교육과정평가연구
The Journal of Curriculum and Evaluation
2008, Vol. 11, No. 1, pp. 187~204

거북 마이크로월드에서 곡선의 행동 표현[1)]

김 화 경(상명대학교 전임강사)

≪ 요 약 ≫

이 글은 곡선을 LOGO 명령으로 구성하는 방법과 그 교육적 의미를 논의한다. LOGO 명령으로 곡선을 표현하는 것은 곡선을 내재적, 동적으로 표현하는 것이다. LOGO의 기본 명령 '가자', '돌자'는 평면기하 현상을 설명하기 위한 기본적인 명령으로 평면기하 현상은 이들의 조합으로 나타낼 수 있다. 쉽고 직관적인 이 명령으로 도형을 표현하는 활동은 도형의 구성 절차를 강조하는 일이다.

이 논문에서 평면기하 현상을 표현하는 기본 명령인 '가자', '돌자'를 이용하여 중등 교육과정 이후에 등장하는 곡선을 표현하는 방법을 모색한다. 곡선을 '가자', '돌자'로 표현하는 활동은 정적인 곡선을 동적으로 표현하는 활동이며, 같은 것에 대한 다른 표현들 사이의 번역 활동이다. 우리는 중간 매개자를 이용하여 번역 방법을 모색하고, 이러한 번역 방법을 교육적으로 활용할 수 있는 방안을 모색한다.

주제어 : 마이크로월드, 거북기하, 로고, 컴퓨터와 수학교육, 자비말

Ⅰ. 들어가며

Tayler(1980)는 교육에서 컴퓨터의 역할을 '개인교사형(tutor)', '보조도구형(tool)', '학생주도형(tutee)'의 세 가지 형태로 나누었다. 이 중 '개인교사형'이나 '보조도구형' 관점은 학생의 교과 학습 향상을 위해 컴퓨터를 이용하는 관점으로, 교사나 학생이 반드시 컴퓨터에 능통할 필요는 없다. 이에 비해 '학생주도형' 관점은 학습자가 컴퓨터에게 이야기를 해야 하고 프로그래밍을 해야 하므로 학습자는 컴퓨터에 익숙해야 한다. 또한 '학생주도형' 관점에서는 특정 수학 내용 학습보다는 수학적 사고력 교육이 더 강조된다. Papert(1980)는 평면기하 현상을 기본 명령의 조합으로 파악하고 학생들이 컴퓨터 화면에 그림을 그리는 과정에서 수학을 사용하고 학습이 일어나기를 기대하는 컴퓨터 환경으로 LOGO를 설계・구현하였다. 학습

1) 본 연구는 2008학년도 상명대학교 자연과학연구소 연구비 지원에 의해 수행되었다.

자는 LOGO에서 ‘거북이(turtle)’를 움직이고 거북 행동의 결과를 자취로 나타내어 자신이 그리고자 하는 그림을 그린다. 학습자는 거북이 등 위에 올라타서 행동의 방향과 크기를 거북이에게 명령으로 알려주어야 한다. LOGO는 학습자가 거북이를 가르치는, ‘학생주도형’으로 분류되는 환경으로 1980년대를 대표하는 교육용 컴퓨터 환경이다. LOGO의 기본 명령은 쉽고 직관적이어서 어린 학습자에게 쉬웠으며, 학습자가 원하는 그림을 그리면서 수학을 사용하고 학습한다는 철학을 가지고 있다.

Abelson과 diSessa(1980)는 좌표기하(coordinate geometry)와 비교하여 LOGO에서 이루어지는 기하를 ‘거북기하(turtle geometry)’라고 이름 붙이고 거북기하의 특징을 ‘내재적(intrinsic)’, ‘국소적(local)’, ‘절차적(procedural)’으로 나누어 설명하고 있다. 먼저 ‘내재적’이란 도형의 본질적인 특징으로만 도형을 구성한다는 의미로, 도형을 구성하는 데 비본질적 외적 요소를 사용하지 않는다는 의미이다. 또한 ‘국소적’이란 시간에 따른 순간적 행동을 만들고 그 행동의 결과로 전체적 그림 모양을 완성한다는 의미로, 조건을 만족하는 점들의 집합으로 ‘전체적’ 모양이 결정되는 좌표기하와 다르다는 의미이다. 마지막으로 ‘절차적’이란 단계별 구성 방법이 제시되어 그에 많은 변화를 줄 수 있다는 뜻이다. 예를 들어 LOGO에서 원을 그려보자. 거북기하에서 원을 그리려면 거북이에게 ‘조금 가고(FORWARD; FD; 가자), 조금 돌고(ROTATE; RT; 돌자)’를 무한히 반복하도록 명령해야 한다. 이에 비해 좌표기하에서는 방정식 $x^2+y^2=r^2$을 만족하는 점들의 집합이 원이다. 거북기하는 좌표축이 필요 없지만, 좌표기하에서는 원과 관계없는 비본질적인 좌표축이 있어야 한다. 또 거북기하는 전체적인 원 모양이 아니라 시간에 따른 국소적 행동, 원을 구성하는 절차를 나타내지만 원의 방정식은 전체적 모양을 나타낸다. 그리고 거북기하에서 원을 그리는 명령은 원을 만드는 절차를 통해 완성된 원의 결과를 시각적으로 확인하게 된다. 그러나 좌표기하에서는 원의 방정식이 중요하다. 거북기하는 원이라는 도형의 구성이 강조되며, 좌표기하는 도형의 분석에 유용하다.

직관적 명령 체계로 비교적 이른 시기에 도입되어 수학교육에 이용될 수 있는 컴퓨터 환경인 LOGO는 나라별로 여러 형태로 실제 구현되어 사용되고 있다. 우리나라의 경우 초등학교 3학년 수학 익힘책(교육인적자원부, 2002a)과 4학년 수학 익힘책(교육인적자원부, 2002b)에 ‘거북 명령 프로그램’이라는 이름으로 소개되어 있고, 인터넷 애플릿 환경으로 구현되어 있다.[2] 정다각형 같은 기본 도형을 그리는 활동과 반복, 변수 사용, 함수 개념 등을 이용하여 보다 복잡한 도형을 그리는 활동에 ‘거북 명령 프로그램’이 이용되고 있다.

LOGO에서 그려지는 그림을 수학적 곡선[3]으로 이해할 수도 있다. 곡선은 조건을 만족하

2) 현재 http://web.edunet4u.net/~javamath라는 인터넷 주소를 통해 ‘거북 명령 프로그램’을 인터넷으로 이용할 수 있도록 하고 있고, 수정된 환경은 ‘JavaMAL(http://www.javamath.com)’에서 접근할 수 있다.

3) 곡선은 1차원 공간에서 2차원 공간으로의 연속 사상을 말한다(Insall, Matt & Weisstein, Eric W. “Curve.” From MathWorld－A Wolfram Web Resource. http://mathworld.wolfram.com/Curve.html).

