\varphi, \theta, \psi) = (-60^\circ, 30^\circ, 45^\circ)$  가 되는 Z-X-Z 형식의 내부회전을 사용한 것이고, Fig. 3은  $(\varphi, \theta, \psi) = (45^\circ, 30^\circ, -60^\circ)$ 가 되는 z-x-z 형식의 외부회전을 사용한 것이다. 두 그림의 결과에서 회전각은 동일하고 회전의 순서는 반대가 되지만 회전의 결과는 동일함을 확인할 수가 있다. 고정된 좌표계 x-y-z에서 각 축을 중심으로 회전하는 각도를 나타내는 세 개의 매개변수  $\alpha, \beta, \gamma$ 를 사용하는 외부 회전을 이용하여 회전 행렬을 구성한다.

2.1.3 회전 행렬의 구성

2차원 직교 좌표계에서 원점을 중심으로  $\theta$ 만큼 회전하는 회전행렬의 구성원리를 Fig. 4의 3차원 좌표계에 적용한다. 고정된 좌표계의 각 축을 중심으로  $\alpha, \beta, \gamma$ 만큼 회전하는 기본회전행렬은 공간상의 한 점을 열 벡터로 뒤쪽에 곱하는 경우와 다음과 같이 구성할 수 있다.

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

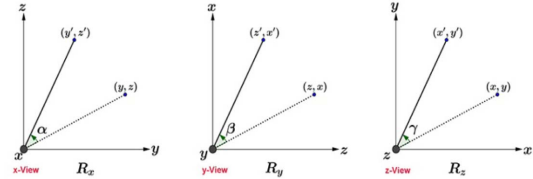


Fig. 4 Rotation about the x, y and z axis

$$R_y = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix}$$

$$R_z = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

여기서 구성한 세 개의 기본회전행렬을 차례로 곱하면 3차원 직교 좌표계에 있는 한 점을 x축으로  $\alpha$ , y 축으로  $\beta$ , z 축으로  $\gamma$ 만큼 회전시키는 회전행렬 R이 구해진다. 행렬의 교환법칙은 성립하지 않기 때문에 이 회전행렬은 x-y-z 형식의 외부 회전이 된다.

$$R = R_z R_y R_x$$

3차원 공간상의 한 점을  $P(p_x, p_y, p_z)$ 라 하고 앞에서 구한 회전행렬을 적용해서 회전한 점을  $Q(q_x, q_y, q_z)$ 라 한다. 또한 3차원 공간의 세 개의 기본 단위벡터(fundamental unit vector)를 각각  $E_1, E_2, E_3$ 라 하면,

$$P = p_x \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + p_y \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + p_z \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

$$= p_x E_1 + p_y E_2 + p_z E_3 \tag{2}$$

$$Q = RP = p_x R_x E_1 + p_y R_y E_2 + p_z R_z E_3 \tag{3}$$

yz 평면에 수직인 x 축 방향을 시점으로 정한다면 2차원 화면에 보여지기 위해 필요한 좌표값은 Q의 y 성분과 z 성분이다.  $R_x E_1, R_y E_2, R_z E_3$ 을 각각  $F_1, F_2, F_3$ 라 하고 이 벡터들의 y 성분과 z 성분을 가지는 2차원 벡터를 각각  $G_1, G_2, G_3$ 라 한다면, 이 세 개의 벡터가 2차원 기저벡터가 된다. Q의 y 성분과 z 성분을 가지는 2차원 벡터를 V라 하면 최종적으로 다음의 식을 얻을 수 있다.

$$F_1 = RE_1, F_2 = RE_2, F_3 = RE_3$$

$$V = p_x G_1 + p_y G_2 + p_z G_3 \quad (4)$$

여기서 구한 2차원 벡터  $V$ 는 3차원 공간좌표표를 2차원 평면좌표에서 표현하기 위해 사용된다.

위의 과정에서 성분 별로 적용하여 바로 기저 벡터를 구할 수도 있지만, 본 논문에서는 3차원 기본 단위인  $E_1, E_2, E_3$  을 회전시켜서 최종적으로 3개의 2차원 기저벡터인  $G_1, G_2, G_3$  을 그래픽스 창에서 가시적으로 구현하는 부분이 중요했기 때문에 식 (2),(3),(4)에서  $F = RE$ 를 도입하여 전개하였다. 또한 3차원 점이 지오지브라의 2차원 그래픽스 창에서 가시적으로 표현이 되지 않기에 이러한 전개 방식( $E$ 가 회전한  $F$  행렬에서 지오지브라의 element 명령을 이용해서  $G$ 를 구하는 과정)이 학생들에게 2차원 기저벡터가 시각적으로 구현되는 과정을 직접 실습해 볼 수 있어서 보다 교육적 효과가 있다.

### 2.1.4 Geogebra에서 기저 벡터 구성

앞서 설명한 내용을 토대로 3차원이 지원되지 않은 Geogebra 4.2 버전에서 기저벡터를 구성해 보도록 한다. Geogebra에서 모델구성의 각 단계를 Table 1의 구성 프로토콜(construction protocol)에서 확인할 수 있다.

#### (1) 오일러 각의 생성

오일러 각을 나타내는 세 개의 슬라이더 객체  $\alpha, \beta, \gamma$ 를 생성한다. 세 슬라이더 모두 각의 범위는 최소값  $0^\circ$ , 최대값  $360^\circ$ 로, 간격은  $1^\circ$ 로 동일하게 설정한다. [구성 프로토콜: 01~03]

#### (2) 기본단위벡터의 구성

3차원 공간에서 한 점을 표시하기 위해 세 개의 기본단위벡터  $E_1, E_2, E_3$  를 구성한다. Geogebra 입력창에 다음의 명령어를 순서대로 입력한다.

