

3차원 벡터필드 탄젠트 곡선 계산을 위한 사면체 분해 방법

정일홍*

요약

본 논문에서는 3차원 벡터필드의 탄젠트 곡선을 계산하는 효율적이고 정확한 방법을 제안한다. 탄젠트 곡선 상의 정확한 값을 구하지 못하고 단지 탄젠트 곡선의 근사치를 구하는 Runge-Kutta 같은 기존의 방법과는 달리 여기서 제안한 방법은 3D 사면체 도메인에서 벡터필드가 선형적으로 변한다는 가정하에 탄젠트 곡선 상의 정확한 값을 계산한다.

새로 제안한 방법은 벡터필드가 3D 사면체 도메인에서 선형적으로 변한다고 가정한다. 우선 이 방법은 3차원 벡터필드에서 육면체 셀을 5 또는 6개의 사면체 셀로 분해하는 것을 요구한다. 임계점은 각 사면체의 간단한 선형 시스템을 풀어서 간단하게 구할 수 있다. 이 방법은 이전 사면체에서 계산된 탄젠트 곡선상의 점들을 기초로 다음 사면체에서 탄젠트 곡선상의 연속적인 점들을 생성함으로써 출구 점을 구한다. 탄젠트 곡선상의 점들은 각 사면체의 명시해에 의해서 계산되었기 때문에 새로운 방법은 3D 벡터필드를 가시화하는데 정확한 위상을 마련한다.

키워드 : 벡터필드, 명시해, 위상, 탄젠트 곡선, 사면체

A Tetrahedral Decomposition Method for Computing Tangent Curves of 3D Vector Fields

Il-Hong Jung*

Abstract

This paper presents the development of certain highly efficient and accurate method for computing tangent curves for three-dimensional vector fields. Unlike conventional methods, such as Runge-Kutta method, for computing tangent curves which produce only approximations, the method developed herein produces exact values on the tangent curves based upon piecewise linear variation over a tetrahedral domain in 3D.

This new method assumes that the vector field is piecewise linearly defined over a tetrahedron in 3D domain. It is also required to decompose the hexahedral cell into five or six tetrahedral cells for three-dimensional vector fields. The critical points can be easily found by solving a simple linear system for each tetrahedron. This method is to find exit points by producing a sequence of points on the curve with the computation of each subsequent point based on the previous. Because points on the tangent curves are calculated by the explicit solution for each tetrahedron, this new method provides correct topology in visualizing 3D vector fields.

Keywords : vector field, explicit solution, topology, tangent curves, tetrahedron

1. 서론

※ Corresponding Author : Il-Hong Jung
Received: January 01, 2015
Revised: February 20, 2015
Accepted: February 28, 2015
* Department of Computer Engineering, Daejeon University
Tel: +82-42-280-2548 , Fax: +82-42-280-2889
email: ijung@dju.kr

벡터 필드(vector field)의 효율적인 시각적 표현 방법은 유체 역학, 해양 연구, 기온 변화 등에서의 연구에 중요한 역할을 담당하고 있기 때문에 오늘날 매우 중요한 연구과제이다. 벡터 필드를 시각적으로 표현하기 위해서는 탄젠트 곡선의 정확한 계산에 대한 연구가 필수적이다[1].

벡터 필드를 시각적으로 표현하기 위해 사용하는 위상(topology) [2,3,4] 은 임계점(critical point)과 탄젠트 곡선(tangent curve)으로 구성되어 있다[5]. 위상적 표현의 장점은 벡터필드의 임계점과 임계점들을 연결하는 탄젠트 곡선이 주어지면, 다른 탄젠트 곡선의 형태와 벡터 필드 전체의 구조를 어느 정도 추론할 수 있다. 기존의 위상적 방법에서는 임계점의 위치 계산은 상당한 문제점을 가지고 있다. 예를 들어, 곡선 그리드(curvilinear grid)의 경우 임계점을 계산하기 위해 먼저 가능한 후보자 셀(cell)를 찾아 비선형 방정식을 풀어야만 한다[2,3,4,5]. 그리고 탄젠트 곡선은 Euler 방법이나 Runge-Kutta 방법 같은 수치 해석적 방법을 사용하여 구하기 때문에 탄젠트 곡선 상의 정확한 값을 구하지 못하고 단지 탄젠트 곡선의 근사치를 구했다[6,7,8]. 이로 인해 잘못된 탄젠트 곡선을 계산하는 오류를 범하는 경우가 생긴다.