는 ‘1차원 집합’이라는 정적인 의미와 함께 시각에 따라 변하는 점이라는 동적인 의미를 함께 가진다(김홍종, 2004). 정적인 의미에서 곡선은 일정한 관계를 만족시키는 점들의 집합이며 동적인 의미에서는 시각에 따른 행동의 자취이다. 정적인 의미에서 같은 곡선이라 하더라도 시간의 흐름을 달리하면 동적인 의미에서는 다른 행동일 수 있다. 시간에 따른 행동은 곡선 구성 방법이고, 점들 집합으로의 곡선은 곡선 분석 방법이다.

LOGO에서 원을 그리는 거북 행동 명령은 시간에 따른 절차가 있는 곡선의 동적 표현이다. 이에 비해 학교수학에 나오는 곡선은 대부분 정적으로 표현되어 있다. 이와 같이 학교수학에서 정적 표현만 제시되는 것은 수학적・역사적 결과이다. 역사적으로 동적・기하적 의미를 가지던 곡선에 관한 연구는, 수학적 확실성을 도모하는 과정에서 정적・대수적 표현으로 발전하게 되었고, 그 결과가 학교수학에 반영된 것이다. 한편 종이와 연필 환경에서는 곡선을 동적으로 표현하는 데에 한계가 있기 때문에 정적 표현만이 제시되어 있기도 할 것이다. 그러나 변화하는 세상에 살고 있는 우리는 동적인 상황에 처해 있으며, 곡선을 현실과 연결시키기 위해서는 동적인 표현이 필요하다. 즉 곡선을 두 가지 방식으로 모두 표현할 수 있어야 곡선에 대한 진정한 이해를 기대할 수 있다. 여러 동적 상황의 결과로 정적 관계식이 도입될 때 곡선의 의미가 더 풍부해질 수 있다. 거북 행동으로 곡선을 동적으로 표현할 수 있다면 곡선에 관한 이해를 넓히는 기회가 될 것이다.

이 글에서 우리는 초등학교 수학 익힘책에 제시된 거북 명령 프로그램(LOGO)의 기본 명령인 ‘가자’, ‘돌자’만을 이용해 곡선을 동적으로 표현하는 방법과 의미를 모색하려 한다.

Ⅱ. 이론적 배경과 마이크로월드

구성주의자들에게 지식은 한쪽에서 다른 쪽으로 일방적으로 전달되는 상품 같은 것이 아니라 학습자에 의해 적극적으로 재구성되는 경험이다. Ackermann(2004)은 이런 구성주의자들의 관점을 다시 Piaget, Papert 그리고 Vygotsky식으로 구별하고 있다. 먼저 Piaget는 합리주의자로 인간의 인지 발달 과정을 규명하려고 노력하면서 가르치는 행위는 반드시 간접적이며, 지식과 정보는 다르며, 아동의 심리적 저항을 무시하는 교육이론은 실패한다고 주장하였다. 이러한 그의 주장은 구성주의자들의 기본 가정이 되었다. 그러나 서로 다른 발달 단계 아동들의 공통적 특징을 설명하려던 Piaget는 개인적 선호도나 문맥, 도구의 역할에 크게 주목하지 않았다. 이에 비해 실제적인 지식 구성을 강조한 직관주의자인 Papert는 지식의 구성에서 ‘도구(media)’의 역할을 중요시 여겼으며, 사회-문화주의자인 Vygotsky는 개인을 포함하는 ‘사회(others)’의 역할을 중요하게 생각했다(Ackermann, 2004). 특히 지식의 구성에서 도구의 역할을

강조하여 '물리적 구성을 통한 정신적 구성'을 주장한 Papert 이론을 구성주의(constructivism)와 구별하여 constructionism이라고 한다.[4] Papert는 물리적 구성을 강조하고 실제 물리적 구성이 일어날 수 있는 바람직한 환경을 설계하고, 이를 통해 지식이라는 정신적 구성을 실현하려고 노력했다. 이런 점에서 constructionism은 Piaget식의 구성주의의 실천적 방법론으로 볼 수 있다.

> Constructionism은 학습이론이면서 교육을 위한 전략이다. 지식은 단순히 교사에서 학습자로 전달되는 것이 아니라 학습자의 마음속에서 활발하게 다시 구성되는 것이라는 Piaget식 구성주의자(constructivist)의 관점을 취하고 있다. 아동들은 아이디어를 얻는 것이 아니라 만드는 것이다. 나아가, constructionism은 학습자들이 외부의 인공물들(로봇, 시, 모래성, 컴퓨터 프로그램)을 활발하게 만들 때, 새로운 아이디어를 얻을 수 있을 것이라고 말하고 있다. 그래서 constructionism은 개인적으로 의미 있는 인공물을 구성하는 상황 속에서 지식 구성이라는 두 종류의 구성이 관계된다(Kafai & Resnick, 1996, p. 1).

Constructionism 관점에서 수학교육을 위한 컴퓨터 환경을 설계하고 구현하는 연구 분야가 '컴퓨터와 수학교육(조한혁, 2003; 김화경, 2006)'이다. 이는 교수(teaching)보다는 학습(learning)을 강조하는 구성주의자의 입장에서 물리적 구성을 강조하는 constructionism 관점을 더해 도구, 컴퓨터와 함께 수학 학습을 추구한다.

'컴퓨터와 수학교육'이 이루어지는 환경, 수학교육을 위한 constructionism 구현을 위해 설계된 환경을 우리는 '마이크로월드(microworld)'라고 부른다. 초기에 마이크로월드라는 개념은 실제 실험이 어려울 때, 가상 실험을 하기 위한 컴퓨터 환경이라는 의미로 사용되었다. 이후 마이크로월드는 수학 실험을 위한 공간인 동시에 '강력한 아이디어(Bers, 2001; Papert, 1980)'가 내재하는 컴퓨터 환경이라는 의미로 바뀌었다. 즉 마이크로월드는 '직접적'으로 수학을 가르치려는 목적에서 설계된 환경이 아니며, 오히려 수학적 원리로 '기본 명령'이라는 구조를 설계한 환경이다. 학습자는 마이크로월드에서 자신이 할 수 있는 것과 없는 것을 이해하고 한계를 테스트한다. 그리고 할 수 있는 것을 조합하여 무언가를 구성하고 그 과정에서 지식을 구성한다. 따라서 마이크로월드를 설계할 때 가장 중요한 것은 적절한 '기본 명령'을 정하는 일이다.