[구성 프로토콜: 04~06]

$$E\_1 = \{\{1\}, \{0\}, \{0\}\}$$

$$E\_2 = \{\{0\}, \{1\}, \{0\}\}$$

$$E\_3 = \{\{0\}, \{0\}, \{1\}\}$$

#### (3) 3차원 회�행렬의 구성

3차원 공간에서의 각 축을 중심으로 각각  $\alpha, \beta, \gamma$  만큼의 회전을 구현하는 세 개의 기본회�행렬

**Table 1** Geogebra construction protocol of 3D model using Euler angles

No	Name	Command
1	Angle $\alpha$	
2	Angle $\beta$	
3	Angle $\gamma$	
4	List $E_1$	
5	List $E_2$	
6	List $E_3$	
7	List $R_x$	$\{\{1,0,0\}, \{0,\cos(\alpha),-\sin(\alpha)\}, \{0,\sin(\alpha),\cos(\alpha)\}\}$
8	List $R_y$	$\{\{\cos(\beta),0,\sin(\beta)\}, \{0,1,0\}, \{-\sin(\beta),0,\cos(\beta)\}\}$
9	List $R_z$	$\{\{\cos(\gamma),-\sin(\gamma),0\}, \{\sin(\gamma),\cos(\gamma),0\}, \{0,0,1\}\}$
10	List $R$	$R_z R_y R_x$
11	List $F_1$	$RE_1$
12	List $F_2$	$RE_2$
13	List $F_3$	$RE_3$
14	Point $G_1$	$(Element[Element[F_1, 2], 1], Element[Element[F_1, 3], 1])$
15	Point $G_2$	$(Element[Element[F_2, 2], 1], Element[Element[F_2, 3], 1])$
16	Point $G_3$	$(Element[Element[F_3, 2], 1], Element[Element[F_3, 3], 1])$
17	Vector $g_1$	$Vector[G_1]$
18	Vector $g_2$	$Vector[G_2]$
19	Vector $g_3$	$Vector[G_3]$
20	Quadrilateral XY	$Polygon [G_1+G_2, -G_1+G_2, -G_1-G_2, G_1-G_2]$
20	Segment a	$Segment[G_1+G_2, -G_1+G_2, XY]$
20	Segment b	$Segment[-G_1+G_2, -G_1-G_2, XY]$
20	Segment c	$Segment[-G_1-G_2, G_1-G_2, XY]$
20	Segment d	$Segment[G_1-G_2, G_1+G_2, XY]$
21	List P	
22	Number $p_x$	$Element[Element[P, 1], 1]$
23	Number $p_y$	$Element[Element[P, 2], 1]$
24	Number $p_z$	$Element[Element[P, 3], 1]$
25	Point V	$p_x G_1 + p_y G_2 + p_z G_3$
26	Vector v	$Vector[V]$

을 구성하고 이들을 차례로 곱해서 3차원 회�행렬을 구성한다. 이 회�행렬은 x-y-z 형식의 외부 회전이다. [구성 프로토콜: 07~10]

$$R_x = \{\{1, 0, 0\}, \{0, \cos(\alpha), -\sin(\alpha)\}, \{0, \sin(\alpha), \cos(\alpha)\}\}$$

$$R_y = \{\{\cos(\beta), 0, \sin(\beta)\}, \{0, 1, 0\}, \{-\sin(\beta), 0, \cos(\beta)\}\}$$

$$R_z = \{\{\cos(\gamma), -\sin(\gamma), 0\}, \{\sin(\gamma), \cos(\gamma), 0\}, \{0, 0, 1\}\}$$

$$R = R_z R_y R_x$$

(4) 기본단위벡터의 회전

(3)에서 구성한 3차원 회전행렬을 이용해서 (2)에서 구성한 기본단위벡터  $E_1, E_2, E_3$  를 회전시킨 벡터  $F_1, F_2, F_3$  를 구성한다. [구성 프로토콜: 11~13]

$$F_1 = R E_1$$

$$F_2 = R E_2$$

$$F_3 = R E_3$$

(5) 2차원 기저벡터의 구성

2차원 기저벡터 구성을 위한  $G_1, G_2, G_3$  를 생성한다. Geogebra 입력창에 다음의 명령어를 차례대로 입력한다. [구성 프로토콜: 14~16]

$$G_1 = (\text{Element}[\text{Element}[F_1, 2], 1], \text{Element}[\text{Element}[F_1, 3], 1])$$

$$G_2 = (\text{Element}[\text{Element}[F_2, 2], 1], \text{Element}[\text{Element}[F_2, 3], 1])$$

$$G_3 = (\text{Element}[\text{Element}[F_3, 2], 1], \text{Element}[\text{Element}[F_3, 3], 1])$$

다음으로 Geogebra 기하창(Graphics View)에 벡터를 구성하기 위해 입력 창에 다음의 명령어를 순서대로 입력한다. [구성 프로토콜: 17~19]

$$g_1 = \text{Vector}[G_1]$$

$$g_2 = \text{Vector}[G_2]$$

$$g_3 = \text{Vector}[G_3]$$

(6) XY 평면의 구성

회전하는 좌표계 X-Y-Z에서 XY 평면을 구성하기 위해 Geogebra 입력창에 다음의 명령어를 입력한다. [구성 프로토콜: 20]

$$XY = \text{Polygon}[G_1 + G_2, -G_1 + G_2, -G_1 - G_2, G_1 - G_2]$$

(7) 한 점의 생성과 성분 추출

Geogebra의 리스트 객체(List Object)를 이용해서 열 벡터(column vector)  $P(1, 1, 1)$ 를 생성하고

$P$ 의 x, y, z 성분을 추출해서 각각  $p_x, p_y, p_z$ 에 저장한다. Geogebra 입력창에 다음의 명령어를 순서대로 입력한다. [구성 프로토콜: 21~24]