본 논문에서 제안하는 탄젠트 곡선을 계산하는 방법은 탄젠트 곡선을 정의하는 상미분 방정식의 명시해를 데카르트 좌표계에서 구하기 때문에 탄젠트 곡선 상에 정확한 값을 계산할 수 있다. 육면체 그리드 셀을 5 또는 6개의 사면체로 분해 한 후 사면체에 대해 제안한 방법으로 상미분 방정식의 명시해를 계산한다. 그리고 사면체의 명시해에 의해 그려지는 곡선과 사면체 한 면을 포함하는 평면과의 교차점을 계산하여 탄젠트 곡선을 연결시킨다. 연결된 탄젠트 곡선으로 3차원 벡터 필드의 위상을 시각적으로 표현한다.

본 논문의 구성으로 2장에서는 탄젠트 곡선을 정의하는 상미분 방정식의 명시해(explicit solution)를 구하기 위해 필요한 수학적 배경에 대해 설명하고 3장에서는 3차원 벡터필드에 대한 상미분 방정식을 풀어 구한 명시해에 대해 서술하고 4장에서는 제안한 방법과 기존의 수치 해석적 방법과의 비교분석 및 성능을 검토한다. 끝으로 5장에서는 결론 및 향후 연구 과제를 제시한다.

2. 수학적 배경

본 논문에서 제안하는 3차원 벡터 필드의 가시화 방법은 상미분 방정식(ordinary differential

equation)과 벡터필드의 위상과의 연결에 기초를 두고 있다[5,6]. 상미분 방정식과 벡터 필드의 위상과의 연결은 Poincare [7,8,9]가 처음으로 제안했다. 벡터필드의 가장 중요한 두 가지는 임계점과 탄젠트 곡선을 계산하는 방법이다. 본 논문의 주안점은 3차원(사면체)의 벡터 필드에 대한 탄젠트 곡선의 계산이다. 본 논문에서 제안하는 방법의 기초가 되는 중요한 개념은 3D에서 육면체 셀을 5 또는 6개의 사면체들의 집합으로 분해하는 것과 각 사면체내에서 벡터 필드가 선형적으로 변화를 한다고 하는 가정이다[10,11,12].

이러한 가정 하에서 탄젠트 곡선을 정의하는 방정식은 상수 계수를 갖는 선형 상미분 방정식이다. 벡터 필드가 다음과 같이 주어지면

$$V(x, y, z) = \begin{bmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{bmatrix} = A \begin{bmatrix} x \\ y \\ z \end{bmatrix} + B,$$

$$\text{여기서 } A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \text{이다.}$$

사면체에서 매개 변수 탄젠트 곡선 $P(t)$ 는

$$P'(t) = V(P(t)) = \begin{bmatrix} x'(t) \\ y'(t) \\ z'(t) \end{bmatrix} = A \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} + B \quad (1)$$

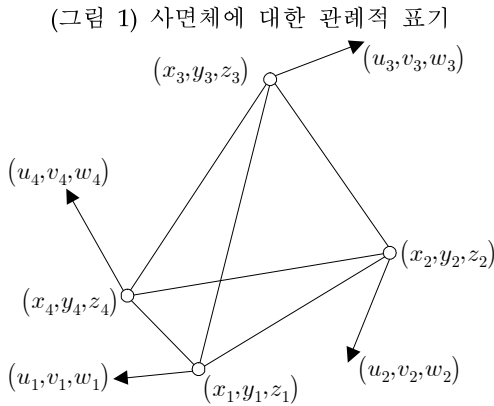
이고 초기 값은 $P(0) = \begin{bmatrix} x(0) \\ y(0) \\ z(0) \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$ 이다. 일

반적으로 상미분 방정식은 무수한 해가 존재하나 초기 값을 갖는 상미분 방정식은 특별한 해가 존재한다[8].

만약에 (그림 1)에서 보여주는 것과 같이 각 사면체의 좌표 값을 (x_1, y_1, z_1) , (x_2, y_2, z_2) , (x_3, y_3, z_3) , (x_4, y_4, z_4) 그리고 각 꼭짓점에서 벡터 값을 (u_1, v_1, w_1) , (u_2, v_2, w_2) , (u_3, v_3, w_3) , (u_4, v_4, w_4) 라 가정하면, 사면체에 대한 3×3 행렬 A 와 3×1 행렬 B 는 사면체의 꼭짓점들과 그 점에서의 벡터필드의 값으로부터 구해진다.

