

수치적분에 의한 정적분으로의 실험적 접근

좌 창 보 (남녕고등학교)

김 철 수 (제주대학교)

1. 서 론

1980년대 이래 컴퓨터는 교육을 포함한 사회의 모든 영역에서 널리 사용되고 있다. 이제 컴퓨터 없이는 하루도 살아갈 수 없는 시대가 도래하고 있다. 새로운 프리젠테이션 도구들은 교육의 질과 환경을 엄청나게 바꿔 놓고 있다. 수학교육에서도 컴퓨터를 교수-학습 과정에 활용하려는 의지와 요구가 극대화되고 있고 실제로 수학이라는 교과목의 성격을 살펴볼 때, 컴퓨터만큼 수학 학습의 과정을 풍요롭게 할 수 있는 교육매체도 드물다.

컴퓨터가 가지는 다양한 기능은 추상적인 수학 내용을 시각화하여 지도 할 수 있을 뿐 아니라, 그 시각화가 학생들의 직접적인 경험이나 통제를 통해 이루어질 수 있다는 점에서 수학 학습의 어려움을 완화시켜 준다. 특히 형식적인 증명이나 개념 학습의 전 단계에서 그래픽이나 애니메이션, 시뮬레이션을 통한 직관적인 탐구적 활동은 수학의 역동적이고 발생적인 측면을 부각시킬 수 있다. 또, 산술적인 계산 뿐만 아니라 대수적 문자식의 변환도 신속히 처리될 수 있게 됨으로써 수학교육은 종래의 계산 기능 위주에서 사고력 배양 중심으로 옮겨갈 수 있게 된다.

또한, 컴퓨터 프로그램 활동은 오류수정의 기회를 통해 사고력 향상을 위한 기회로 사용될 수 있다. 오류는 예상하지 못한 곳에서 일어나기 때문에 학생들의 흥미를 끌 수 있으며 그 오류를 제거하기 위해 반드시 무엇을 할 수밖에 없기 때문에 자신의 행동에 대한 새로운 통

찰로 이끌 수 있다.

수학은 매우 추상적인 학문이기 때문에 학생들에게 이해시키기 어려운 특성을 가지고 있다. 컴퓨터의 시각적·조작적 기능은 학생들로 하여금 추상과 구체의 만남을 통해 수학을 보다 쉽게 접근할 수 있게 해준다. 이러한 이유로 인하여 근래 많은 전문가들 사이에는 종이와 연필에만 의존하는 재래식 교육방법을 탈피하여 첨단기술을 수학 교육에 도입함으로써 창의적이고 생동감 있는 도전이 과학기술 현장에 이용될 수 있는 것이 되도록 교과내용과 교육방법이 개혁하여야 한다는 시대적 요청이 꾸준히 제기되고 있다.

본 논문은 이러한 관점에서 입시 위주의 주입식 수학교육에서 탈피하여 수학교실에서 컴퓨터를 활용함으로써 탐구와 관찰이라는 실험적 활동을 통해 수학교육에 새로운 활력을 불어 넣고자 하는 데 그 목적을 갖는다. 이러한 목적에 즈음하여 현재 고등학교에서 다루는 정적분의 도입에 이용되는 구분구적법의 이론을 실험적인 방법을 통해 고찰해 봄으로써 학생들로 하여금 무한개념과 더불어 정적분의 개념을 구체적으로 이해할 수 있도록한다.

2. Riemann 적분과 정적분의 근사계산법

2.1 Riemann 적분

Riemann 적분의 개념은 다음과 같다. 함수 $f(x)$ 가 폐구간 $[a, b]$ 에서 유계인 함수(연속함수일 필요는 없음)라고 하자. 즉, 임의의 $x \in [a, b]$ 에 대하여, $m \leq f(x) \leq M$ 을 만족하는

실수 m 과 M 이 존재한다고 하자.

폐구간 $[a, b]$ 의 분할 $a = x_0 < x_1 < x_2 < \dots < x_{n-1}$

$x_n = b$ 를 P 라고 할 때, 분할 P 에 대하여

$$M_i = \sup f(x), \quad m_i = \inf f(x) \quad (x_{i-1} \leq x \leq x_i)$$

라고 하면 분할 P 에 대한 f 의 상합(upper sum) $U(P, f)$ 와 하합(lower sum) $L(P, f)$ 는 다음과 같이 정의된다.

$$U(P, f) = \sum_{i=1}^n M_i \Delta x_i, \quad L(P, f) = \sum_{i=1}^n m_i \Delta x_i$$

$$(\text{단, } \Delta x_i = x_i - x_{i-1})$$

일반적으로 $c_i \in [x_i, x_{i-1}]$ 일 때, 분할 P 에 대한 f 의 리만 합(Riemann Sum)은 다음과 같이 정의된다.

$$R(P, f) = \sum_{i=1}^n f(c_i)(x_i - x_{i-1})$$

f 가 폐구간에서 연속이면 최대값, 최소값을 가지므로 소구간 $[x_{i-1}, x_i]$ 에서 최대점을 a_i , 최소점을 β_i 라고 하면,

$$U(P, f) = \sum f(a_i)(x_i - x_{i-1})$$

$$L(P, f) = \sum f(\beta_i)(x_i - x_{i-1})$$

가 되고, $c_i \in [x_i, x_{i-1}]$ 이면 $f(\beta_i) \leq f(c_i) \leq f(a_i)$ 이므로 $L(P, f) \leq R(P, f) \leq U(P, f)$ 가 된다. 다시 말해서 리만 합은 항상 상합과 하합 사이에 있다. 다음

$$\int_a^b f(x) \Delta x = \inf U(P, f),$$

$$\int_a^b f(x) \Delta x = \sup L(P, f)$$

라 놓고 이것을 각각 함수 f 의 $[a, b]$ 에서의 상적분(upper intergral), 하적분(lower integral)이라고 한다. 만일 위의 상적분과 하적분의 값이 같으면 함수 f 는 $[a, b]$ 에서 리만 적분가능(Riemann integrable)하다고 하고, 그 공통된 값을 기호로 $\int_a^b f(x) dx$ 와 같이 나타내고 함수 f 의 구간 $[a, b]$ 에서의 정적분이라고 한다.

구간 $[a, b]$ 의 임의의 분할 P 에 대하여 다음 부등식이 성립한다.

$$m(b-a) \leq L(P, f) \leq U(P, f) \leq M(b-a)$$

따라서 임의의 유계인 함수 f 에 대하여 상적분과 하적분은 존재한다.

2.2. 적분의 근사계산

미적분법의 기본 정리는 단지 피적분함수에 대한 역도함수를 구할 수 있을 때만 적분계산의 쉬운 방법을 우리에게 제공한다. 정적분의 정확한 값을 구하지 못하는 경우를 살펴보면 다음과 같다. 첫번째 경우는 '미적분학의 기본 정리'를 이용하여 $\int_a^b f(x) dx$ 를 구하기 위하여 f 의 원시함수를 찾아야 한다는 사실로부터 생겨난다. 그러나 때때로 원시함수를 구한다는 것은 어렵고 심지어 불가능할 경우도 있다. 예를 들어, 아래 적분들의 정확한 원시함수를 구하는 것은 불가능하다.

$$\int_1^2 e^x dx, \quad \int_1^2 \sqrt{1+x^3} dx$$

e^x 과 $\sqrt{1+x^3}$ 은 구간 $[1, 2]$ 에서 둘 다 연속이기 때문에 위의 정적분은 존재한다. 그 정적분값은 곡선 $y = e^x$ 과 $y = \sqrt{1+x^3}$ 아래의 면적을 나타낸다. 그러나 문제는 지금까지 연구한 방법이나 또는 그밖의 어떠한 다른 방법으로도 우리가 알고있는 함수의 식으로 원시함수를 표현할 수 없는 데 기인한다. 두 번째 경우는 함수의 기계적인 수치를 통해 과학적 실험으로부터 나온 것일 경우이다. 두 가지 경우 모두 정적분의 근사값을 구할 필요가 있다. 정적분이 리만(Riemann)합들의 극한으로 정의된 것을 생각해 본다면, 리만 합은 그러한 적분의 근사값으로써 이용되어질 수 있다.