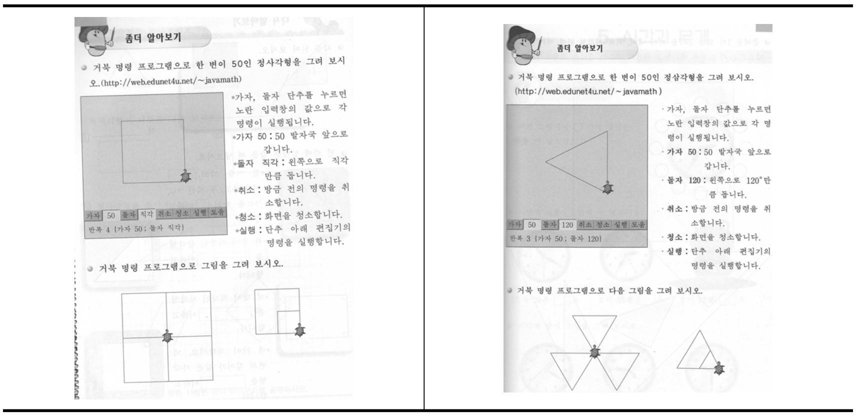
바람직한 마이크로월드는 간단한 '기본 명령'으로부터 아주 많은 것을 만들 수 있는 환경이다(Resnick & Silverman, 2005). 가장 대표적인 마이크로월드인 LOGO는 평면기하 현상을 '가자', '돌자'의 조합으로 파악하고 이 두 명령을 기본 명령으로 삼았다. Cho 등(2007)과 김화경, 송민호(2007)는 '가자', '돌자'의 거북 행동을 벡터 관점에서 이해하여 새로운 '기본 명령'인 행동 벡터 명령 'move'[5]를 제안하고 있다. 그리고 거북 머리 방향을 'move a, b' 방향

---

4) Papert는 교수주의(instructionism)에 대비되는 용어로 constructionism이라는 용어를 사용한다.

5) 'move a, b'라는 명령은 거북이를 가로축으로 a만큼, 세로축으로 b만큼 움직이는 명령이다. 만약 현

으로 바꾸는 명령 'head a, b'의 필요성도 살펴보고 있다. 이와 같이 LOGO를 포함하여 거북이(고양이)를 움직이고 그 결과가 시각적으로 제시되는 환경을 우리는 거북 마이크로월드라고 한다. 그 중 하나가 이 글에서 사용하는 JavaMAL 환경이다.



〔그림 1〕 초등학교 수학 익힘책

〔그림 1〕은 초등학교 수학 익힘책에 나오는 거북 명령 프로그램과 학습 활동이다. 이 환경을 인터넷 게시판과 연결하고 'move'를 비롯한 몇 가지 주요한 기본 명령을 추가하여 JavaMAL 마이크로월드가 설계·구현되었다(조한혁, 2003; http:www.javamath.com). 이 환경은 거북 마이크로월드 환경인 동시에 동적 기하 환경(DGS; ynamic geometry system)이 통합된 환경이다(김화경, 2006). 이제 이 환경에서 거북 마이크로월드의 기본 명령인 '가자', '돌자'를 이용해 여러 곡선을 만드는 방법을 살펴보자.

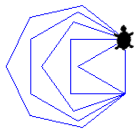
Ⅲ. 곡선의 거북 행동 표현

초등학교 수학 익힘책에는 정사각형과 정삼각형 행동을 만드는 명령에서 시작하여 '가자' 값과 초기 위치를 변형하여 길이가 다른 정사각형과 정삼각형, 위치가 다른 정사각형과 정삼각형을 그리는 명령을 만들어보는 활동이 제시되어 있다. 이 명령에서 '돌자' 값을 변형하여 여러 가지 정다각형 행동을 만드는 명령을 얻을 수 있다. 〈표 1〉[6)]은 변수 n을 3부터 7까

재 거북이의 위치가 (0, 0)이었다면, 'move a, b'의 명령을 하면 거북이의 위치는 (a, b)가 된다. 'move'는 좌표를 이용하고 있으므로 한편으로는 거북기하의 성질을, 다른 한편으로는 좌표기하의 성질을 가진다. 그러나 절차적 특징으로 곡선을 동적으로 표현하는 데 이용될 수 있다.

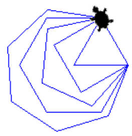
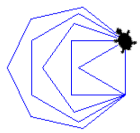
지 변화시키면서 정n각형을 그리는 절차를 나타내는 명령과 그 결과이다. 이 명령은 〈표 1〉과 같이 삼각함수를 이용해 'move' 명령으로 나타낼 수 있다. 서로 다른 두 가지 표현으로 그린 그림은 〈표 1〉과 같이 같다. 정말 같은 결과일까?

〈표 1〉 거북 명령으로 정다각형 그리기

'가자', '돌자' 명령	결과	'move' 명령
for n=3 to 7 for m=1 to n 돌자 $\frac{360}{n}$; 가자 30 next next		for n=3 to 7 for m=1 to n move $-30\cdot\sin\frac{360m}{n}$, $30\cdot\cos\frac{360m}{n}$ next next

만약 초기 거북이의 머리 방향이 30°만큼 돌아간 상태에서 두 명령을 실행한다면, 〈표 1〉의 '가자', '돌자' 명령의 실행 결과와 'move' 명령의 실행 결과는 초기 상태만큼 차이가 있게 된다. 〈표 2〉는 그 결과를 나타낸다. 〈표 2〉에서 '가자', '돌자'의 거북 명령과 'move' 명령을 실행하기 전의 거북의 초기 상태는 같지만 그 결과는 서로 다르다. 이는 '가자', '돌자' 명령은 외부의 조건에 영향을 받지 않는 '내재적' 명령이지만, 'move' 명령은 가로축과 세로축이라는 외부의 조건에 종속되기 때문이다. 즉, 내재적 명령 체계는 초기 상태에 영향을 받지만 'move' 명령은 거북이의 초기 상태와 무관하게 항상 같은 결과를 만들어낸다. 초기 상태와 결과가 연결된다는 측면에서 '내재적' 특징이 더 실제 운동과 유사하고 자연스럽다.

〈표 2〉 거북 명령의 내재적 특징

명령	초기 상태	결과
for n=3 to 7 for m=1 to n 돌자 $\frac{360}{n}$; 가자 30 next next		
for n=3 to 7 for m=1 to n move $-30\cdot\sin\frac{360m}{n}$, $30\cdot\cos\frac{360m}{n}$ next next		

6) 실제 JavaMAL에서는 $360/n(=\frac{360}{n})$, Sin(x)$(=\sin x)$, $a*b$ $(=a\cdot b)$와 같이 표기한다.