임계점은 벡터의 크기가 제로가 되는 점이므로 간단한 선형방정식 $AP_c + B = 0$ 을 풀므로써 간

단히 구해질 수 있다.



(그림 1) 사면체에 대한 관례적 표기

(Figure 1) Notation and conventions for a tetrahedron

3. 3D 벡터 필드의 명시해 및 사면체 분해 방법

이 장에서는 탄젠트 곡선을 정의하는 상미분 방정식을 데카르트 좌표계에서 일반적인 해에 대해 서술하고 고유값에 따라 9가지 경우로 분류하고 그에 대해 명시해와 사면체 분해 방법의 알고리즘을 제시한다.

3.1 사면체 도메인에 대한 명시해

탄젠트 곡선을 정의하는 상미분 방정식 (1)의 일반해는 다음의 형태를 가진다.

$$P(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = E_1\Phi_1(t) + E_2\Phi_2(t) + E_3\Phi_3(t) + C$$

여기서 함수 Φ_1, Φ_2, Φ_3 와 계수 E_1, E_2, E_3 는 행렬 A 의 고유값(eigenvalue) [13]에 따라 결정된다. 3차원 벡터필드에서 이러한 고유값에 따라 탄젠트 곡선의 궤적을 9가지로 분류한다. 9가지 경우에 대한 분류와 그에 따른 명시해는 다음과 같이 구하였다.

Case 1) 제로가 아닌 세 개의 실근
($0 \neq r_1 \neq r_2 \neq r_3 \neq 0$)

$$P(t) = E_1e^{tr_1} + E_2e^{tr_2} + E_3e^{tr_3} + P_c$$

$$E_1 = \left(\frac{A - r_2I}{r_1 - r_2} \right) \left(\frac{r_3I - A}{r_3 - r_1} \right) (P_0 - P_c),$$

$$E_2 = \left(\frac{A - r_3I}{r_2 - r_3} \right) \left(\frac{r_1I - A}{r_1 - r_2} \right) (P_0 - P_c),$$

$$E_3 = \left(\frac{A - r_1I}{r_3 - r_1} \right) \left(\frac{r_2I - A}{r_2 - r_3} \right) (P_0 - P_c),$$

$$AP_c + B = 0$$

Case 2) 제로가 아닌 한 개의 실근과 두 개의 중근 ($0 \neq r_1 \neq r_2 = r_3 \neq 0$)

$$P(t) = E_1e^{tr_1} + E_2e^{tr_2} + E_3te^{tr_2} + P_c$$

$$E_1 = \left(\frac{A - r_2I}{r_2 - r_1} \right)^2 (P_0 - P_c),$$

$$E_2 = \frac{(r_1I - A)(r_1I - 2r_2I + A)}{(r_2 - r_1)^2} (P_0 - P_c),$$

$$E_3 = \frac{(A - r_1I)(A - r_2I)}{(r_2 - r_1)} (P_0 - P_c),$$

$$AP_c + B = 0$$

Case 3) 제로가 아닌 세 개의 중근
($r_1 = r_2 = r_3 \neq 0$)

$$P(t) = E_1e^{tr_1} + E_2te^{tr_1} + E_3t^2e^{tr_1} + P_c$$

$$E_1 = P_0 - P_c,$$

$$E_2 = (A - r_1I)(P_0 - P_c),$$

$$E_3 = \frac{(A - r_1I)^2}{2} (P_0 - P_c),$$

$$AP_c + B = 0$$

Case 4) 제로가 아닌 한 개의 실근과 두 개의 허근 ($r_1 \neq 0, \mu \pm \lambda i, \lambda \neq 0$)

$$P(t) = E_1e^{tr_1} + E_2e^{\mu t} \cos(\lambda t) + E_3e^{\mu t} \sin(\lambda t) + P_c,$$

$$E_1 = \frac{(A - \mu I)^2 + \lambda^2}{(\mu - r_1)^2 + \lambda^2} (P_0 - P_c),$$

$$E_2 = \frac{(\mu I - A)(r_1I - 2\mu I + A)}{(\mu - r_1)^2 + \lambda^2} (P_0 - P_c),$$