$$P = \{\{1\}, \{1\}, \{1\}\}$$

$$P_x = \text{Element}[\text{Element}[P, 1], 1]$$

$$P_y = \text{Element}[\text{Element}[P, 2], 1]$$

$$P_z = \text{Element}[\text{Element}[P, 3], 1]$$

(8) 2차원 벡터 구성

(7)에서 구한 결과들을 식 (4)에 대입해서 2차원 벡터  $v$ 를 구성한다. Geogebra 입력창에 명령어를 순서대로 입력한다. [구성 프로토콜: 25~26]

$$V = p_x G_1 + p_y G_2 + p_z G_3$$

$$v = \text{Vector}[V]$$

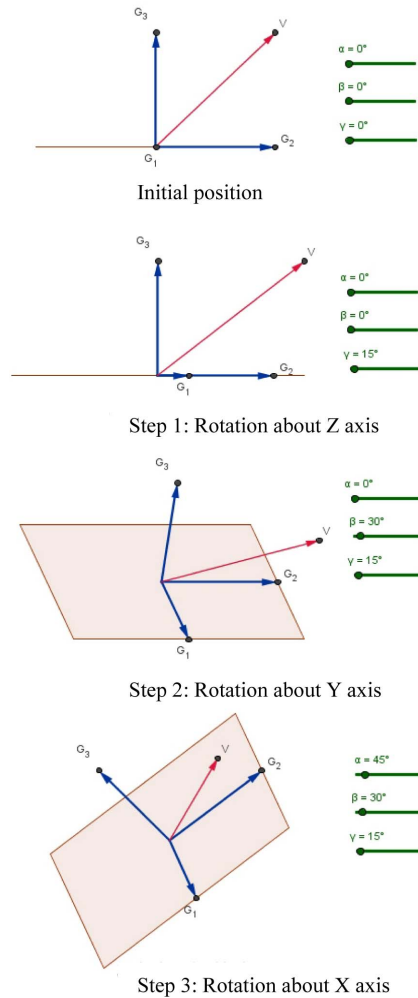


Fig. 5 A rotation represented by Euler angles  $(\varphi, \theta, \psi) = (15^\circ, 30^\circ, 45^\circ)$ , using Z-Y-X intrinsic rotations

Fig. 5는 지금까지 Geogebra에서 구성한 모델의 결과이다. Fig. 5에서 볼 수 있는 바와 같이 x-y-z 형식의 외부 회전  $(\varphi, \theta, \psi) = (45^\circ, 30^\circ, 15^\circ)$ 와 Z-Y-X 형식의 내부 회전  $(\varphi, \theta, \psi) = (15^\circ, 30^\circ, 45^\circ)$ 의 결과가 같음을 알 수 있다. x-y-z 형식의 외부 회전을 가정하고 3차원 회전행렬을 구성했기 때문에 내부 회전을 화면상에서 시각적으로 확인하려면 반드시  $\gamma, \beta, \alpha$  순서대로 회전각을 변화시켜야 한다. Table 1은 최종적으로 완성된 모델의 구성 프로토펙이다.

**2.2 구면 좌표계의 편각, 방위각을 이용한 방법**

**2.2.1 편각과 방위각**

구면 좌표계(spherical coordinate system)는 3차원 공간 상의 점들을 나타내는 좌표계의 하나로  $r, \theta, \varphi$ 의 세 좌표를 사용한다<sup>[6]</sup>. 공간상에 주어진 점을 P라 하면,  $r$ 은 원점으로부터 P까지의 거리이고  $\theta$ 는 z 축의 양의 방향으로부터 원점과 P가 이루는 직선까지의 각,  $\varphi$ 는 x 축의 양의 방향으로부터 원점과 P가 이루는 직선을 xy 평면에 투영시킨 직선까지의 각이다. 여기서  $r$ 은 반경(radius or radial distance),  $\theta$ 를 편각(inclination or polar angle),  $\varphi$ 를 방위각(azimuth or azimuthal angle)이라 한다. Fig. 6은 구면 좌표계의  $r, \theta, \varphi$ 를 보여주고 있다<sup>[5]</sup>.

구면 좌표계에서 직교 좌표계로 변환할 때 다음과 같은 관계식이 사용된다.

$$\begin{aligned} x &= r \sin \theta \cos \varphi \\ y &= r \sin \theta \sin \varphi \\ z &= r \cos \theta \end{aligned} \tag{5}$$

역으로, 직교 좌표계에서 구면 좌표계로 변환하면 다음과 같은 관계식이 사용된다.

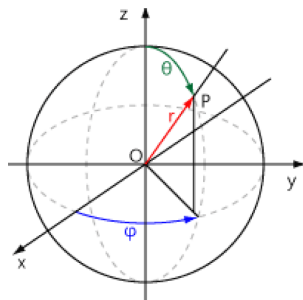


Fig. 6 Spherical coordinates  $(r, \theta, \varphi)$

$$\begin{aligned} r &= \sqrt{x^2 + y^2 + z^2} \\ \theta &= \cos^{-1}(z/r) \\ \varphi &= \tan^{-1}(y/x) \end{aligned} \tag{6}$$

본 연구에서는 2차원 직교 좌표계의 x, y 좌표를 구면 좌표계의 반경  $r$ , 편각  $\theta$ 와 방위각  $\varphi$ 의 식으로 표현하고 이 값들을 변경해 가면서 2차원 직교 좌표계에서 3개의 기저벡터를 만들어 3차원 보기(3D view)를 구현한다.