다시 초기 상태를 북쪽(초기 상태)으로 고정하고 두 명령 체계에 주목해 보자. 이 두 명령 체계는 모두 평면 위의 거북 행동을 나타내는 명령이므로 두 체계 사이의 번역을 생각해 볼 수 있다. 〈표 1〉은 같은 결과에 다른 표현이며 이들 표현 사이의 번역은 삼각함수를 이용해 가능하다. 먼저 '가자', '돌자'로 만들어지는 거북 행동을 'move' 명령으로 번역하는 방법을 생각해 보자. 임의의 거북 행동(v)는 다음과 같은 형태로 나타낼 수 있으며 이를 거북 행동의 표준형이라고 하자.7)

$$v = \text{돌자 } r_1;\ \text{가자 } f_1;\ \text{돌자 } r_2;\ \text{가자 } f_2;\ \cdots;\ \text{돌자 } r_n;\ \text{가자 } f_n;$$

거북이가 초기 상태(머리 방향 북쪽)라고 가정하고 거북 행동(v_i)을

$$v_i = \text{돌자 } \sum_{j=1}^{i} r_j;\ \text{가자 } f_i;$$

로 정의하고 이를 위치 벡터로 표현하면

$$v_i = \text{돌자 } \sum_{j=1}^{i} r_j;\ \text{가자 } f_i; = \left(-f_i \cdot \sin\sum_{j=1}^{i} r_j,\ f_i \cdot \cos\sum_{j=1}^{i} r_j\right)$$

와 같다. 다시 이를 행동 벡터 'move' 명령으로 표현하면

$$v_i = \text{move } -f_i \cdot \sin\sum_{j=1}^{i} r_j,\ f_i \cdot \cos\sum_{j=1}^{i} r_j$$

이다. 따라서 전체 거북 행동(v)는 다음과 같이 'move' 명령으로 나타낼 수 있다.

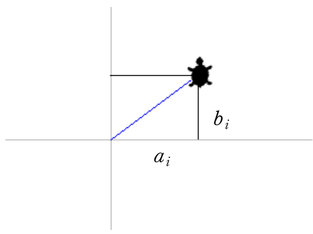
$$v = \sum_{i=1}^{n} v_i = \sum_{i=1}^{n} \left(\text{move } -f_i \cdot sin\sum_{j=1}^{i} r_j,\ f_i \cdot cos\sum_{j=1}^{i} r_j\right).$$

즉, 임의의 '가자', '돌자'의 거북 행동은 삼각함수를 이용해 'move'로 나타낼 수 있다.

반대로 'move' 명령을 '가자', '돌자'로 번역하는 방법을 생각해 보자. 일반적으로 'move' 명령(v)가

$$v = \text{move } a_1, b_1;\ \text{move } a_2, b_2;\ \cdots;\ \text{move } a_n, b_n;$$

으로 나타낼 수 있다. 이때, i 번째 행동 벡터 'move a_i, b_i'는 〔그림 2〕에서



〔그림 2〕 거북 상태

$$\text{move } a_i, b_i = \text{돌자 } s_i;\ \text{가자 } \sqrt{a_i^2 + b_i^2}\ (\text{단, } \tan s_i = -\frac{a_i}{b_i})$$

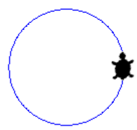
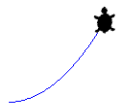
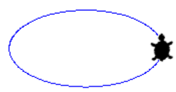
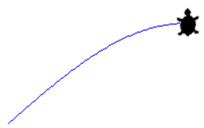
임을 알 수 있다. 따라서 전체 행동 벡터 v는 다음과 같다.

$$v = \sum_{i=1}^{n} move\, a_i, b_i = \sum_{i=1}^{n} \text{돌자} s_i; \text{가자} \sqrt{a_i^2 + b_i^2}; \text{돌자} -s_i \qquad (\text{단, } \tan s_i = -\frac{a_i}{b_i})$$

7) 만약 '가자' 혹은 '돌자'가 연속으로 나온다면 이는 하나로 묶일 수 있으므로 모든 거북 행동은 표준 표현으로 나타낼 수 있다.

이때, $s_i = arctan\left(-\frac{a_i}{b_i}\right)$ 8)으로 'move' 명령을 '가자', '돌자' 명령으로 바꾸는 것이 그 반대의 경우보다 조금 번거롭다. 삼각함수와 역삼각함수의 값을 이용해야 할 때 컴퓨터 계산 기능은 큰 도움이 될 수 있다. 예를 들어 포물선은 세로축 변화량이 선형적으로 증가하는 경우이므로 <표 3>과 같이 'move 1, n/30'으로 얻어지고 이를 '가자', '돌자' 명령으로 나타낼 수 있다. <표 3>은 몇 가지 곡선을 거북 명령으로 그린 것이다.

〈표 3〉 거북 행동으로 곡선 그리기

	move	가자, 돌자	
원	for n=1 to 360 move $-\sin n,\ \cos n$ next	for n=1 to 360 돌자 n; 가자 1; 돌자 -n (돌자 1; 가자 1) next	
포물선	for n=1 to 50 move 1, $\frac{n}{30}$ next	for n=1 to 50 돌자 $arctan(-\frac{30}{n})$; 가자 $\sqrt{\frac{900+n^2}{900}}$; 돌자 $-arctan(-\frac{30}{n})$; next	
타원	for n=1 to 90 move $-\sin n,\ 0.5\cdot\cos n$ next	for n=1 to 90 돌자 $arctan(2\cdot\tan n)$; 가자 $\sqrt{\sin^2 n+0.25\cos^2 n}$; 돌자 $-\arctan(2\cdot\tan n)$; next	
$y=\sin x$ 그래프	for n=1 to 90 move 1, $\sin(n)-\sin(n-1)$ next	for n=1 to 90 $k=\sin(n)-\sin(n-1)$ 돌자 $arctan\left(-\frac{1}{k}\right)$; 가자 $\sqrt{1+k^2}$; 돌자 $-arctan\left(-\frac{1}{k}\right)$; next	

8) 거북이의 초기 상태는 '북쪽'을 바라보며, 반시계 방향이 각의 양(+)의 방향이다. 이로 인해 조금 혼동이 있을 수 있다. 이러한 혼동을 피하려면 초기 상태의 거북이를 먼저 '돌자 -90' 명령으로 돌려놓는 것도 하나의 방법이다. 이때, 삼각함수와 역삼각함수의 값을 구할 때는 컴퓨터의 계산 기능을 이용하게 되므로 학습자는 계산에서 조금은 자유로워질 수 있다.

우리는 '가자', '돌자'의 명령과 'move' 명령 사이의 번역을 생각해 보았다. 이때, 'move' 명령은 외재적이지만 '가자', '돌자'는 내재적이다. 또 'move' 명령은 좌표기하와 잘 연결되며 '가자', '돌자' 명령은 직관적 행동 표현이다. 두 명령 체계는 같은 것에 대한 서로 다른 표현 체계이며 각자의 체계에서 잘 그릴 수 있는 곡선이 있다. 예를 들어 원은 '가자', '돌자'의 명령 체계에서 더 쉽게 만들 수 있다. 몇 가지 나선도 마찬가지이다. 이에 비해, 포물선, 타원, 사이클로이드 같은 곡선은 'move' 명령으로 그리는 것이 쉽다. 〈표 3〉에서 포물선과 타원은 'move' 명령으로 먼저 그린 후 이를 '가자', '돌자'로 번역한 것이고, 또 원은 그 반대이다.