$$E_3 = \frac{(A - r_1I)((A - \mu I)(\mu - r_1) + \lambda^2 I)}{\lambda((\mu - r_1)^2 + \lambda^2)} (P_0 - P_c),$$

$$AP_c + B = 0.$$

Case 5) 제로가 아닌 두 개의 실근과 한 개가 제로인 실근 ($r_1 = 0, 0 \neq r_2 \neq r_3 \neq 0$)

$$P(t) = E_1 t + E_2 (e^{tr_2} - 1) + E_3 (e^{tr_3} - 1) + P_0$$

$$E_1 = \left(\left(\frac{A - r_3 I}{r_2 - r_3} \right) \left(I - \frac{A}{r_2} \right) + \left(\frac{r_2 I - A}{r_2 - r_3} \right) \left(I - \frac{A}{r_3} \right) \right) B,$$

$$E_2 = \left(\frac{A - r_3 I}{r_2 - r_3} \right) \frac{A}{r_2} \left(P_0 + \frac{B}{r_2} \right),$$

$$E_3 = \left(\frac{r_2 I - A}{r_2 - r_3} \right) \frac{A}{r_3} \left(P_0 + \frac{B}{r_3} \right),$$

Case 6) 제로가 아닌 한 개의 실근과 두 개가 제로인 실근 ($r_1 = r_2 = 0, r_3 \neq 0$)

$$P(t) = E_1 t + E_2 t^2 + E_3 (e^{tr_3} - tr_3 - 1) + P_0$$

$$E_1 = AP_0 + B,$$

$$E_2 = \left(I - \frac{A}{r_3} \right) \frac{AB}{2},$$

$$E_3 = \left(\frac{A}{r_3} \right)^2 \left(P_0 + \frac{B}{r_3} \right).$$

Case 7) 세 개가 제로인 중근 ($r_1 = r_2 = r_3 = 0$)

$$P(t) = E_1 t + E_2 t^2 + E_3 t^3 + P_0$$

$$E_1 = AP_0 + B,$$

$$E_2 = \frac{A}{2} (AP_0 + B),$$

$$E_3 = \frac{A^2 B}{6}.$$

Case 8) 제로가 아닌 두 개의 중근과 한 개가 제로인 실근 ($r_1 = 0, r_2 = r_3 \neq 0$)

$$P(t) = E_1 t + E_2 (e^{tr_2} - 1) + E_3 t e^{tr_2} + P_0$$

$$E_1 = \left(I + \frac{A}{r_2} \left(\frac{A}{r_2} - 2I \right) \right) B,$$

$$E_2 = \frac{A}{r_2} \left(2 \left(I - \frac{A}{r_2} \right) \left(P_0 + \frac{B}{r_2} \right) + \frac{AP_0 + B}{r_2} \right),$$

$$E_3 = A \left(\frac{A}{r_2} - I \right) \left(P_0 + \frac{B}{r_2} \right).$$

Case 9) 제로인 한 개의 실근과 두 개의 허근 ($r_1 = 0, \mu \pm \lambda i, \lambda \neq 0$)

$$P(t) = E_1 t + E_2 (e^{\mu t} \cos(\lambda t) - 1) + E_3 e^{\mu t} \sin(\lambda t) + P_0$$

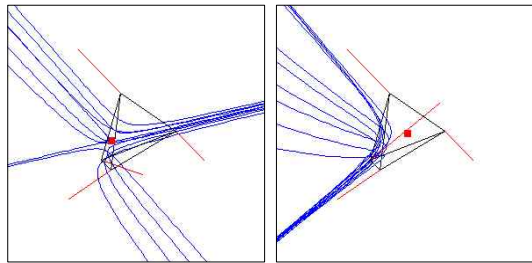
$$E_1 = \left(I - A \left(\frac{2\mu - A}{\mu^2 + \lambda^2} \right) \right) B,$$

$$E_2 = \left(\frac{A}{\mu^2 + \lambda^2} \right) \left((2\mu I - A) P_0 + \left(\frac{3\mu^2 - \lambda^2}{\mu^2 + \lambda^2} I - 2\mu A \right) B \right),$$