**2.2.2 회전행렬의 구성**

본 연구에서는 시점(view point)에 따른 기저벡터를 2차원 직교 좌표계에서 표현하기 위해서 앞서 정의한 구면 좌표계의 편각과 방위각의 증가 방향을 다른 방법으로 정의하기로 한다. Fig. 7은 xy 평면에 수직인 z 축 방향에서 바라본 평면도(top view)를 나타내는데 구면 좌표계의 적도면(equatorial plane)을 위에서 본 모습과 같다. 시점(view point)을 적도면에서 북극(north celestial pole)으로 이동하면서 측정한 각을  $\theta$ 의 양의 방향으로 정의하기로 한다면 Fig. 7은  $\theta = 90^\circ$ 에서 바라본 모습이 된다. 시점(view point)을 오른쪽으로 이동하면서 측정한 각을  $\varphi$ 의 양의 방향으로 정의한다면 시점을 오른쪽으로  $\varphi$  만큼 이동하면 XY 평면은 왼쪽으로  $\varphi$  만큼 이동하게 된다. 또한 Fig. 7의 xy 평면은 이후의 내용에서 구성하게 되는 기저벡터의 기준좌표계가 된다.

앞서 정리한 내용을 토대로 회전행렬을 구성해 보면, 먼저  $\theta = 90^\circ$ 이고, Z 축을 중심으로  $\varphi$  만큼 회전하는 경우의 회전행렬은 Fig. 7을 참고해서 다음과 같이 구성한다.  $\varphi = 0^\circ$ 에서 XY 평면이 됨을 확인할 수 있다.

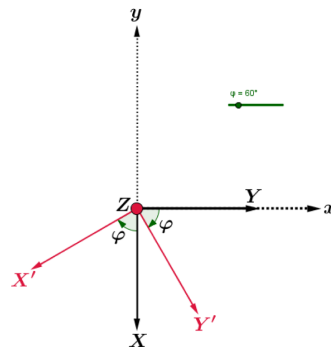


Fig. 7 Top view of rotation about z axis in spherical coordinate system

$$\begin{bmatrix} \cos(-(90^\circ + \varphi)) & -\sin(-(90^\circ + \varphi)) & 0 \\ \sin(-(90^\circ + \varphi)) & \cos(-(90^\circ + \varphi)) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\sin(\varphi) & -\cos(\varphi) & 0 \\ \cos(\varphi) & -\sin(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

여기에 방위각  $\theta$ 의 영향을 고려해서 위에서 구한 회전행렬의 y 성분(2열)에  $\sin(\theta)$ 를 곱해준다.  $\theta=0^\circ$ 에서는 y 성분은 0이 되고  $\theta$ 가 증가하면서 y성분도 증가하게 되어  $\theta=90^\circ$ 에서는 Fig. 7과 같은 모습이 된다. 또한 회전행렬의 z 성분(3열)에  $\cos(\theta)$ 를 곱해주면  $\theta=0^\circ$ 에서는 z 성분은 1이 되고  $\theta$ 가 증가하면서 z 성분은 감소하게 되어  $\theta=90^\circ$ 에서는 0이 된다. 요약 정리하면 2차원 xy 평면에서 3차원을 구현하기 위한 3개의 2차원 기저벡터는 다음과 같이 표현된다.

$$\begin{aligned} g_1 &= (-\sin(\varphi), -\cos(\varphi) \cdot \sin(\theta)) \\ g_2 &= (\cos(\varphi), -\sin(\varphi) \cdot \sin(\theta)) \\ g_3 &= (0, \cos(\theta)) \end{aligned} \quad (8)$$

### 2.2.3 Geogebra에서 기저벡터 구성

앞서 설명한 내용을 토대로 3차원이 지원되지 않은 Geogebra 4.2 버전에서 기저벡터를 구성해 보도록 한다. Geogebra에서는 모델구성의 각 단계를 Table 2의 구성 프로토콜(construction protocol)에서 확인할 수 있다.

#### (1) 편각과 방위각의 생성

편각과 방위각 나타내는 두 개의 슬라이더 객체  $\theta, \varphi$ 를 생성한다. 슬라이더  $\theta$ 의 각의 범위는 최소값  $0^\circ$ , 최대값  $90^\circ$ 로, 간격은  $1^\circ$ 로 설정하고 슬라이더  $\varphi$ 의 각의 범위는 최소값  $0^\circ$ , 최대값  $360^\circ$ 로, 간격은  $1^\circ$ 로 설정한다.

[구성 프로토콜: 01~02]

#### (2) 2차원 고정기저벡터의 구성

세 개의 2차원 고정기저벡터(fixed basis vector)  $g_1, g_2, g_3$ 를 구성한다. Geogebra 입력창에 다음의 명령어를 순서대로 입력한다.

[구성 프로토콜: 03~05]

$$\begin{aligned} g_1 &= \text{Vector}[(-\sin(\varphi), -\cos(\varphi) \sin(\theta))] \\ g_2 &= \text{Vector}[(\cos(\varphi), -\sin(\varphi) \sin(\theta))] \\ g_3 &= \text{Vector}[(0, \cos(\theta))] \end{aligned}$$

#### (3) 기준점과 확대계수 생성

원점(origin point) O와 확대계수(magnification factor) M을 생성한다. 슬라이더 M의 범위는 최소값 1, 최대값 5로, 간격은 0.1로 설정한다. [구성 프로토콜: 06~07]

#### (4) 2차원 이동기저벡터의 구성

세 개의 2차원 이동기저벡터(movable basis vector)  $u, v, w$ 를 구성한다. Geogebra 입력창에 다음의 명령어를 순서대로 입력한다.

$$\begin{aligned} &[\text{구성 프로토콜: 08~13}] \\ U &= \text{Translate}[O, \text{Vector}[M g_1]] \\ V &= \text{Translate}[O, \text{Vector}[M g_2]] \\ W &= \text{Translate}[O, \text{Vector}[M g_3]] \\ u &= \text{Vector}[O, U] \\ v &= \text{Vector}[O, V] \\ w &= \text{Vector}[O, W] \end{aligned}$$