그러한 이유 때문에 필요한 만큼의 소수점까지 정적분 값을 유추하는 방법이 고안되었다. 여기서는 이러한 광범위한 주제를 일반적으로 논하지는 않을 것이나, 고등학교 학생이 이해할 수 있는 범주에서 몇 개의 유용하고 효과적인 방법을 이론적으로 소개하고 이를 실험적인 방법을 통해 확인케 함으로써 극한개념과 정적분의 정의를 충분히 이해하고 활용케 하고자 한다.

2.2.1 직사각형 공식

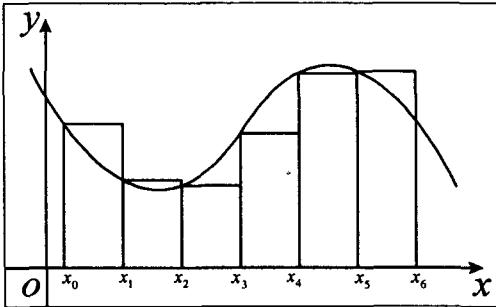
함수 f 가 구간 $[a, b]$ 에서 연속이고, 구간 $[a, b]$ 를 $\Delta x = (b-a)/n$ 의 길이를 갖는 n 개의

부분구간으로 이루어진 분할을 정한다. 그러면 x_i^* 가 그러한 분할의 i 번째 부분구간 $[x_{i-1}, x_i]$ 안에 있는 점일 때, 정적분의 정의에 의하면

$$\int_a^b f(x) dx = \lim_{n \rightarrow \infty} \sum_{i=1}^n f(x_i^*) \Delta x$$

이다. 따라서, 분할된 소구간들의 길이를 충분히 작게 하면 (즉, n 의 값을 충분히 크게 잡으면) 적분값은 다음과 같이 근사값으로 대체할 수 있다.

$$\int_a^b f(x) dx \approx \sum_{i=1}^n f(x_i^*) \Delta x$$



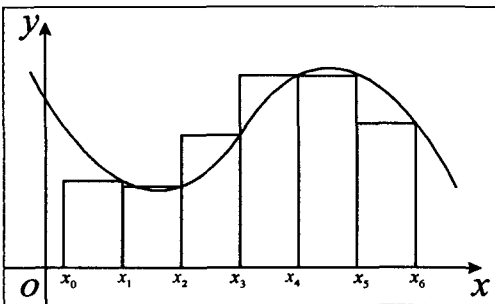
<그림 2.2.1(a)> 좌종점 근사값

x_i^* 가 그 구간의 좌종점일 때 $x_i^* = x_{i-1}$ 이고,

$$\int_a^b f(x) dx \approx L_n = \sum_{i=1}^n f(x_{i-1}) \Delta x \quad (2.1)$$

라고 쓸 수 있다.

$f(x) \geq 0$ 이면 그 적분은 면적을 나타내고, 식 (2.1)은 <그림 2.2.1(a)>에서 보여지는 직사각형들에 의한 이 면적의 근사값을 나타낸다.

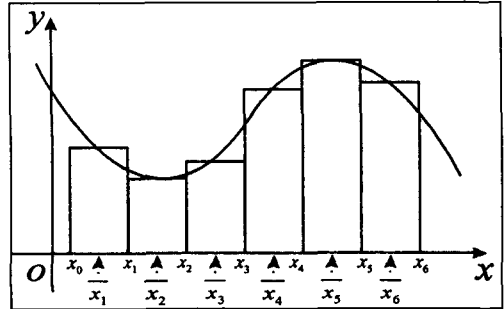


<그림 2.2.1(b)> 우종점 근사값

x_i^* 가 그 구간의 우종점일 때 $x_i^* = x_i$ 이고,

$$\int_a^b f(x) dx \approx R_n = \sum_{i=1}^n f(x_i) \Delta x \quad (2.2)$$

라고 쓸 수 있다<그림 2.2.1(b) 참조>. 식 (2.1)과 식 (2.2)에 의해 정의된 적분값 L_n 과 R_n 은 각각 좌종점 근사값, 우종점 근사값이라 부른다.



<그림 2.2.1(c)> 중간점 근사값

<그림 2.2.1(c)>는 L_n 혹은 R_n 보다 나은 형태인 중간점 근사값을 보여준다.

\bar{x}_i 가 주어진 분할의 i 번째 부분구간 $[x_{i-1}, x_i]$ 의 중간점일 때, 중간점 근사값은 다음과 같이 표현할 수 있다.

$$\int_a^b f(x) dx \approx M_n = \sum_{i=1}^n f(\bar{x}_i) \Delta x \quad (2.3)$$

$$\text{단, } \bar{x}_i = \frac{1}{2} (x_{i-1} + x_i)$$

2.2.2 사다리꼴 공식

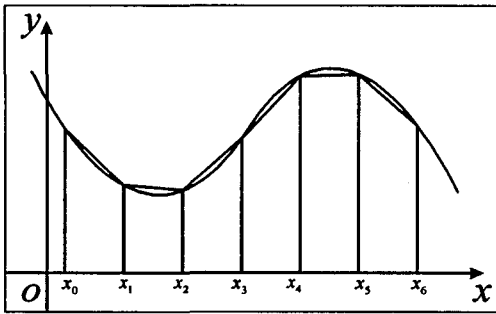
적분의 근사값을 구하는 수치적분의 간단한 방법 중 하나는 사각형 대신 사다리꼴로 곡선 아래의 면적을 추정하는 것이다.

함수 f 가 구간 $[a, b]$ 에서 연속이고, 구간 $[a, b]$ 를 $\Delta x = (b-a)/n$ 의 길이를 갖는 n 개의 부분구간으로 이루어진 분할을 정한다. 각 소구간의 길이가 같을 필요는 없으나, 만일 같게 하면 공식의 결과가 더 간단해진다.

<그림 2.2.2(a)>에서 알 수 있듯이, $f(x) \geq 0$ 인 구간에서의 적분값은 다음과 같이 사다리꼴의 면적의 합으로 근사값을 추정할 수 있다.

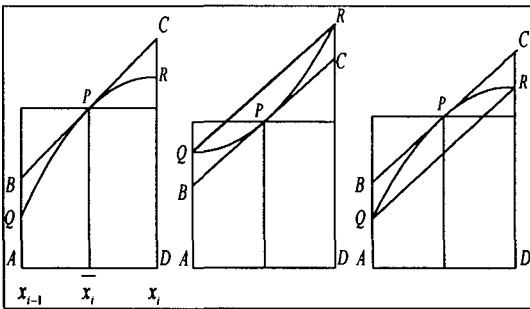
$$\int_a^b f(x) dx \approx T_n$$

$$\begin{aligned}
 &= \sum_{i=1}^n \frac{1}{2} [f(x_{i-1}) + f(x_i)] \Delta x \\
 &= \frac{\Delta x}{2} [\{f(x_0) + f(x_1)\} + \{f(x_1) + f(x_2)\} + \\
 &\quad \dots + \{f(x_{n-1}) + f(x_n)\}] \\
 &= \frac{\Delta x}{2} [f(x_0) + 2f(x_1) + 2f(x_2) + \\
 &\quad \dots + 2f(x_{n-1}) + f(x_n)] \tag{2.4}
 \end{aligned}$$



<그림 2.2(a)> 사다리꼴 근사값

다음 절에서 논하게 될 오차의 추정에서 알 수 있듯이, <그림 2.2(b)>는 중간점 근사값이 사다리꼴 근사값보다 더 정확한 이유를 보여준다.



<그림 2.2(b)>

중간점 공식에서 일반적인 직사각형의 면적은 윗면 위의 점 P에서의 곡선의 접선으로 이루어지는 사다리꼴 ABCD의 면적과 같다.

이 사다리꼴의 면적은 사다리꼴 공식에서 이용되어지는 사다리꼴 AQRD의 면적보다 곡선의 아래 면적에 더 가깝다. 즉, 중간점 오차는 사다리꼴 오차보다 더 작음을 알 수 있다.