연속 함수 $y=f(x)$의 그래프도 같은 방식으로 그릴 수 있다. 함수의 그래프[9]는 좌표축이 존재하는 상황에서 함수식을 만족하는 점들의 집합을 나타낸 것이므로, 'move' 명령을 이용하는 것이 수월하다. 먼저 함수의 그래프 위의 점들

$$(0, f(0)),\ (1, f(1)),\ (3, f(3)),\ \cdots,\ (n, f(n)),\ \cdots$$

을 연결하는 벡터 $v_n=(n, f(n))-(n-1, f(n-1))=(1, f(n)-f(n-1))$을 이용해, 'move' 명령으로 나타내고

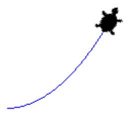
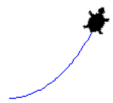
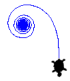
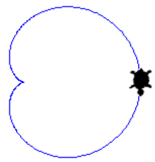
move 1, $f(1)-f(0)$; move 1, $f(2)-f(1)$; …; move 1, $f(n)-f(n-1)$; …,

이를 '가자', '돌자' 명령으로 번역할 수 있다.

몇 가지 곡선을 '가자', '돌자' 명령으로 그리는 방법에 대해 살펴보았다. 물론 이런 방법으로 모든 곡선을 그릴 수 있는 것은 아니며 정확한 곡선을 그릴 수 있는 것도 아니다. 그러나 이 명령은 정적으로 표현된 곡선을 움직이는 동적 표현으로 번역할 수 있다는 점에서 의의를 갖는다. 또한 곡선을 그리기 위해 등장하는 여러 가지 계산은 컴퓨터에게 맡기고 곡선을 그리는 절차에 집중할 수 있다는 점이 장점이다. 그러나 〈표 3〉의 '가자', '돌자' 명령이 거북이의 자연스러운 행동을 나타내지는 못한다. 한 절차에서 '돌자' 명령을 두 번 사용하고 있기 때문이다. 앞선 명령에서 돌자 s_i; 가자 $\sqrt{a_i^2+b_i^2}$; 돌자 $-s_i$는 거북 머리를 두 번 돌리는 결과를 초래한다. 즉, 곡선의 자연스러운 행동 표현을 위해서는 순간의 행동 방향이 접선 방향이어야 한다. 〈표 3〉에서 우리는 원하는 곡선 결과를 얻었지만 자연스러운 곡선 행동을 얻지는 못했다.

9) 곡선 중에는 함수의 그래프가 아닌 것도 존재한다. 반면 연속 함수의 그래프는 곡선이다. 이 글은 끊어지지 않는 곡선을 다루고 있어 일반적 함수의 그래프보다 연속 함수의 그래프만을 고려한다. 또한 〈표 3〉에서는 함수의 그래프는 $x\geq 0$인 경우만을 다루고 있다. 이는 함수의 그래프를 x축의 방향으로는 단위 시간에 1만큼 움직일 때 y축 방향으로의 변화량에 따른 거북이의 자취로 파악하기 때문이다. 이는 김화경(2006)에서 일정한 속도로 x축 방향으로 흐르는 강물 위에서 y축 방향으로 헤엄치는 거북이의 자취로 함수 그래프를 도입하는 것과 같은 맥락이다.

〈표 4〉 거북 행동으로 곡선 그리기

	move	가자, 돌자	
포물선	for n=1 to 50 move 1, $\frac{n}{30}$ next	돌자 -90; for n=1 to 50 돌자 $arctan(\frac{30}{n^2-n+900})$; 가자 $\sqrt{\frac{900+n^2}{900}}$; next	
현수선	for $n=1$ to 100 head 1, $\frac{n}{30}$; 가자 1; next	돌자 -90; for $n=1$ to 100 돌자 $arctan(\frac{30}{n^2-n+900})$; 가자 1; next	
나선	for n=1 to 360 move $-\sin\left(\frac{n^2+n}{2}\right)$, $\cos\left(\frac{n^2+n}{2}\right)$ next	for n=1 to 360 돌자 n; 가자 1 next	
나선 2	for n=1 to 1000 move $-n\sin n$, $n\cos n$ next	for n=1 to 1000 돌자 1; 가자 n; next	
cycloid	for n=1 to 360 move $-\sin n \cdot \sin n$, $\sin n \cdot \cos n$ next	for n=1 to 360 돌자 1; 가자 $\sin n$ next	
cardioid	for n=1 to 540 move $-\cos\frac{n}{3} \cdot \sin n$, $\cos\frac{n}{3} \cdot \cos n$ next	for n=1 to 360 돌자 1; 가자 $\cos\frac{n}{3}$ next	

좀 더 자연스러운 거북 행동을 만들기 위해서 원을 그리는 명령인 ‘조금 돌고, 조금 움직이고’와 같이 한 절차를 하나의 명령, 접선 방향으로 머리를 돌릴 필요가 있다. 이를 위해 $\tan(s_n - s_{n-1}) = \frac{\tan s_n - \tan s_{n-1}}{1 + \tan s_n \cdot \tan s_{n-1}}$라는 공식을 이용해 보자. 초기 상태로 머리 방향을 돌리는 대신 도는 각도를 정확히 나타내는 것이다. 〈표 4〉의 ‘포물선’과 ‘현수선’은 삼각함수

의 차 공식을 이용해 만든 자연스러운 ‘가자’, ‘돌자’ 명령이다. 포물선의 경우 각 단계의 ‘돌자’ 값을 이 공식에 대입하여 차이의 각도를 구하고 이를 통해 좀 더 자연스러운 곡선을 만들어 낼 수 있다. 또 〈표 4〉의 두 나선은 거북 명령을 ‘move’ 명령으로 번역한 것이며, cycloid와 cardioid에 대한 표현 방법은 Armon(1999)의 ‘내재적 식(intrinsic equation)’을 거북 행동으로 나타낸 것이다. 주목할 점은 〈표 3〉과 다르게 〈표 4〉에 제시된 명령을 실행한 결과에서 거북이는 항상 곡선의 접선 방향을 향하고 있다는 점이다. 완성된 결과만으로는 같은 곡선이지만 실제 거북 행동 절차는 다르다. 또한 행동을 관찰하면 〈표 4〉가 훨씬 자연스럽다. 이 문제는 거북 행동을 이용해 움직임을 만들 때 굉장히 중요하다.

Ⅳ. 거북 행동 표현의 교육적 의미 모색

LOGO 환경에서 학습자는 거북이 등에 올라타고 행동을 만들고 그 행동의 결과로 곡선을 얻게 된다. 일단 거북이 등에 올라탄 학습자는 외부 변수인 절대적 방향을 알 수 없다. 오로지 거북 머리 방향에 따른 상대적 방향만을 알 수 있을 뿐이다. 이런 상태에서 ‘가자’, ‘돌자’의 LOGO 기본 명령을 이용해 원하는 행동과 곡선을 찾아야 한다. LOGO는 운동을 가장 자연스러운 기본 명령, ‘가자’, ‘돌자’로 재구현하는 환경이다. 만약 거북 기본 명령으로 더 복잡한 곡선을 구성하는 절차를 이해한다면 거북 명령은 평면기하 현상을 이해하는 중요한 모델 역할을 하는 것이다.