#### (5) 세 개의 평면 생성

XY 평면, YZ 평면, ZX 평면을 생성하기 위해 Geogebra 입력창에 다음의 명령어를 순서대로 입력한다.

$$\begin{aligned} &[\text{구성 프로토콜: 14~16}] \\ XY &= \text{Polygon}[O + 5u + 5v, O - 5u + 5v, O - 5u - 5v, O + 5u - 5v] \\ YZ &= \text{Polygon}[O + 5v + 5w, O - 5v + 5w, O - 5v - 5w, O + 5v - 5w] \\ ZX &= \text{Polygon}[O + 5w + 5u, O - 5w + 5u, O - 5w - 5u, O + 5w - 5u] \end{aligned}$$

#### (6) 점의 생성과 2차원 벡터 구성

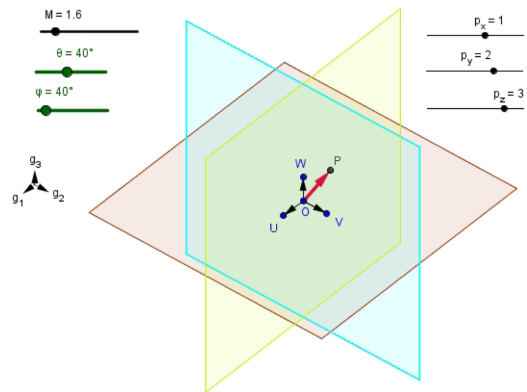
공간상의 한 점 P를 표시하기 위해 각 성분을 나타내는 세 개의 슬라이더를 만들어 점 P를 생성하고 원점과 연결해 2차원 벡터를 구성한다. 세 개의 슬라이더의 범위는 최소값 -5, 최대값 5로, 간격은 1로 설정한다. Geogebra 입력창에 다음의 명령어를 순서대로 입력한다.

$$\begin{aligned} &[\text{구성 프로토콜: 17~21}] \\ P &= O + p_x u + p_y v + p_z w \\ z &= \text{Vector}[O, P] \end{aligned}$$

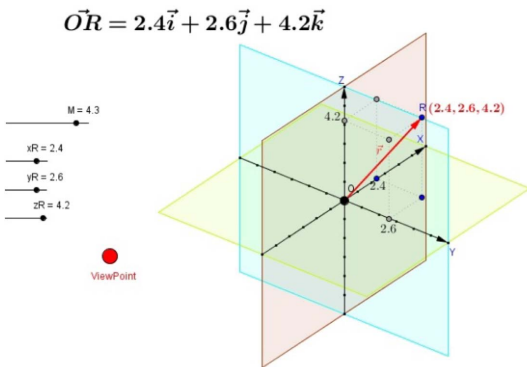
**Table 2** Geogebra Construction Protocol of 3D model using polar and azimuthal angle

No.	Name	Command
1	Angle $\theta$	
2	Angle $\varphi$	
3	Vector $g_1$	Vector[(-sin( $\varphi$ ), -cos( $\varphi$ )sin( $\theta$ ))]
4	Vector $g_2$	Vector[(cos( $\varphi$ ), -sin( $\varphi$ )sin( $\theta$ ))]
5	Vector $g_3$	Vector[(0, cos( $\varphi$ ))]
6	Point O	
7	Number M	
8	Point U	Translate[O, Vector[Mg <sub>1</sub> ]]
9	Point V	Translate[O, Vector[Mg <sub>2</sub> ]]
10	Point W	Translate[O, Vector[Mg <sub>3</sub> ]]
11	Vector u	Vector[O, U]
12	Vector v	Vector[O, V]
13	Vector w	Vector[O, W]
14	Quadrilatera l XY	Polygon[O+5u+5v, O-5u+5v, O-5u-5v, O+5u-5v]
14	Segment a	Segment[O+5u+5v, O-5u+5v, XY]
14	Segment b	Segment[O-5u+5v, O-5u-5v, XY]
14	Segment c	Segment[O-5u-5v, O+5u-5v, XY]
14	Segment d	Segment[O+5u-5v, O+5u+5v, XY]
15	Quadrilatera l YZ	Polygon[O+5v+5w, O-5v+5w, O-5v-5w, O+5v-5w]
15	Segment e	Segment[O+5v+5w, O-5v+5w, YZ]
15	Segment f	Segment[O-5v+5w, O-5v-5w, YZ]
15	Segment g	Segment[O-5v-5w, O+5v-5w, YZ]
15	Segment h	Segment[O+5v-5w, O+5v+5w, YZ]
16	Quadrilatera l ZX	Polygon[O+5w+5u, O-5w+5u, O-5w-5u, O+5w-5u]
16	Segment i	Segment[O+5w+5u, O-5w+5u, ZX]
16	Segment j	Segment[O-5w+5u, O-5w-5u, ZX]
16	Segment k	Segment[O-5w-5u, O+5w-5u, ZX]
17	Segment l	Segment[O+5w-5u, O+5w+5u, ZX]
18	Number $p_x$	
19	Number $p_y$	
20	Number $p_z$	
21	Vector z	Vector[O, P]

편각과 방위각을 슬라이더 객체로 조정하지 않고 시점 객체(Viewpoint object)를 만들어 시점 객체의 x 좌표를  $\varphi$ 로, 시점 객체의 y 좌표를  $\theta$ 로 사용할 수 있다. 시점 객체의 이름을 VP라고 한다면 지오지브라 입력창에서의 명령은 각각  $\varphi = x(VP)$ ,



**Fig. 8** Geogebra Model using Slider object



**Fig. 9** Geogebra model using Viewpoint object

$\theta = y(VP)$ 이다.

Fig. 8은 슬라이더 객체를 이용해서 편각과 방위각을 변화시키는 모델이고, Fig. 9는 시점 객체(Viewpoint object)를 이용한 모델이다. Table 2는 최종적으로 완성된 모델의 구성 프로토콜이다.