따라서 중간점 근사값이 사다리꼴 근사값보다 더 정밀하며 다음과 같은 식으로 표현할 수

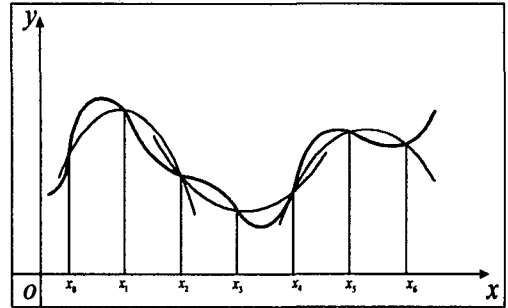
있다.

$$\left| \int_a^b f(x) - M_n \right| \leq \left| \int_a^b f(x) - T_n \right|$$

2.2.3 Simpson 공식

적분의 근사계산의 또 다른 방법은 곡선 아래의 면적을 근사적으로 구하기 위하여 직사각형이나 사다리꼴 대신에 포물선을 이용하는 경우이다. 이 경우는 포물선의 부분들로 이루어진 함수에 의하여 접근시키는 방법으로 사다리꼴 공식이나 중점 공식보다 더 좋은 근사값을 구하게 해준다.

함수 f 가 구간 $[a, b]$ 에서 연속이고, 앞에서와 같이 구간 $[a, b]$ 를 길이가 $\Delta x = (b-a)/n$ 인 n 개의 부분구간으로 이루어진 분할을 정한다. 그러나 이번에는 n 이 짝수라고 가정한다.

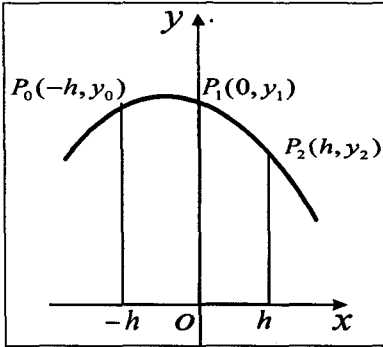


<그림 2.2.3(a)> Simpson 근사값

각 연속되는 구간의 쌍위에서 <그림 2.3.3(a)>에 표현된 포물선에 의해 곡선 $y=f(x)$ 를 근사적으로 구하여 보자.

이를 구하기 위해 $y_i = f(x_i)$ 라고 하면 점 $P_i(x_i, y_i)$ 는 $x=x_i$ 에서의 곡선 위의 점이다. 또한 연속된 세 점 P_i, P_{i+1}, P_{i+2} 을 지나는 포물선은 유일하게 존재한다. 이제 계산을 단순화하기 위하여 우선 $x_0 = -h, x_1 = 0, x_2 = h$ 라고 생각하자 <그림 2.3.3(b)>.

연속된 세 점 P_0, P_1, P_2 을 지나는 포물선의 방정식은 $y = Ax^2 + Bx + C$ 의 형태이고, $x = -h$ 에서 $x = h$ 에 이르는 포물선 아래의 면적은 다음과 같이 나타낼 수 있다.



<그림 2.2.3(b)>

$$\begin{aligned} & \int_{-h}^h (Ax^2+Bx+C) dx \\ &= \left[A \frac{x^3}{3} + B \frac{x^2}{2} + Cx \right]_{-h}^h \\ &= A \frac{h^3}{3} + B \frac{h^2}{2} + Ch + A \frac{h^3}{3} - B \frac{h^2}{2} + Ch \\ &= \frac{h}{3} (2Ah^2+6C) \end{aligned}$$

그러나, 포물선은 $P_0(-h, y_0), P_1(0, y_1),$

$P_2(h, y_2)$ 를 지나기 때문에

$$\begin{aligned} y_0 &= A(-h)^2+B(-h)+C = Ah^2-Bh+C \\ y_1 &= C \\ y_2 &= Ah^2+Bh+C \end{aligned}$$

가 되고,

$$y_0 + 4y_1 + y_2 = 2Ah^2 + 6C$$

가 된다.

따라서 포물선 아래의 면적은

$$\frac{h}{3} (y_0 + 4y_1 + y_2) \text{로 다시 쓸 수 있다.}$$

마찬가지로 세 점 P_2, P_3, P_4 을 지나는 포물선의 $x=-h$ 에서 $x=h$ 에 이르는 포물선 아래의 면적은 $\frac{h}{3} (y_2 + 4y_3 + y_4)$ 이다. 이러한 방법으로 포물선 아래의 면적을 계산하여 모두 합하면,

$$\begin{aligned} & \int_a^b f(x) \approx S_n \\ &= \frac{h}{3} (y_0 + 4y_1 + y_2) + \frac{h}{3} (y_2 + 4y_3 + y_4) \\ & \quad + \dots + \frac{h}{3} (y_{n-2} + 4y_{n-1} + y_n) \\ &= \frac{h}{3} (y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 \\ & \quad + \dots + 2y_{n-2} + 4y_{n-1} + y_n) \end{aligned}$$

이다. 따라서 <그림 2.2.3(a)>에서 알 수 있듯이 $f(x) \geq 0$ 인 구간에서의 적분값은 다음과 같이 포물선의 아래부분 면적의 합으로 근사값을 추정할 수 있다.

$$\begin{aligned} \int_a^b f(x) dx &\approx S_n \\ &= \frac{\Delta x}{3} [f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) \\ & \quad \dots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)] \end{aligned} \quad (2.5)$$

단, n 은 짝수이고, $\Delta x = (b-a)/n$ 이다.

또한, Simpson 근사값은 사다리꼴과 중간점 근사값을 이용하면,

$$S_{2n} \approx \frac{1}{3} T_n + \frac{2}{3} M_n$$

과 같은 관계식을 얻을 수 있으므로 이를 활용하면 보다 쉽게 정확한 근사값을 얻을 수 있다.

2.2.4 근사적 수치적분의 오차

구간의 개수를 늘리면 근사값은 참값에 더욱 가까워질 것은 당연한 일이다. 그렇지만 수치적분에 의하여 구한 답이 얼마나 정밀한 것인가 하는 의문이 생긴다.

일반적으로 수치적분을 사용하면 두 가지 오차가 발생한다. 하나는 연속된 각 구간에서의 곡선을 직선 또는 포물선 등으로 어렵잡아 발생한 오차이며, 또 다른 하나는 계속적인 반올림을 함으로써 생기는 반올림 오차이다. 이러한 오차는 유한 개의 점 또는 이산 상태의 많은 점에서의 함수를 생각하기 때문에 발생하는 이산오차인 것이다. 구간의 개수를 증가시키면 곡선 아래 면적의 근사치의 오차를 줄일 수 있다. 이것은 분명히 이산오차를 감소시키는 효과를 얻는다. 그 반면에, 소구간의 개수를 증가시키면 계산하는 횟수의 증가로 반올림 오차를 증가시킨다. 사실 이러한 두 오차 사이에 미묘한 균형이 있고, 또 종종 이 두 개의 오차를 합한 총오차를 최소화하는 최적의 소구간의 개수를 정할 필요가 있다.

[정리 2.4.1] 함수 f 와 f', f'' 이 구간 $[a, b]$ 에서 연속이면 식(2.4)에서 표현된 근사값 T_n 에 대하여

$$T_n - \int_a^b f(x) dx = \frac{(b-a)h^2}{12} f''(c) \quad (2.6)$$

[따름정리 2.4.2] 함수 f 와 f', f'' 이 구간 $[a, b]$ 에서 연속이고, 식(2.4)에서 표현된 근사값 T_n 에 대하여, $M = \sup \{ |f''(x)| : x \in [a, b] \}$ 라고 하면

$$\left| T_n - \int_a^b f(x) dx \right| \leq \frac{(b-a)^3}{12n^2} M \quad (2.7)$$

을 만족한다.

[정리 2.4.3] 함수 f 와 f', f'' 이 구간 $[a, b]$ 에서 연속이면 식(2.3)에서 표현된 근사값 M_n 에 대하여

$$\int_a^b f(x) dx - M_n = \frac{(b-a)h^2}{24} f''(r) \quad (2.8)$$

를 만족하는 한 점 $r \in [a, b]$ 가 존재한다.

[따름정리 2.4.4] 함수 f 와 f', f'' 이 구간 $[a, b]$ 에서 연속이고, 식(2.3)에서 표현된 근사값 M_n 에 대하여 $M = \sup \{ |f''(x)| : x \in [a, b] \}$ 라고 하면,

$$\left| M_n - \int_a^b f(x) dx \right| \leq \frac{(b-a)^3}{24n^2} M \quad (2.9)$$

을 만족한다.

[정리 2.4.5] 함수 f 와 $f', f'', f''', f^{(4)}$ 가 구간 $[a, b]$ 에서 연속이면 식(2.5)에서 표현된 근사값 S_n 에 대하여

$$S_n - \int_a^b f(x) dx = \frac{(b-a)h^4}{180} f^{(4)}(s) \quad (2.10)$$

를 만족하는 한 점 $s \in [a, b]$ 가 존재한다.