앞선 연구 결과는 초등학교에서 처음 접한 거북 마이크로월드에서 좀 더 복잡한 곡선을 구성할 수 있는 방법을 보여준다. 포물선 같은 학교 수학에 자주 등장하는 곡선을 익숙한 거북 마이크로월드 환경에서 구성하게 된다. 또한 포물선, 타원이나 쌍곡선, 연속 함수의 그래프 같은 학교 수학에 등장하는 곡선 이외에 보다 복잡한 곡선도 구성할 수 있다. 주목할 점은 이러한 과정이 많은 명령을 통해 이루어지는 것이 아니라 ‘가자’, ‘돌자’의 조합으로 이루어진다는 사실이다.

Resnick과 Silverman(2005)은 자신들의 마이크로월드 설계에 대한 경험을 바탕으로 ‘바닥은 낮게 천장은 높게’라는 설계 원칙을 제시하고 있다. 이는 초보자도 쉽게 접근할 수 있으면서 전문가도 사용할 수 있는 환경이어야 한다는 의미이다. ‘가자’, ‘돌자’의 거북 명령 체계는 직관적으로 쉽게 도입될 수 있으면서도 가자값과 돌자값을 조절하면 높은 수준의 곡선을 표현하는 명령 체계이기도 하다. 이 환경은 학생들이 쉬운 명령으로 복잡한 곡선을 동적으로 표현하는 경험을 제공한다. 적은 수의 공리와 공준, 정의로부터 수학적 지식이 생성되듯이 학습자는 ‘가자’, ‘돌자’의 두 가지 명령으로 다각형, 포물선, 타원, 나아가 복잡한 곡선을 구

성하게 된다.

이때, 곡선의 행동 표현은 주어지는 것이 아니라 번역 활동의 과정에서 자연스럽게 발생한다. 함수의 그래프라는 곡선을 순간순간 변화의 조합으로 이해하고, 이를 ‘move’ 명령 체계로 재구성하고, 다시 이를 ‘가자’, ‘돌자’로 번역하는 활동을 통해 원하는 곡선의 행동 표현을 구현하게 된다. 이때, ‘move’ 명령은 곡선의 동적 표현을 가능하게 하는 기본 명령인 동시에 ‘가자’, ‘돌자’ 명령 표현이 가능하도록 도와주는 중간 매개자 역할을 한다. 처음부터 포물선을 ‘가자’, ‘돌자’ 명령으로 구성하는 것은 거의 불가능하지만, 먼저 ‘move’ 명령으로 포물선을 만드는 과정을 거친다면 목표에 도달하게 된다.

<표 5>는 Abelson과 diSessa(1980)의 거북기하와 좌표기하의 구분에서 ‘move’ 명령의 위치를 나타낸다. 새로운 명령 ‘move’는 거북기하의 세 가지 특징 중 ‘국소적’, ‘절차적’ 특징을 공유한다. 이 두 가지 특징으로 ‘move’ 명령은 곡선을 동적으로 표현하는 도구가 된다. 하지만 ‘move’ 명령은 좌표기하의 ‘외재적’ 특징도 함께 가지고 있어 좌표기하 관점에서 곡선을 표현하기에 적절한 도구가 된다. 좌표기하와 ‘move’ 거북기하, ‘가자’, ‘돌자’ 거북기하 사이의 번역 활동은 어느 곳에서도 알려줄 수 없는 곡선에 대한 통찰을 학습자에게 던져줄 수 있을 것이다.

<표 5> ‘move’ 명령

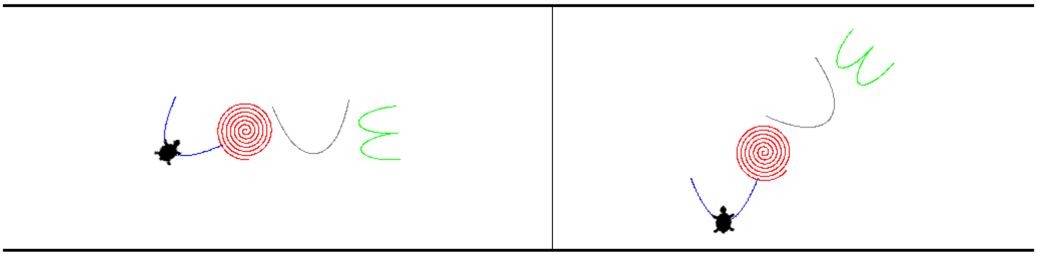
거북기하 ‘가자’, ‘돌자’	거북기하 ‘move’	좌표기하
내재적(intrinsic)	외부적(extrinsic)	외부적(extrinsic)
국소적(local)	국소적(local)	전체적(global)
절차(procedure) 강조	절차(procedure) 강조	식(equation) 강조

또한 ‘move’ 명령은 벡터의 좌표 표현으로 그 합이 각각의 성분의 합10)으로 표현할 수 있다는 장점을 가진다. 김화경, 송민호(2007)는 ‘circle model’을 이용하여 타원이나 epicycloid, hypocycloid를 두 원의 합성으로 파악하고 각각의 원을 그리는 명령을 단계별로 합하는 방식으로 epicycloid, hypocycloid를 만들고 이들을 동적으로 표현하는 방법과 학습-지도 사례에 대해 논의하고 있다. 이번 연구에서 살펴본 두 가지 명령들 사이의 번역 방법을 이용하면 이들 곡선은 다시 ‘가자’, ‘돌자’ 명령으로 표현이 가능하다. 이때 ‘move’ 명령의 좌표 표현과 그 합 표현이 없었다면 자연스러운 거북 운동으로 epicycloid를 만드는 것은 거의 불가능할 것이다. 우리는 ‘circle model’과 ‘move’라는 중간 매개자를 이용하여 원하는 자연스러운 거북 행동을 얻을 수 있는 것이다. 이는 DGS에서 원 운동의 합성으로 만들어진 ‘circle model’을

10) move a, b + move c, d = move a+c, b+d, $(a, b)+(c, d)=(a+c, b+d)$를 의미한다.

이용해 곡선을 관계 만족하는 자취로 도입하고, 각각의 원 운동을 ‘move’ 명령으로 번역하고, 이 글에서 살펴본 방법으로 ‘가자’, ‘돌자’로 번역하는 활동을 통해 자연스러운 거북 행동을 얻게 된다. 이러한 내용에 깊은 이해의 기회가 되는 번역 활동의 핵심은 ‘circle model’이나 ‘move’ 같은 중간 매개자의 적절한 이용에 있다.