### 3. 3차원 모델의 구성 예

다음의 Fig. 10은 앞서 제시한 기저벡터의 구성 방법에 대한 강의를 듣고 인덕대학교 메카트로닉스과 학생들이 제출한 프로젝트 과제들의 일부이며, 지오지브라 수학공동체 웹사이트에서 실제 모델의 구성 과정을 확인해 볼 수 있다<sup>[7]</sup>. 지오지브라에서 제공하는 구성 프로토콜(construction protocol)을 통해 지오지브라 전문가나 다른 사용자가 구성한 수학 및 공학모델을 쉽게 분석해서 더 많은 응용모델들을 확장해서 만들어 낼 수 있다는 점에서, 지오지브라는 수학교육과 공학교육



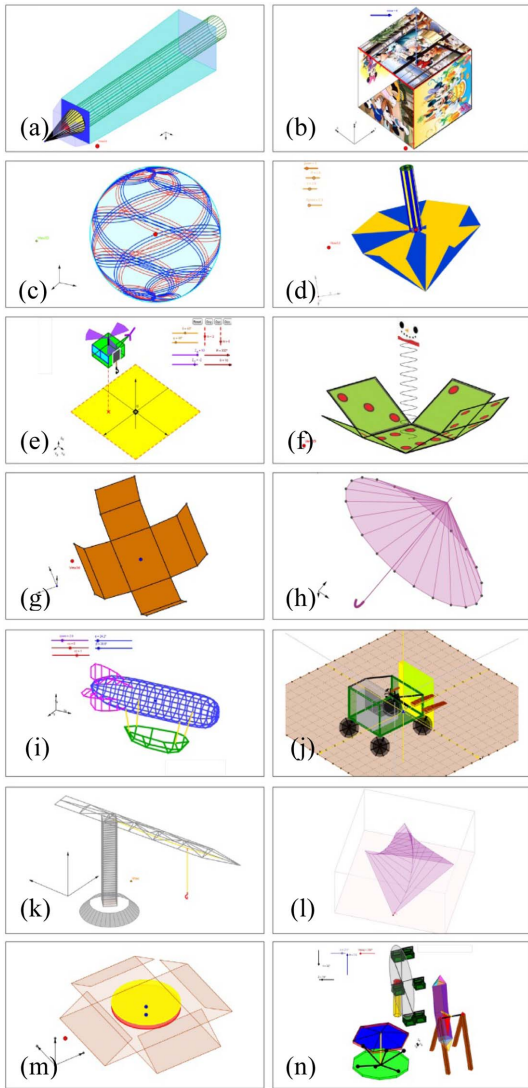


Fig. 10 Geogebra 3D model collection

적으로 생산성과 효율성이 큰 학습도구이다. Fig. 10에서 (a)는 공간 운동을 하면서 선을 긋는 3D 펜, (b)는 모든 방향에서 볼 수 있는 3D 캐릭터 큐브, (c)는 구슬의 무늬, 크기, 시점을 변경할 수 있는 3D 구슬, (d)는 지름, 높이, 넓이의 조절이 가능한 3D 팽이, (e)는 주 날개와 꼬리 날개가 회전하며 이동하는 헬리콥터, (f)와 (g)는 접고 펴지는 입체상자, (h)는 높이와 우산 살 개수를 조절할 수 있는 입체 우산, (i)는 입체 헬륨 비행선, (j)는 3D 지게차, (k)는 크레인, (l)은 삼각형을 기본으로 한 입체 프랙탈(fractal) 구조의 거미집, (m)은 입체 피자박스, (n)은 회전 놀이기구를 만든 프

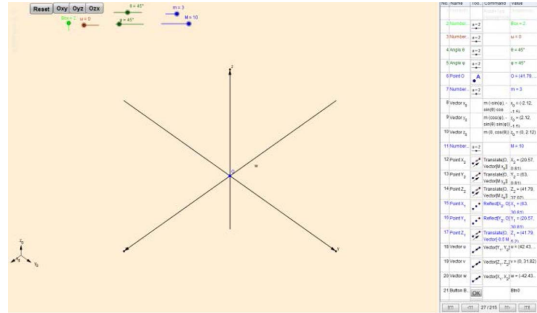


Fig. 11 Slider creation and construction of basal vectors

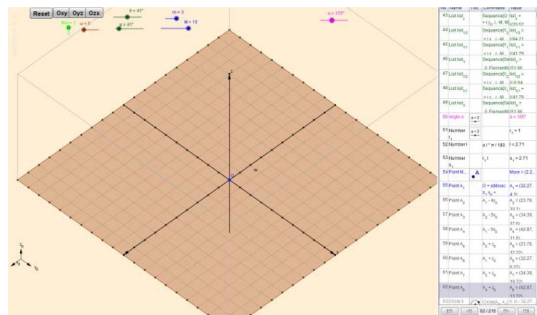


Fig. 12 Construction of 3 dimensional coordinate spaces

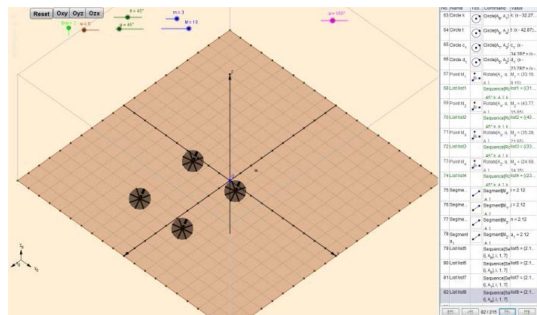


Fig. 13 Creation of initial positions for four wheels

로젝트 결과이다. 다음은 학생들이 제출한 프로젝트 과제물 중에서 3D 지게차 모델(j)을 단계별로 세분화한 구성 과정이다. 지게차 모델의 구성 프로토콜은 총 215 단계이며 편각과 방위각을 조절하기 위한 슬라이더 객체를 이용해서 3차원을 구현하였다. 처음 단계에서는 Fig. 11에서 보는 바와 같이 슬라이더를 생성하고 기저벡터를 구성한다. 이를 바탕으로 3차원 좌표 공간을 구성하는 모습이 Fig. 12에 나와 있다. 3차원 좌표 공간 상에 4개의 바퀴에 해당하는 초기 위치를 Fig. 13에서와 같이 설정한다.