[따름정리 2.4.6] 함수 f 와 $f', f'', f''', f^{(4)}$ 이 구간 $[a, b]$ 에서 연속이고

$M = \sup \{ |f^{(4)}(x)| : x \in [a, b] \}$ 라고 하면, 식(2.5)에서 표현된 근사값 S_n 에 대하여,

$$\left| S_n - \int_a^b f(x) dx \right| \leq \frac{(b-a)^5}{180n^4} M \quad (2.11)$$

을 만족한다.

3. 실험적 방법에 의한 정적분의 근사계산

3.1. $\int_1^2 \frac{1}{x} dx$ 의 근사계산

위의 정적분 값을 미적분학의 기본정리와 Mathematica를 이용하여 소수점 아래 20 자리까지 구하면 다음과 같다.

$$\int_1^2 \frac{1}{x} dx = [\ln]_1^2 = \ln 2 = 0.69314718055994530942 \dots$$

<표 3.1.1> Mathematica Procedure for Validating Riemann Sum

```
(* User Type *) f[x_]:=1/x; a=1; b=2; n=16; k=15;
(* Sample Input *)
ClearAll[LHSum,RHSum,Trap,Midpt];
LHSum[f_,over_List,n_]:=
Module[{a=over[[1]],b=over[[2]],h=(b-a)/n;
Return[Sum[f[a+h*i],{i,0,n-1}]*h]];
RHSum[f_,over_List,n_]:=
Module[{a=over[[1]],b=over[[2]],h=(b-a)/n;
Return[Sum[f[a+h*i],{i,1,n}]*h]];
Trap[f_,over_List,n_]:=
Module[{a=over[[1]],b=over[[2]],h=(b-a)/n;
Return[(Sum[f[a+h*i],{i,1,n-1}]+(f[a]+f[b])/2)*h]];
Midpt[f_,over_List,n_]:=
Module[{a=over[[1]],b=over[[2]],h=(b-a)/n;
Return[h*Sum[f[a+h*(i-1/2)],{i,1,n}]]];
m=Midpt[f,{a,b},n/2]; t=Trap[f,{a,b},n/2];
Print["Check Values for Numerical Integration:"]
Print[" "];
Print["Integration of ",OutputForm[f[x]]," on [",a,",",b,":"];
Print[" "];
Print["Exact Value      :",N[Integrate[f[x],{x,a,b}],k]];
Print[" LH Sum (n=",n,") : ",N[LHSum[f,{a,b},n],k]];
Print[" RH Sum (n=",n,") : ",N[RHSum[f,{a,b},n],k]];
Print[" Midpoint Sum (n=",n,") : ",N[Midpt[f,{a,b},n],k]];
Print[" Trapezoidal Sum (n=",n,") : ",N[Trap[f,{a,b},n],k]];
Print[" Simpson Sum (n=",n,") : ",N[(2*m+t)/3,k]];
(* Sample Output *)
Check Values for Numerical Integration:
Integral of 1/x on [1,2]:
Exact Value      : 0.693147180559947
LH Sum (n=16)    : 0.709016202207527
RH Sum (n=16)    : 0.677766202207527
Midpoint Sum (n=16) : 0.693025214330971
Trapezoidal Sum (n=16) : 0.693391202207527
Simpson Sum (n=16) : 0.693147652819419
```

이제 위에서 논한 적분의 근사값 계산들을 Mathematica(<표 3.1.1> 참조) 또는 수치적분

을 응용한 프로그램을 이용하여 그 근사값들을 구하여 보자.

편의상 구간 [1, 2] 을

$$1 = x_0 < x_1 < x_2 < \dots < x_{15} < x_{16} = 2$$

의 16 개의 부분구간으로 나누고 소수점 아래 8 자리까지 구하여 비교하기로 하자.

[좌중점 근사값]

$$\begin{aligned} \int_1^2 \frac{1}{x} dx &\approx L_{16} \\ &= \sum_{i=1}^{16} f(x_{i-1}) \Delta x \\ &= \Delta x (f(x_0) + f(x_1) + f(x_2) + \dots + f(x_{15})) \\ &\approx 0.70901620 \end{aligned}$$

[우중점 근사값]

$$\begin{aligned} \int_1^2 \frac{1}{x} dx &\approx R_{16} \\ &= \sum_{i=1}^{16} f(x_i) \Delta x \\ &= \Delta x (f(x_1) + f(x_2) + f(x_3) + \dots + f(x_{16})) \\ &\approx 0.67776620 \end{aligned}$$

[중간점 근사값]

$$\begin{aligned} \int_1^2 \frac{1}{x} dx &\approx M_{16} \\ &= \sum_{i=1}^{16} f(\bar{x}_i) \Delta x = \sum_{i=1}^{16} f\left(\frac{1}{2}(x_{i-1} + x_i)\right) \Delta x \\ &= \frac{\Delta x}{2} \left[f\left(\frac{1}{2}(x_0 + x_1)\right) + f\left(\frac{1}{2}(x_1 + x_2)\right) + \dots \right. \\ &\quad \left. + f\left(\frac{1}{2}(x_{15} + x_{16})\right) \right] \approx 0.69302521 \end{aligned}$$

[사다리꼴 근사값]

$$\begin{aligned} \int_1^2 \frac{1}{x} dx &\approx T_{16} \\ &= \sum_{i=1}^{16} \frac{1}{2} [f(x_{i-1}) + f(x_i)] \Delta x \\ &= \frac{\Delta x}{2} [f(x_0) + 2f(x_1) + 2f(x_2) + \dots \\ &\quad + 2f(x_{15}) + f(x_{16})] \approx 0.69339120 \end{aligned}$$

[Simpson 근사값]

$$\int_1^2 \frac{1}{x} dx \approx S_{16}$$

$$\begin{aligned} &= \frac{\Delta x}{3} [f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) \\ &\quad \dots + 2f(x_{14}) + 4f(x_{15}) + f(x_{16})] \approx 0.69314765 \end{aligned}$$

이제 n의 값을 증가시키면서 각 근사값과의 오차 및 오차율을 계산하여 보자.

X = L_n, R_n, M_n, T_n, S_n 일 때, 각 근사값의 오차와 오차율을

$$\begin{aligned} EX &= X - \int_1^2 \frac{1}{x} dx, \%X \\ &= \left\{ |EX| / \int_1^2 \frac{1}{x} dx \right\} \times \frac{1}{100} (\%) \end{aligned}$$

과 같이 정의하면 <표 3.1.2>를 얻을 수 있으며, 그로부터 다음과 같은 몇 가지를 관찰할 수 있다.

<표 3.1.2>

f(x)=1/x from 1 to 2					0.69314718
n	Ln	Rn	Mn	Tn	Sn
4	0.75952381	0.63452381	0.69121989	0.69702381	0.69325397
8	0.72537185	0.66287185	0.69266056	0.69412185	0.69315453
16	0.70901620	0.67776620	0.69302521	0.69339120	0.69314765
32	0.70102071	0.68539571	0.69311667	0.69320821	0.69314721
64	0.69706869	0.68925619	0.69313955	0.69316244	0.69314718
128	0.69510412	0.69119787	0.69314527	0.69315100	0.69314718
256	0.69412470	0.69217157	0.69314670	0.69314813	0.69314718
512	0.69363570	0.69265914	0.69314706	0.69314742	0.69314718
n	ELn	ERn	EMn	ETn	ESn
4	0.06637663	-0.05862337	-0.00192729	0.00387663	0.00010679
8	0.03222467	-0.03027533	-0.00048663	0.00097467	0.00000735
16	0.01586902	-0.01538098	-0.00012197	0.00024402	0.00000047
32	0.00787353	-0.00775147	-0.00003051	0.00006103	0.00000003
64	0.00392151	-0.00389099	-0.00000763	0.00001526	0.00000000
128	0.00195694	-0.00194931	-0.00000191	0.00000382	0.00000000
256	0.00097752	-0.00097561	-0.00000048	0.00000095	0.00000000
512	0.00048852	-0.00048804	-0.00000012	0.00000024	0.00000000
n	%Ln	%Rn	%Mn	%Tn	%Sn
4	9.57612%	8.45756%	0.27805%	0.55928%	0.01541%
8	4.64904%	4.36781%	0.07021%	0.14062%	0.00106%
16	2.28942%	2.21901%	0.01760%	0.03520%	0.00007%
32	1.13591%	1.11830%	0.00440%	0.00880%	0.00000%
64	0.56575%	0.56135%	0.00110%	0.00220%	0.00000%
128	0.28233%	0.28123%	0.00028%	0.00055%	0.00000%
256	0.14103%	0.14075%	0.00007%	0.00014%	0.00000%
512	0.07048%	0.07041%	0.00002%	0.00003%	0.00000%