김화경(2006)과 Kafai(1995) 등의 연구는 ‘디자인 활동을 통한 학습(learning by designing)’을 ‘learning by making’, 즉 constructionism의 구체적 실현으로 본다. 디자인 활동의 의의에 대하여 김남희(2004)와 Kafai(1995)는 구체적 수학 내용 학습과 관련하여 그 의미를 주목하였고, Eisenberg(2002) 그리고 Resnick과 Silverman(2005), 김화경(2006)은 창의적 인공물 구성 디자인 활동을 살펴보고 있다. 전자의 연구들은 특정 수학 지식 학습을 위한 디자인 활동을 설계하고 있고, 후자의 연구는 자유로운 디자인 활동을 통한 창의적 학습을 중요하게 여기고 있다. 때문에 후자의 연구는 특정 수학 지식에 관한 학습보다는 자유로운 창의적 사고력과 연결되어 수학교육과 동떨어져 연구되는 경우가 많다. 그러나 이 글에서 논의한 바와 같이 ‘가자’, ‘돌자’의 명령으로 포물선, 타원, 쌍곡선, 현수선과 같이 수학적 의미를 담고 있는 곡선을 이용한다면 구체적 수학 학습과도 연결할 수 있을 것이다. 즉 곡선의 동적 표현을 통해 타원이나 물방울 모양의 타일을 만들고 이 타일을 포물선 운동 시키는 디자인 활동을 생각해 볼 수 있다.



〔그림 3〕 디자인 활동의 예

예를 들어, 〔그림 3〕은 ‘가자’, ‘돌자’의 명령으로 만든 거북 운동이다. 여기서 ‘L’은 포물선, ‘O’는 나선, ‘V’는 현수선, 그리고 ‘E’는 타원의 행동 표현이다. 이 표현을 만드는 과정은 먼저 ‘move’로 표현하고 이를 ‘가자’, ‘돌자’로 번역하는 과정을 거쳤다. 이는 ‘LOVE’라는 자취를 만든 것이며 동시에 그 모양의 운동을 만든 것이다. 거북이 대신 움직일 수 있는 타일을 이용하면 ‘LOVE’ 운동을 하는 타일의 동영상을 만든 것이기도 하다. 거북 명령 ‘가자’, ‘돌자’로 타원 모양의 밑그림을 그리고 이를 타일로 만들어 그 타일을 거북 명령으로 포물선 운동시키는 디자인 활동을 생각해 볼 수 있다. 〔그림 3〕은 수학적 곡선과 그 운동을 이용해 만든 디자인 활동의 예로 명령의 내재적 특징을 보여준다. 학습자가 이 환경에서 원하는 운

동을 만들기 위해서는 곡선을 이해하고 곡선을 동적으로 표현하는 디자인 활동은 바람직한 '컴퓨터와 수학교육'을 중등학교 수준까지 확장해서 생각해 볼 수 있다. 현재의 교육과정에 무리하게 이러한 구성 활동을 추가하는 것은 현실적 문제가 있겠지만, 영재교육이나 창의성 수학교육 프로그램에서의 실현은 가능할 것이다. 기본 명령인 '가자', '돌자'를 생성자로 하여 곡선을 만들고 곡선 운동을 만드는 활동은 학교 수학의 곡선을 동적 관점에서 이해하는 기회를 제공하는 심화학습 소재가 될 수 있다. 이러한 활동에서 중요한 것은 디자인의 결과가 아니라 디자인의 과정이며, 이 과정에서 곡선의 내재적 성질을 이용하는 것이다. 곡선이 학습의 최종 목표가 되는 것이 아니라 원하는 디자인을 완성하기 위한 수단이 된다. 자신이 원하는 디자인, 동영상을 만들기 위해, 즉 '날아가는 공 운동'을 만들기 위해 학습자는 포물선 등의 곡선을 동적으로 이해하고 이를 이용한다. 이는 수학을 도구적 지식으로 이용하는 기회가 될 것이다.

Ⅴ. 맺으며

우리는 현행 교육과정에서 곡선을 다룰 때 동적 표현이 소홀하게 다루어져 있고, 상보적 관점에서 곡선의 동적 표현이 가능한 환경과 그 환경에서 곡선의 동적 표현 방법에 관한 연구가 필요하다는 인식에서 출발하였다. 이에 초등학교 수학 익힘책에 제시된 거북 명령 프로그램을 이용해 곡선을 동적으로 표현하는 방법을 살펴보았다. 거북 명령을 '가자', '돌자'의 내재적 명령 체계에 'move'라는 좌표기하의 특징을 갖는 명령으로 구분하고, 두 가지 명령으로 여러 가지 곡선을 그리는 방법을 논의하였다. 어떤 곡선은 '가자', '돌자'로 표현하기가 수월하였으며 다른 곡선들은 'move'로 그리는 것이 쉬웠다. 그리고 우리는 두 가지 명령 체계 사이의 번역 방법을 수학적으로 모색하고, 곡선을 동적으로 표현하는 환경과 활동에 관해 논의하였다. 특히 중간 매개자인 'move'의 역할과 교육적 의미를 살펴보았다.

이 글은 몇 가지 한계점을 가지고 있어 비판이 있을 수 있다. 먼저 곡선을 거북 마이크로월드에서 동적으로 표현하는 활동은 삼각함수나 역삼각함수, 복잡한 수식을 이용해야 했다. 이는 교육과정의 흐름을 고려하면 올바른 순서가 아니다. 삼각함수나 코사인 제2법칙과 같은 공식은 고등학교에서 학습하는 내용이며 역삼각함수는 대학 미적분의 내용이다(Anton et al., 2005). 다음으로 곡선의 동적 표현을 위해 너무 많은 프로그래밍이 요구된다. 수학교육적으로 의미 있는 부분은 곡선에 관해 사고하고 표현하는 방식이며 프로그래밍은 비본질적 요소이다. 이에 너무 많은 프로그래밍을 요구하는 것은 수학적 사고에 방해가 될 수 있다.

교육과정에 적합하지 않다는 한계로 곡선의 동적 표현을 현재의 교육과정과 무리하게 연

결시키는 것은 결코 바람직하지 않다. 그러나 앞서 언급한 대로 영재교육이나 창의력 수학 교육과 연결한다면 의미를 가질 수 있을 것이다. 삼각함수가 역사적으로 자연스러운 도입(Maor, 1998)이었음을 고려한다면 비교적 일찍 도입을 고려할 수도 있을 것이다. 여러 가지 의미 있는 곡선을 시각적으로 구현하는 방법은 가능한 명령의 개수를 늘리는 방법과, 적은 수의 명령으로 복잡한 수식을 이용하는 방법이 있을 수 있다. 우리는 후자가 더 교육적 의미를 가질 수 있다고 보고 후자를 택한 것이다.