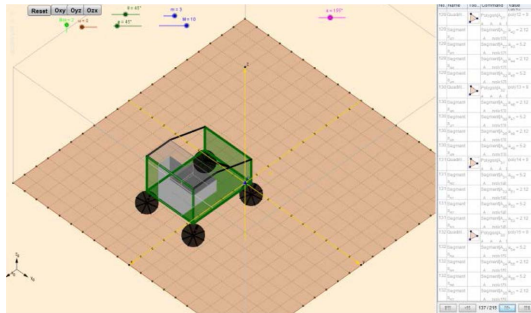


Fig. 14 Construction of the forklift base frames

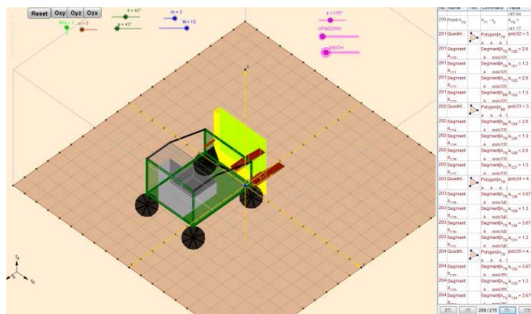


Fig. 15 Construction of the forklift assembly

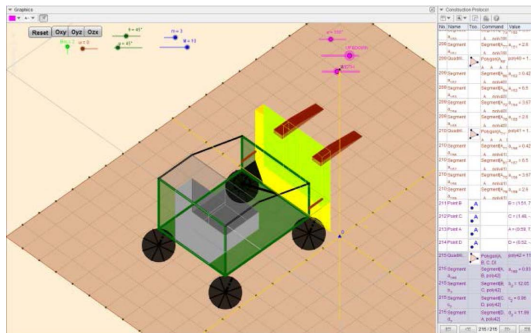


Fig. 16 Animation of the forklift motions using slider

지게차의 기본 프레임을 구성하는 모습이 Fig. 14에 나와 있으며, 포크와 마스트 등 부속장치를 구성하는 모습이 Fig. 15에 나와 있다.

슬라이더의 기능을 이용하여 포크를 상하 또는 좌우로 움직이면서 지게차의 작동을 구현하는 모습이 Fig. 16에 나와 있다. Fig. 16의 화면 우측에서 해당 작업에 대한 구성 프로토콜을 볼 수 있다.

Fig. 17에는 지오지브라에서 보여지는 1단계부터 최종 215 단계까지의 구성 프로토콜 중 처음 부분과 마지막 부분을 나타내었다.

2차원 기능의 지오지브라를 이용하여 지게차와

Construction Protocol					Construction Protocol				
No.	Name	Too...	Command	Value	No.	Name	Too...	Command	Value
1	Text text1		Rotate Text	rotatebox1	208	Segment		Segment[A <sub>150</sub> , A <sub>151</sub> ]	a <sub>150</sub> = 2.6
2	Number...	a = 2		Box = 2	209	Quadril...		Polygon[A <sub>151</sub> , A <sub>152</sub> , A <sub>153</sub> , A <sub>154</sub> ]	poly40 = 1...
3	Number...	a = 2		ω = 0	209	Segment		Segment[A <sub>152</sub> , A <sub>153</sub> ]	a <sub>152</sub> = 0.42
4	Angle θ	a = 2		θ = 45°	209	Segment		Segment[A <sub>153</sub> , A <sub>154</sub> ]	a <sub>153</sub> = 6.5
5	Angle φ	a = 2		φ = 45°	209	Segment		Segment[A <sub>154</sub> , A <sub>155</sub> ]	a <sub>154</sub> = 3.67
6	Point O			O = (41.79, ...)	209	Segment		Segment[A <sub>155</sub> , A <sub>156</sub> ]	a <sub>155</sub> = 2.6
7	Number...	a = 2		m = 3	210	Quadril...		Polygon[A <sub>156</sub> , A <sub>157</sub> , A <sub>158</sub> , A <sub>159</sub> ]	poly41 = 1...
8	Vector x <sub>0</sub>		m (-sin(φ), -sin(θ) cos	x <sub>0</sub> = (-2.12, -1.5)	210	Segment		Segment[A <sub>157</sub> , A <sub>158</sub> ]	a <sub>156</sub> = 0.42
9	Vector y <sub>0</sub>		m (cos(φ), -sin(θ) sin(φ))	y <sub>0</sub> = (2.12, -1.5)	210	Segment		Segment[A <sub>158</sub> , A <sub>159</sub> ]	a <sub>157</sub> = 6.5
10	Vector z <sub>0</sub>		m (0, cos(θ))	z <sub>0</sub> = (0, 2.12)	210	Segment		Segment[A <sub>159</sub> , A <sub>160</sub> ]	a <sub>158</sub> = 3.67
11	Number...	a = 2		M = 10	211	Point B		B = (1.51, 7...	
12	Point X <sub>2</sub>		Translate[O, Vector[M x <sub>0</sub> ]]	X <sub>2</sub> = (20.57, 0.81)	212	Point C		C = (1.48, ...	
13	Point Y <sub>2</sub>		Translate[O, Vector[M y <sub>0</sub> ]]	Y <sub>2</sub> = (63, 0.81)	213	Point A		A = (0.59, 7...	
14	Point Z <sub>2</sub>		Translate[O, Vector[M z <sub>0</sub> ]]	Z <sub>2</sub> = (41.79, 37.02)	214	Point D		D = (0.52, ...	
15	Point X <sub>1</sub>		Reflect[X <sub>2</sub> , O]	X <sub>1</sub> = (63, 30.81)	215	Quadril...		Polygon[A, B, C, D]	poly42 = 11...
16	Point Y <sub>1</sub>		Reflect[Y <sub>2</sub> , O]	Y <sub>1</sub> = (20.57, 30.81)	215	Segment		Segment[A, B, poly42]	a <sub>160</sub> = 0.93
17	Point Z <sub>1</sub>		Translate[Z <sub>1</sub> , Z <sub>2</sub> ]	Z <sub>1</sub> = (41.79, 5.2)	215	Segment		Segment[B, C, poly42]	b <sub>2</sub> = 12.05
18	Vector u		Vector[Y <sub>1</sub> , Y <sub>2</sub> ]	u = (42.43, ...)	215	Segment		Segment[C, D, poly42]	c <sub>2</sub> = 0.96
19	Vector v		Vector[Z <sub>1</sub> , Z <sub>2</sub> ]	v = (0, 31.82)	215	Segment		Segment[D, A, poly42]	d <sub>2</sub> = 11.99
20	Vector w		Vector[X <sub>1</sub> , X <sub>2</sub> ]	w = (-42.43, ...)					
21	Button B...		OK	Btn0					