1. n 값이 증가할수록 더욱 정확한 근사값을 얻을 수 있다. (그러나 n 이 아주 큰 경우에 누적된 반올림의 오차를 주의해야 할 많은 산술적인 연산들이 필요하다.)
2. 좌 · 우중점 근사값에서의 오차는 부호가 서로 다르고 n 의 값이 2배로 늘어날 때, 약 1/2 만큼 줄어든다.
3. 사다리꼴과 중간점 근사값은 좌 · 우중점 근사값보다 더 정확하다.
4. 사다리꼴과 중간점 근사값의 오차들은 부호가 서로 다르고 n 의 값이 2배로 늘어날 때, 약 1/4 만큼 줄어든다.
5. 중간점 근사값의 오차의 크기는 사다리꼴 근사값에서의 오차의 크기의 약 1/2에 해당한다.
6. Simpson 근사값은 그 어느것 보다도 더 정확하며 n 의 값이 2배로 늘어날 때, 약 1/16 만큼 줄어든다.

3.2. $\int_1^2 e^{x^2} dx$ 의 근사계산

$\int_1^2 e^{x^2} dx$ 의 정확한 원시함수를 구하는 일은 불가능하다. 그러나 위에서 논한 수치적분의 이론과 Mathematica 등의 프로그램을 이용하면 근사적으로 적분의 값을 구할 수 있으며, 또한 그 값의 오차를 추정할 수 있다. 다음의 <표 3.2.1>은 Mathematica에서 <표 3.1.1>과 같은 프로그램을 실행하여 얻은 출력의 결과이다.

<표 3.2.1>

Check Values for Numerical Integration: Integration of e^{x^2} on [1,2] :	
Exact Value	: 14.98997602
LH Sum (n=16)	: 13.43795136
RH Sum (n=16)	: 16.68044313
Midpoint Sum (n=16)	: 14.95540303
Trapezoidal Sum (n=16)	: 15.05919725
Simpson Sum (n=16)	: 14.99037334

[예제 3.2.1] $n = 16$ 인 사다리꼴 공식을 이용하여 적분 $\int_1^2 e^{x^2} dx$ 을 근사적으로 구하고, 그 근사값에 대한 오차의 상한을 추정하시오.

[풀이] $a=1, b=2, n=16$ 이므로 사다리꼴 공식을 사용하면,

$$\begin{aligned} \int_1^2 e^{x^2} dx &\approx T_{16} \\ &= \sum_{i=1}^{16} \frac{1}{2} [f(x_{i-1}) + f(x_i)] \Delta x \\ &= \frac{\Delta x}{2} [f(x_0) + 2f(x_1) + 2f(x_2) + \dots \\ &\quad + 2f(x_{15}) + f(x_{16})] \approx 15.05919725 \end{aligned}$$

$f(x) = e^{x^2}$ 이기 때문에 $f'(x) = 2xe^{x^2}$, $f''(x) = (2+4x^2)e^{x^2}$ 이다. 또한, $1 \leq x \leq 2$ 이므로 $1 \leq x^2 \leq 4$ 이며 $|f''(x)| = |(2+4x^2)e^{x^2}| \leq 18e^4$ 이다. 오차추정(식 2.7)에서 $M=18e^4, a=1, b=2, n=16$ 으로 택하면 오차의 상한은

$$\frac{(2-1)^3}{12 \cdot 16^2} \cdot 18e^4 \approx 0.319911$$

으로 추정할 수 있다. ■

[예제 3.2.2] $n = 16$ 인 중간점 공식을 이용하여 적분 $\int_1^2 e^{x^2} dx$ 을 근사적으로 구하고, 그 근사값에 대한 오차의 상한을 추정하시오.

[풀이] $a=1, b=2, n=16$ 이므로 중간점 공식을 사용하면,

$$\begin{aligned} \int_1^2 e^{x^2} dx &\approx M_{16} \\ &= \sum_{i=1}^{16} f(\bar{x}_i) \Delta x = \sum_{i=1}^{16} f\left(\frac{1}{2}(x_{i-1} + x_i)\right) \Delta x \\ &= \frac{\Delta x}{2} \left[f\left(\frac{1}{2}(x_0 + x_1)\right) + f\left(\frac{1}{2}(x_1 + x_2)\right) + \dots \right. \\ &\quad \left. + f\left(\frac{1}{2}(x_{15} + x_{16})\right) \right] \approx 14.95540303 \end{aligned}$$

$f(x) = e^{x^2}$ 이기 때문에 $f'(x) = 2xe^{x^2}$, $f''(x) = (2+4x^2)e^{x^2}$ 이다. 또한, $1 \leq x \leq 2$ 이므로 $1 \leq x^2 \leq 4$ 이며 $|f''(x)| = |(2+4x^2)e^{x^2}| \leq 18e^4$ 이다. 오차추정(식 2.9)에서 $M=18e^4, a=1, b=2, n=16$ 으로 택하면 오차의 상한은

$$\frac{(2-1)^3}{24 \cdot 16^2} \cdot 18e^4 \approx 0.159956$$

으로 추정할 수 있다. ■

[예제 3.2.3] $n=16$ 인 Simpson 공식을 이용하여 적분 $\int_1^2 e^{x^2} dx$ 을 근사적으로 구하고, 그 근사값에 대한 오차의 상한을 추정하시오.

[풀이] $a=1, b=2, n=16$ 이므로 Simpson 공식을 사용하면,

$$\begin{aligned} \int_1^2 e^{x^2} dx &\approx S_{16} \\ &= \frac{4x}{3} [f(x_0)+4f(x_1)+2f(x_2)+4f(x_3)+2f(x_4) \\ &\quad \dots +2f(x_{14})+4f(x_{15})+f(x_{16})] \approx 14.99037334 \\ f(x) &= e^{x^2} \text{ 이기 때문에 } f^{(4)}(x) = (12+48x^2+16x^4)e^{x^2} \text{ 이다.} \end{aligned}$$

또한, $1 \leq x \leq 2$ 이므로 $1 \leq x^2 \leq 4$ 이며 $|f^{(4)}(x)| = |(12+48x^2+16x^4)e^{x^2}| \leq 460e^4$ 이다. 오차추정(식 2.11)에 $M=460e^4, a=1, b=2, n=16$ 으로 택하면 오차의 상한은

$$\frac{(2-1)^5}{180 \cdot 16^4} \cdot 460e^4 \approx 0.002129$$

으로 추정할 수 있다. ■

3.3. 곡선의 곡률과 수치적분의 고찰

벡터함수 r 로 주어지는 원활한 곡선에서는 $r'(t) \neq 0$ 이다.

단위접선 벡터는

$$T(t) = \frac{r'(t)}{|r'(t)|}$$

로 주어지고, 곡선의 방향을 나타낸다. 곡선 위의 주어진 점에서의 곡률은 그 점에서 곡선이 얼마나 빠르게 방향을 바꾸는가 하는 것을 잴 것이다. 정확하게는, 곡률은 호의 길이에 대한 단위접선 벡터의 순간변화율의 크기로 정의된다.

[정의 3.3.1] 곡선의 곡률은 T 가 단위접선 벡터일 때, $\kappa = \left| \frac{dT}{ds} \right|$ 이다.

곡률은 s 보다는 매개변수 t 의 함으로 표시

되어 있으면 보다 쉽게 계산 할 수 있으며

$$\frac{dT}{dt} = \frac{dT}{ds} \frac{ds}{dt} \Rightarrow \kappa = \left| \frac{dT}{ds} \right| = \left| \frac{dT/dt}{ds/dt} \right|$$

이다. 또한, $ds/dt = |r'(t)|$ 이므로

$$\kappa(t) = \frac{|T'(t)|}{|r'(t)|}$$

로 표현할 수 있다.