다음으로 비본질적인 프로그래밍이 과도한 것은 분명 문제이다. 하지만 프로그래밍을 포기하는 것은 옳은 방법이 아니다. 정보화 사회의 유창성의 핵심이 프로그래밍이기 때문이다(NRC, 1999). 오히려 환경의 수정을 통해 사용자 친화적 명령 체계로 환경을 수정할 필요가 있겠다. 예를 들어 '마우스 끌기'로 명령을 실행하는 'Squeak(Allen-Conn와 Rose, 2003)', 'Boxer(diSessa, 2000; Sherin, 2002)', 그리고 'Scratch(http://scratch.mit.edu)' 같은 방식은 프로그래밍 환경이면서 사용자 친화적이다. 마우스 끌기를 이용한 프로그래밍은 자판을 통한 입력 방식보다 직관적인 명령 방식으로 훨씬 사용자 친화적이다(Resnick, 2007).

이 글이 모든 곡선을 LOGO 명령으로 바꾸는 방법을 조사한 것은 아니다. 또한 같은 곡선을 그리더라도 그 절차는 달라질 수 있다. 그런 측면에서 이 글에 제시된 명령들이 '가장 짧은' 효율적 명령이라고 볼 수도 없다. 또한 이 글에서 살펴본 명령은 곡선의 개형만을 중요하게 여기고 크기에 대해 논의하지 않았다. 이로 인해 정확한 번역 활동이 이루어지는 데에는 한계를 가진다. 이에 개형뿐만 아니라 크기도 같은 곡선을 정적·동적으로 표현하는 방식을 찾는 연구가 더 요구되며, 곡선 길이나 넓이와 같은 곡선의 기하학적 성질(김홍종, 2004)을 이용하고 연결하는 방법도 찾아야 한다. 또한 여기 제시된 방법들은 역삼각함수가 모든 단계에 적용되지 않는다는 점도 주목해야 한다. 분모가 0이 되는 경우는 컴퓨터가 계산할 수 없어 '돌자 90'과 '돌자 −90'을 구별하지 못하는 문제가 발생한다. 이러한 문제점을 해결할 수 있는 방법을 모색해야 하며 컴퓨터 환경이 적절하게 수정되어야 한다.

무엇보다도 이 글은 이론적으로 곡선의 동적 표현을 위한 방법만을 모색하고 있을 뿐, 심층적인 교육적 논의가 부족하다. 더 깊은 교육적 논의를 위해서는 여기에 제시된 환경과 표현 방법을 이용해 실제 학습-지도 상황에서 연구되어야 한다. 그리고 그 결과를 분석하여 컴퓨터 환경은 다시 수정되어야 하고 이론적 논의도 새롭게 다듬어질 필요가 있다. 이 글은 '가자', '돌자' 명령으로 곡선을 그리는 방법을 확인한 것에 불과하다. 교육적 의미를 확인할 수 있는 후속 연구를 기대한다.

참 고 문 헌

교육인적자원부 (2002a). **수학 익힘책 3-가** . 대한교과서주식회사.

교육인적자원부 (2002b). **수학 익힘책 4-가** . 대한교과서주식회사.

김남희 (2004). 중등수학 탐구를 위한 예비수학교사의 수학프로그램(GrafEq.) 활용 사례. **수학교육**(한국수학교육학회)**,** **43**(4), 405-417.

김홍종 (2004). **미적분학 1**. 서울: 서울대학교 출판부.

김화경 (2006). **'컴퓨터와 수학교육' 학습-지도 환경에 관한 연구**. 박사학위 논문, 서울대학교 대학원.

김화경, 송민호 (2007). LOGO와 DGS 매개 모델과 오류 사례. **수학교육학연구**(대한수학교육학회)**,** **17**(2), 111-125.

조한혁 (2003). 컴퓨터와 수학교육. **수학교육**(한국수학교육학회)**,** **42**(2), 177-192.

Abelson, H., & diSessa, A. (1980). *Turtle geometry.* Cambridge, MA: MIT Press.

Ackerman, E. K. (2004). Constructing knowledge and transforming the world. In M. Tokoro & L. Steels (Eds.), *A learning zone of one's own: Sharing representations and flow in collaborative learning environments* (pp. 15-36). Amsterdam: IOS Press.

Allen-Conn, B. J., & Rose, K. (2003). *Powerful ideas in the classroom: Using squeak to enhance math and science learning.* Viewpoints Research Institute.

Anton, H., Bivens, I., & Davis, S. (2005). *Calculus* (8th ed). Wiley.

Armon, U. (1999). An algorithm that translates intrinsic equations of curves into intrinsic procedures of these. *International Journal of Mathematical Education in Science and Technology, 30*(6), 833-854.

Bers, M. U. (2001). *Identity construction environments: The design of computational tools for exploring a sense of self and moral values.* Thesis of doctor of philosophy at the MIT.

Cho, H., Kim, H., & Song, M. (2007). Mediating model between LOGO and DGS for planar curves. *Proceeding of Psychology of Mathematics Education, 31.* Seoul: PME.

diSessa, A. (2000). *Changing minds.* Cambridge. MA: MIT Press.

Eisenberg, M. (2002). Output devices, computation, and the future of mathematical crafts. *International journal of computers in mathematical learning, 7*(1), 1-44.

Kafai, Y. (1995). *Minds in play: computer game design as a context for children's learning.* NY: Lawrence Erlbaum Associates, Publishers.

Kafai, Y., & Resnick, M. (1996). *Constructionism in practice.* NJ: Lawrence Erlbaum Associates, Publishers.

Maor, E. (1998). *Trigonometric Delights.* Princeton University Press.

National Research Council (1999). *Being fluent with information technology.* National Academies Press.

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas.* Cambridge, Massachusetts: Perseus Publishing.

Resnick, M. (2007). All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten. *Proceedings of the ACM SIGCHI conference on Creativity & Cognition.* Washington, DC.

Resnick, M., & Silverman, B. (2005). Some reflections on designing construction kits for kids. *Proceeding of interaction design and children conference.* Boulder, CO.

Sherin, B. (2002). Representing geometric constructions as programs: a brief exploration. *International Journal of Computers for Mathematical Learning,* 7(1), 101-115.

Taylor, R. (1980). *The computer in the school: tutor, tool, tutee.* NY: Teachers College Press.

• *논문 접수 : 2008년 2월 1일 / 수정본 접수 : 2008년 4월 3일 / 게재 승인 : 2008년 4월 15일*

ABSTRACT

Dynamic Representations of Curves in Turtle Microworld

Hwa-Kyung Kim(Full-time lecturer, Sangmyung University)

In this paper, we discuss the method of drawing a curve using LOGO command. There are two different representations of a curve. One is a static representation as set of points, the other is a dynamic representation. LOGO is an intrinsic environment where a curve can be dynamically represented as turtle actions. We emphasize the importance of translation activity from a static to a dynamic representation of LOGO command.

For this purpose, we first consider constructionism and microworld. And we survey the elementary mathematics textbook where LOGO is introduced. We consider the method of representation a curve using ‘FD’, ‘RT’ command. Next we expand our the method to representing continuous function graphs. And we consider meanings of this translation activity between two representations in the view of constructionism.

Key Words : microworld, turtle geometry, LOGO, computers and mathematics education, JavaMAL