Fig. 17 Construction protocol from step 1 to step 215

같은 3차원 구조물의 기구학적 모델링을 수행함으로써, 직교좌표계의 구성과 3차원 구조물의 구성을 학생들이 직접 실습할 수 있었으며, 슬라이더 기능을 이용하여 해당 기구의 동작 특성을 익힐 수 있었다.

지금까지 수학 교육 현장에서는 지오지브라가 주로 도형이나 함수의 궤적을 구현하는데 활용되었으나, 3차원 구조물의 기구학적 저동을 구현하는데도 활용될 수 있었으며, 구성 프로토콜의 작업 순서를 따라 가면서 학생들이 기존의 작업을 따라가거나 구성 프로토콜을 추가하면서 공간 운동을 구현할 수 있는 교육적 효과가 있음을 확인할 수 있었다.

#### 4. 결 론

본 연구에서는 기하(algebra)와 대수(geometry)

의 상호작용으로 움직이는 수학을 구현할 수 있는 수학 교육용 소프트웨어인 지오지브라(Geogebra)를 활용해서 2차원에서 3차원 모델을 구성하기 위해 세 개의 2차원 기저벡터를 구성하는 여러 가지 기법들을 정리하였다. 이러한 기법들을 이용해서 2차원 직교 좌표계에서 임의의 위치에 3차원 모델을 표현하기 위한 3개의 기저벡터들을 구성한 후에 이 벡터들을 기준으로 점, 선, 면 등의 기본적인 수학적 객체(mathematical object)들을 이용해서 3차원 벡터를 표현하였다. 이러한 3차원 벡터의 표현원리를 3차원 구조물에 적용해서 모델링하고 지오지브라의 핵심 객체인 슬라이더 도구를 이용해 평행이동, 회전이동, 확대, 축소 등의 다양한 애니메이션을 구현한 실제 모델의 예를 살펴 보았다.

본 연구를 통해 수학적 원리의 시각화에 도움을 주는 수학학습 도구인 지오지브라가 기구학적 모델링이나 3차원 CAD의 영역에서 기하학적 의미의 이해와 학습 효과를 극대화시키는 데 효율적인 도구임을 확인할 수 있었다.

### 감사의 글

본 연구는 2014학년도 인덕대학교 학술연구비에 의해 수행된 과제임을 밝힙니다.

### References

1. Kim, T.H. and Kim, H.C., 2013, *Principle of Geogebra*, Kyowoo-sa, Seoul, pp.2-3.
2. <http://www.geogebra.org/cms/en>
3. Yang, S.H. and Kang, O.K., 2011, Instruction Method for Quadratic Curve Based on Dynamic Visual Representation by Applying Geogebra, *Journal of Korea Society of Educational Studies in Mathematics*, 13(40), pp.447-468.
4. Kim, P., 2013, Rigid Body Dynamics for Beginners: Euler Angles & Quaternions, CreateSpace Independent Publishing Platform.
5. [http://en.wikipedia.org/wiki/Euler\\_angles](http://en.wikipedia.org/wiki/Euler_angles)
6. [http://en.wikipedia.org/wiki/Spherical\\_coordinate\\_system](http://en.wikipedia.org/wiki/Spherical_coordinate_system)
7. <http://www.geogebra.kr>



#### 정 태 은

1986년 서울대학교 기계공학과 학사  
 1988년 서울대학교 기계설계학과 석사  
 1993년 서울대학교 기계설계학과 박사  
 1993년~1999년 고등기술연구원 책임연구원  
 1999년~현재 인덕대학교 메카트로닉스과 교수  
 2007년~2008년 University of British Columbia, Visiting Associate Professor  
 관심분야: CAD/CAM Education, Biomechanics, Human Modeling



#### 김 태 환

1995년 서울대학교 기계설계학과 학사  
 2002년 서울대학교 기계항공공학부 석사  
 2005년 서울대학교 기계항공공학부 박사과정 수료  
 1995년~1998년 대우종합기계 연구원  
 2002년~현재 인덕대학교 메카트로닉스과 겸임교수  
 2010년~2012년 (주)스콜통 대표  
 2012년~현재 (주)에듀프레소 대표  
 관심분야: CAD/CAM Education, CAD/CAE 및 전산설계