[정리 3.3.1] 벡터함수 r 로 주어진 곡선의 곡률은 다음과 같이 표현된다.

$$\kappa(t) = \frac{|r'(t) \times r''(t)|}{|r'(t)|^3}$$

방정식이 $y=f(x)$ 인 평면곡선의 특별한 경우에 x 를 매개변수로 택하여, $r(x) = xi + f(x)j$ 로 쓸 수 있으며

$$r'(x) = i + f'(x)j, r''(x) = f''(x)j, j \times j = 0$$

이므로 $r'(x) \times r''(x) = f''(x)k$ 를 얻는다.

또한, $|r'(x)| = \sqrt{1+[f'(x)]^2}$ 이므로 곡률은 다음과 같이 표현된다.

$$\kappa(x) = \frac{|f''(x)|}{[1+(f'(x))^2]^{3/2}}$$

※ 구간 $[1, 2]$ 에서의 $y = \log x, y = e^{x-2}, y = \sin x$ 의 근사적분

구간을 몇 개의 소구간으로 나누고 이들 소구간의 각 끝점에서의 곡률들의 평균을 그 구간에서의 구간곡률이라고 하자. 소구간의 길이를 작게 할수록 곡선의 성질을 잘 표현할 수

<표 3.3.1>

(* Curvature *)	
f[x_]=Log[x]; a=1; b=2; n=1000; k=10;	
Curv[f_ over List, n_] =	
Module[{a=over[1], b=over[2], h}, h=(b-a)/n;	
Return[Sum[Abs[f'[a+h*i]]/((1+(f'[a+h*i])^2)^(3/2)), {i, 0, n}]]];	
Print["Averaged Curvature of ", OutputForm[f[x]],	
on [", a.", ", b.", "]:	
Print[" "];	
Print["Exact Value (n=", n, ") : ", N[Curv[f, {a, b}, n]/(n+1), k]]];	
Averaged Curvature of Log[x] on [1,2]	Exact Value (n=1000) : 0.2598995099
Averaged Curvature of Exp[x-2] on [1,2]	Exact Value (n=1000) : 0.3618160002
Averaged Curvature of Sin[x] on [1,2]	Exact Value (n=1000) : 0.8593460309

있게 된다.

<표 3.3.1>은 구간 [1, 2]에 대한 세 곡선의 구간곡률을 Mathematica를 이용하여 1000 개의 소구간으로 나누어 구한 것이다. 표에서 알 수 있듯이 세 곡선은 $y = \log x$, $y = e^{x-2}$, $y = \sin x$ 의 순서로 구간곡률이 커지고 있다.

<표 3.3.2>는 $\int_1^2 \log x \, dx$, $\int_1^2 e^{(x-2)} \, dx$, $\int_1^2 \sin x \, dx$ 의 값을 구하여 좌중점 공식(Ln), 사다리꼴 공식(Tn), 중간점 공식(Mn), Simpson 공식(Sn)으로 구분하여 표현한 것이다. 위 표를 관찰해 보면, 구간곡률이 클수록 근사적분값은 참값으로의 수렴 속도가 빠름을 알 수 있다. 즉, 곡선이 많이 될수록 근사적분값은 정적분값과의 오차가 훨씬 작아지고 있음을 실험적으로 알 수 있다.

<표 3.3.2>

Integration of $y = \log(x)$ on [1,2]								
Value : 0.38629				Curvature : 0.2599				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	0.29706	23.099%	0.38370	0.670%	0.38759	0.337%	0.38626	0.008%
8	0.34232	11.383%	0.38564	0.168%	0.38662	0.065%	0.38629	0.000%
16	0.36447	5.649%	0.38613	0.041%	0.38638	0.023%	0.38629	0.000%
32	0.37542	2.814%	0.38625	0.010%	0.38631	0.005%	0.38629	0.000%
64	0.38067	1.403%	0.38628	0.003%	0.38630	0.003%	0.38629	0.000%
Integration of $y = \exp(x-2)$ on [1,2]								
Value : 0.63212				Curvature : 0.3618				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	0.55639	11.980%	0.63541	0.520%	0.63048	0.259%	0.63213	0.002%
8	0.59344	6.119%	0.63294	0.130%	0.63171	0.065%	0.63212	0.000%
16	0.61257	3.093%	0.63233	0.033%	0.63202	0.016%	0.63212	0.000%
32	0.62230	1.564%	0.63217	0.008%	0.63209	0.005%	0.63212	0.000%
64	0.62719	0.780%	0.63213	0.002%	0.63211	0.002%	0.63212	0.000%
Integration of $y = \sin(x)$ on [1,2]								
Value : 0.95645				Curvature : 0.8593				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	0.94238	1.408%	0.95146	0.520%	0.95894	0.260%	0.95647	0.002%
8	0.95086	0.574%	0.95520	0.131%	0.95707	0.065%	0.95645	0.000%
16	0.95402	0.254%	0.95614	0.032%	0.95660	0.016%	0.95645	0.000%
32	0.95631	0.119%	0.95637	0.008%	0.95649	0.004%	0.95645	0.000%
64	0.95690	0.068%	0.95643	0.002%	0.95646	0.001%	0.95645	0.000%

※ 구간에 따른 $y = e^{x^2}$ 의 근사적분

동일한 곡선에서 상이한 구간곡률을 갖는 구간에서의 근사적분의 수렴속도를 알아보기 위해 함수 $y = e^{x^2}$ 의 구간곡률을 Mathematica를 이용하여 세 구간 [0, 0.5], [0.5, 1], [1, 1.5]을 각각

1000 개의 소구간으로 나누어 구하여보면 <표 3.3.3>과 같다.

<표 3.3.3>

```
(• Curvature •)
f[x_]:=Exp[x^2]; a=0.5; b=1; n=1000; k=10;
Curv[f_,over_List,n_]:=
Module[{a=over[[1]],b=over[[2]],h=(b-a)/n;
Return[Sum[Abs[f'[a+h*i]]/(1+(f'[a+h*i])^2)^(3/2),(i,0,n)]];
Print["Averaged Curvature of ",OutputForm[f[x]],
" on [" ,a," ,",b," ,"]"];
Print[" " ];
Print["Exact Value (n=",n,") : ", N[Curv[f,{a,b},n]/(n+1),k]];
```

Averaged Curvature of Exp[x^2] on [0,0.5]	Exact Value (n=1000) : 1.5777907222
Averaged Curvature of Exp[x^2] on [0.5,1]	Exact Value (n=1000) : 0.3891853682
Averaged Curvature of Exp[x^2] on [1,1.5]	Exact Value (n=1000) : 0.0317843695

또한 이들 구간에 대한 함수 $y = e^{x^2}$ 의 근사적분을 수치적분 프로그램인 Integral을 이용하여 구하면 위의 <표 3.3.4>를 얻을 수 있다.

<표 3.3.4>

Integration of $y = \exp(x^2)$ on [0,0.5]								
Value : 0.54499				Curvature : 1.5778				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	0.52890	2.952%	0.54666	0.306%	0.54415	0.154%	0.54500	0.002%
8	0.53653	1.552%	0.54540	0.075%	0.54478	0.039%	0.54499	0.000%
16	0.54065	0.796%	0.54509	0.018%	0.54493	0.011%	0.54499	0.000%
32	0.54279	0.404%	0.54501	0.004%	0.54497	0.004%	0.54499	0.000%
64	0.54388	0.204%	0.54500	0.002%	0.54499	0.000%	0.54499	0.000%
Integration of $y = \exp(x^2)$ on [0.5,1]								
Value : 0.91766				Curvature : 0.3892				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	0.83341	9.181%	0.92306	0.588%	0.91497	0.293%	0.91772	0.007%
8	0.87419	4.737%	0.91902	0.148%	0.91699	0.073%	0.91767	0.001%
16	0.89560	2.404%	0.91803	0.037%	0.91750	0.017%	0.91766	0.000%
32	0.90654	1.212%	0.91775	0.010%	0.91762	0.004%	0.91766	0.000%
64	0.91208	0.608%	0.91769	0.003%	0.91765	0.001%	0.91766	0.000%
Integration of $y = \exp(x^2)$ on [1,1.5]								
Value : 2.60046				Curvature : 0.0318				
n	Ln	%Ln	Tn	%Tn	Mn	%Mn	Sn	%Sn
4	2.20723	15.122%	2.63032	1.148%	2.58558	0.572%	2.60095	0.019%
8	2.33640	7.847%	2.60795	0.288%	2.59672	0.144%	2.60049	0.001%
16	2.49556	3.995%	2.60233	0.072%	2.59953	0.036%	2.60046	0.000%
32	2.54904	2.016%	2.60093	0.018%	2.60023	0.009%	2.60046	0.000%
64	2.57414	1.012%	2.60058	0.006%	2.60040	0.002%	2.60046	0.000%

위의 경우에도 표를 관찰해 보면, 구간곡률이 클수록 근사적분값은 참값으로의 수렴 속도가 빠름을 알 수 있다. 즉, 곡선의 구간곡률은 근사적분값의 오차에 영향을 주고 있다는 추측

을 할 수 있다.

3.4. 수학교실에서의 컴퓨터 활용

학습환경의 개선으로 인한 수학교실에서의 컴퓨터와 Mathematica를 비롯한 여러 가지의 수학 패키지를 활용한 교수·학습의 예를 살펴보기로 하자. <표 3.4.1>과 <표 3.4.2>는 Mathematica(<표 3.1.1> 참조)를 이용하여 곡선 $f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$ 의 근사값들을 n 의 값을 증가시키면서 좌종점 공식(Ln), 우종점 공식(Rn), 사다리꼴 공식(Tn), 중간점 공식(Mn), Simpson 공식(Sn)으로 구분하고, 각 근사값과 그의 오차 및 오차율을 다음과 같이 계산하여 표현한 것이다.

<표 3.4.1>

Check Values for Numerical Integration:	
Integration of	
$0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$	
on [0,1] with n=16 :	
Exact Value	: 1.45
LH Sum (n=16)	: 1.458087158
RH Sum (n=16)	: 1.458087158
Midpoint Sum (n=16)	: 1.445975494
Trapezoidal Sum (n=16)	: 1.458087158
Simpson Sum (n=16)	: 1.450203451

$X = L_n, R_n, M_n, T_n, S_n$ 일 때, 각 근사값의 오차와 오차율을

$$EX = X - \int_1^2 \frac{1}{x} dx ,$$

$$\%X = \left\{ |EX| / \int_1^2 \frac{1}{x} dx \right\} \times \frac{1}{100} (\%)$$

로 나타내기로 하자.

위에 주어진 함수에 대해서는 좌종점 근사값(Ln), 우종점 근사값(Rn)이 서로 같으므로

$T_n = \frac{1}{2} (L_n + R_n)$ 의 관계에 의하여 사다리꼴 근사값도 일치한다는 것을 쉽게 확인할 수 있다.

<표 3.4.2>

f(x)=0.2+25x+200x^2+675x^3-900x^4+400x^5 from 0 to 1					1.45000000
n	Ln	Rn	Mn	Tn	Sn
4	1.56718750	1.56718750	1.39628906	1.56718750	1.50208333
8	1.48173828	1.48173828	1.43443604	1.48173828	1.45325521
16	1.45808716	1.45808716	1.44597549	1.45808716	1.45020345
32	1.45203133	1.45203133	1.44898653	1.45203133	1.45001272
64	1.45050843	1.45050843	1.44974586	1.45050843	1.45000079
128	1.45012714	1.45012714	1.44993643	1.45012714	1.45000005
n	ELn	ERn	EMn	ETn	ESn
4	-0.11718750	-0.11718750	0.05371094	-0.11718750	-0.05208333
8	-0.03173828	-0.03173828	0.01556396	-0.03173828	-0.00325521
16	-0.00808716	-0.00808716	0.00402451	-0.00808716	-0.00020345
32	-0.00203133	-0.00203133	0.00101447	-0.00203133	-0.00001272
64	-0.00050843	-0.00050843	0.00025414	-0.00050843	-0.00000079
128	-0.00012714	-0.00012714	0.00006357	-0.00012714	-0.00000005
n	%Ln	%Rn	%Mn	%Tn	%Sn
4	8.08190%	8.08190%	3.70420%	8.08190%	3.59195%
8	2.18885%	2.18885%	1.07338%	2.18885%	0.22450%
16	0.55774%	0.55774%	0.27755%	0.55774%	0.01403%
32	0.14009%	0.14009%	0.06996%	0.14009%	0.00088%
64	0.03506%	0.03506%	0.01753%	0.03506%	0.00005%
128	0.00877%	0.00877%	0.00438%	0.00877%	0.00000%

그러나 <표 3.4.2>을 통하여 중간점 근사값(Mn)은 사다리꼴 근사값(Tn) 보다 더 정확하다는 것을 다시 한 번 확인할 수 있다. 현행 고등학교 교육과정에서는 계산의 복잡성 때문에 중간점 근사값(Mn)을 다루지 않지만, 시대의 변천에 따른 교실환경의 개선으로 컴퓨터를 수학학습에 활용하면 보다 쉽게 학생들로 하여금 수치적분에 의한 정적분값의 계산과 극한의 개념을 파악하게 할 수 있다. 또한 그리기 어려운 여러 가지 함수의 그래프를 수학 패키지를, 활용한 교수·학습에서는 단순한 조작으로 쉽게 해결 할 수 있어 수학에 대한 흥미를 갖게 하여 생기있고 활기찬 수학교실을 이끌 수 있다.

<표 3.4.3-1>

(* IntegralGraphics *)

```
(*
rightrect::usage="rightrect[f[x],{x,a,b,n}] draws right-hand
endpoint rectangles."
leftrect::usage="leftrect[f[x],{x,a,b,n}] draws left-hand
endpoint rectangles."
midrect::usage="midrect[f[x],{x,a,b,n}] draws mid-point
rectangles."
trapezoid::usage="trapezoid[f[x],{x,a,b,n}] draws trapezoids."
error::usage="error[estimate] computes the error between
estimate and the area of the region under consideration."
*)

box[x_.,y_.,h_] = Graphics[{RGBColor[0,0,1],
  Line[{x, 0},{x, y},
    {x + h, y},
    {x + h, 0}]}];

rightrect[func_.,paramlist_] :=
  Module[{boxes, curve, k, a, b, h, n, param, x},
    param = paramlist[[1]];
    a = N[paramlist[[2]]];
    b = N[paramlist[[3]]];
    n = N[paramlist[[4]]];
    h = (b - a)/n;

    boxes = Table[box[a + k h,func /. param->(a + k h),-h],
      {k,1,n}];
    curve = Plot[func /. param -> x,{x,a,b},
      PlotStyle->RGBColor[1,0,0],DisplayFunction->Identity];
    Show[{boxes,curve}, DisplayFunction -> $DisplayFunction,
      Axes -> True, AxesOrigin -> {a,0}]
  ]
```

```
leftrect[func_.,paramlist_] :=
  Module[{boxes, curve, k, a, b, h, n, param, x},
    param = paramlist[[1]];
    a = N[paramlist[[2]]];
    b = N[paramlist[[3]]];
    n = N[paramlist[[4]]];
    h = (b - a)/n;
    boxes = Table[box[a + k h,func /. param->(a + k h),h],
      {k,0,n-1}];
    curve = Plot[func /. param -> x,{x,a,b},
      PlotStyle->RGBColor[1,0,0],DisplayFunction->Identity];
    Show[boxes,curve, DisplayFunction -> $DisplayFunction,
      Axes -> True, AxesOrigin -> {a,0}]
  ]
```

<표 3.4.3-2>

```
midrect[func_.,paramlist_] :=
  Block[{boxes, curve, k, a, b, h, n, param, x},
    param = paramlist[[1]];
    a = N[paramlist[[2]]];
    b = N[paramlist[[3]]];
    n = N[paramlist[[4]]];
    h = (b - a)/n;
    boxes = Table[box[a + k h,func /. param->(a + (k-0.5)h),
      -h], {k,1,n}];
    curve = Plot[func /. param -> x,{x,a,b},
      PlotStyle->RGBColor[1,0,0],DisplayFunction->Identity];
    Show[boxes,curve, DisplayFunction -> $DisplayFunction,
      Axes -> True, AxesOrigin -> {a,0}]
  ]

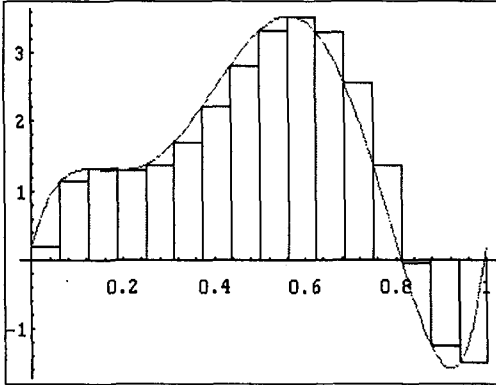
trap[x0_.,y0_.,x1_.,y1_] = Graphics[{RGBColor[0,0,1],
  Line[{(x0,0),(x0,y0),(x1,y1),(x1,0)}]}];

trapezoid[func_.,paramlist_] :=
  Module[{boxes, curve, k, a, b, h, n, param, x},
    param = paramlist[[1]];
    a = N[paramlist[[2]]];
    b = N[paramlist[[3]]];
    n = N[paramlist[[4]]];
    h = (b - a)/n;
    boxes = Table[trap[a + k h,func /. param ->(a + k h),
      a + (k+1)h,func /. param ->(a + (k+1)h)],{k,0,n-1}];
    curve = Plot[func /. param -> x,{x,a,b},
      PlotStyle->RGBColor[1,0,0],DisplayFunction->Identity];
    Show[boxes,curve, DisplayFunction -> $DisplayFunction,
      Axes -> True, AxesOrigin -> {a,0}]
  ]

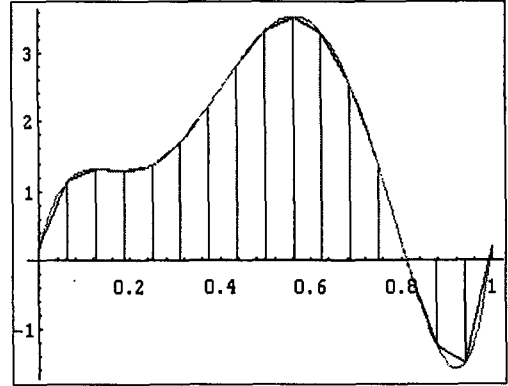
error[est_] := est
```

<표 3.4.3>은 Mathematica를 이용하여 수치 적분에서 활용하는 좌중점 공식(Ln), 우중점 공식(Rn), 사다리꼴 공식(Tn), 중간점 공식(Mn)을 시각적으로 표현할 수 있는 수리 패키지이며, <그림 3.4.4a~3.4.4d>는 그 출력의 결과이다.

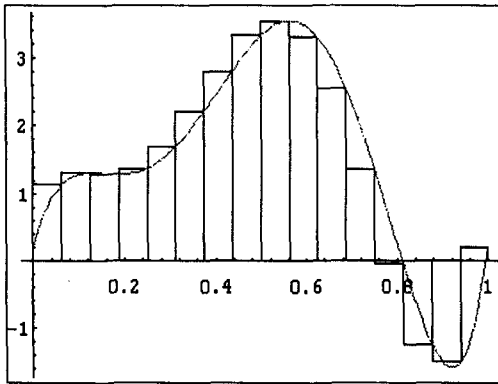
※ $f(x) = 0.2 + 25x + 200x^2 + 675x^3 - 900x^4 + 400x^5$



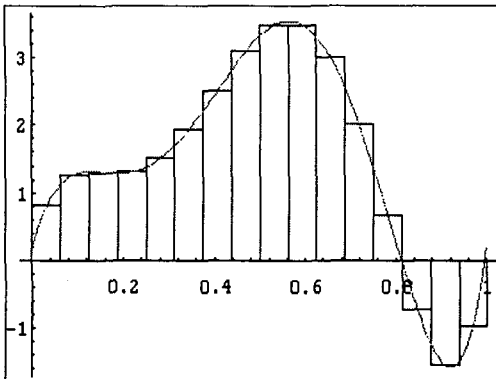
<그림 3.4.4a> leftrect[f(x),{x,0,1,16}]



<그림 3.4.4d> trapezoid[f(x),{x,0,1,16}]



<그림 3.4.4b> rightrect[f(x),{x,0,1,16}]



<그림 3.4.4c> midrect[f(x),{x,0,1,16}]

4. 결론

수학교육의 목적은 수학적으로 사고하는 능력을 계발하는 것이라고 할 수 있다. 그러나 직관적인 사고로 개념을 이해시키려 하는 해석적인 방법은, 개념을 형상화하기가 쉽지 않아 학생들로 하여금 빨리 흥미를 잃게 한다. 수학교사들이, 비록 학생 스스로의 탐구와 사고활동을 통해 수학공부에 대한 내재적 동기를 갖게 하려고 발표와 토론을 많이 시킨다든지 사고를 요하는 질문을 자주 하려고는 하지만, 이를 달갑게 여기지 않는 학생들의 반응에 부딪혀 곧 좌절하곤 한다.

본 논문은 이러한 문제를 조금이나마 해소하고자 학생들이 스스로 참여하고 해결하는 수치해석적 방법을 도입하여 정적분으로의 실험적 접근을 모색하여 보았다. 컴퓨터를 활용하여 문제를 해결하고자 하는 수치해석적 방법은 최근 급격히 강화되고 있는 실정이다. 여러 가지 수리패키지를 적절히 활용하면, 교실현장은 생기가 넘치는 살아있는 교육의 장이 될 것이다. 그러한 가운데 학생들은 수학적 아이디어를 더욱 발전시키고 중요한 수학적 개념을 쉽게 이해하거나 스스로 발견하게 되는 것이다.

본 논문에서 고찰한 결과에 의하면, 현실에 맞는 교육과정의 개선이 필요하다고 할 수 있

다. 현행 고등학교 수학 교육과정에서는 구분구적법을 통한 정적분의 도입에 있어서 좌종점 근사값이나 우종점 근사값을 사용하고 있다. 그러나 본 논문에서 밝힌 바와 같이 학생들로 하여금 충분히 이해할 수 있는 중간점 근사값이, 좌종점 근사값, 우종점 근사값, 심지어는 사다리꼴 근사값보다 더 정밀하다. 지필 환경에서는 구간의 중간점을 잡고 그의 함수값을 구하는 일이 쉽지 않았지만, 컴퓨터가 교실 현장까지 보급되고 있는 현실에서는 까다로운 계산에 구애받지 않으므로 Riemann 적분 이론으로의 접근을 쉽게 할 수 있는 중간점 근사값을 이용하는 수학 교육과정의 개선 노력이 필요하다고 할 수 있다. 더 나아가서는 Simpson 방식에 의한 접근도 가능한 것이다.

참 고 문 헌

- 강시중 (1985). 수학교육론, 교육출판사.
- 김두호 외 1인 (1992). 미분적분학, 청문각.
- 류계구 (1995). 매스매티카, 크라운출판사.
- 변정선 (1991). 구분구적법을 통해 본 정적분 개념, 이화여자대학교 교육대학원, 석사학위 논문.
- 송수양 (1979). 적분법에 관한 연구, 고려대학교 교육대학원, 석사학위 논문.
- Hae-Soo Oh (1992). Use of computer as a tool of Mathematics Education, *Mathematics Education*, Korean Mathematical Society, Vol 10.
- Stewart, J. (1996). 수학교재편찬위원회 역, 미분적분학, 청문각.
- Burden, R. L. 송만수·장건수 역 (1986). 수치해석학, 김영사.
- R. G. Bartle & D. R. Sherbert 김용태 역 (1987). 실해석학, 서울: 경문사.
- Wolfram, S. (1994). *Mathematica*, Addison Wesley.

Experimental Approaches to Definite Integration with Numerical Integration

Joa, Chang-Bo

Nam-nyeong High School, 2288, Yeon-dong, Cheju-si, Cheju-do, Koea.

Kim, Chul-Soo

Dept. of Computer Sci. & Stat., College of Natural Sci., Cheju National Univ.

1 Ara 1-Dong, Cheju, Cheju-Do 690-756, Korea; email: cskim@cheju.cheju.ac.kr

In this thesis, We tried to introduce definite integration to the curriculum of high school mathematics with numerical integration, which had been introduced with quadrature method. For this purpose, We used new experimental mathematics approaches, so-called investigation and examination.

In chapter II, We examined how much computers had been used in teaching mathematics. In chapter III, We presented the theoretical background of approximation integration within numerical integration. In chapter IV, We studied and compared various methods of numerical integration, and examined the relation between curvature of a curved line and numerical integration. In order to study more easily, We used some of computer programs.

We hope that this thesis will be a turning point in developing new teaching methods and improving curriculum of mathematics in high school.