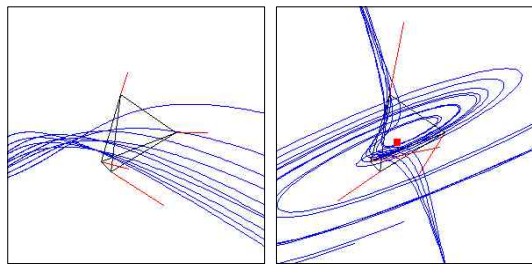
$$E_3 = \left(\frac{A}{\lambda(\mu^2 + \lambda^2)} \right) \left((\mu A - (\mu^2 - \lambda^2) I) P_0 + \left(\frac{(\mu^2 - \lambda^2) A - \mu(\mu^2 - 3\lambda^2) I}{\mu^2 + \lambda^2} \right) B \right)$$

(그림 2)는 위의 아홉 가지 경우의 탄젠트 곡선의 궤적을 보여준다. (그림 2)의 각 이미지의 탄젠트 곡선은 한 면에서 10개의 초기 값을 사용하여 계산된 것이다. 이 초기 점들에서 시작해서 일정한 양만큼 양의 방향 및 음의 방향으로 탄젠트 곡선 상의 점들을 계산한다. 파란 선은 탄젠트 곡선을 나타내고, 빨간 선은 사면체의 각 정점에서의 벡터 값을 나타낸다. 빨간 점은 임계점(Critical Point)을 표시한다.

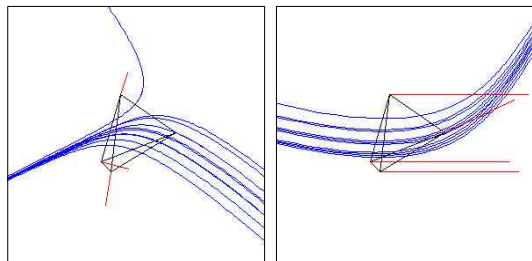
(그림 2) 9가지 경우에서 탄젠트 곡선의 예

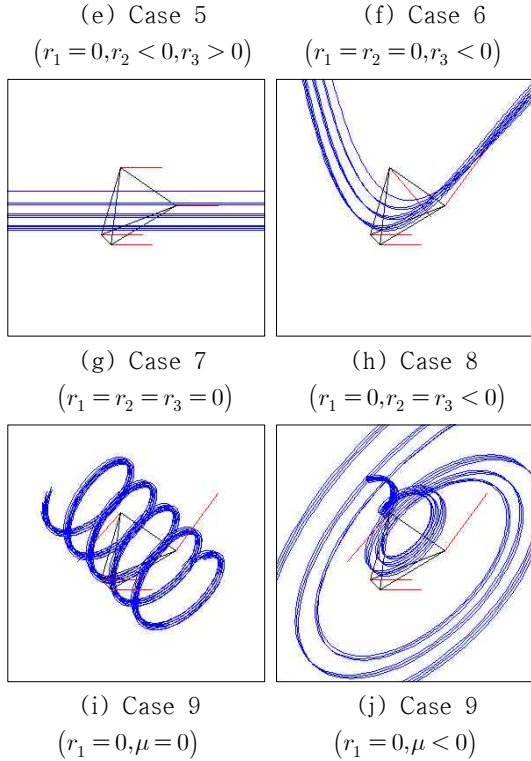


(a) Case 1 ($r_1 < 0, r_2 \neq r_3 > 0$) (b) Case 2 ($r_1 < 0, r_2 = r_3 > 0$)



(c) Case 3 ($r_1 = r_2 = r_3 > 0$) (d) Case 4 ($r_1 > 0, \mu < 0$)





(Figure 2) Examples of tangent curves for all nine cases

3.2 사면체 분해 방법

위에서 제시한 명시해로 탄젠트 곡선을 계산하고 3차원 벡터필드의 위상을 시각적으로 표현하기 위한 사면체 분해 방법의 알고리즘을 다음과 같다

- (1) 육면체 그리드 셀을 5 또는 6개의 사면체 셀로 분해한다.
- (2) 임계점의 좌표 값을 구한다.
- (3) 임계점을 분류한다.
- (4) 각 사면체의 고유값(eigenvalue)을 구한다.
- (5) 사면체를 9가지 경우로 분류한다.
- (6) 각 사면체에 대한 명시해를 계산한다.
- (7) 사면체의 명시해에 의해 그려지는 탄젠트 곡선과 사면체의 한 면을 포함하는 평면과 교차점을 구한다.
- (8) 탄젠트 곡선을 연결시킨다.
- (9) 3차원 벡터필드의 위상을 시각적으로 표시한다.

4. 성능 평가

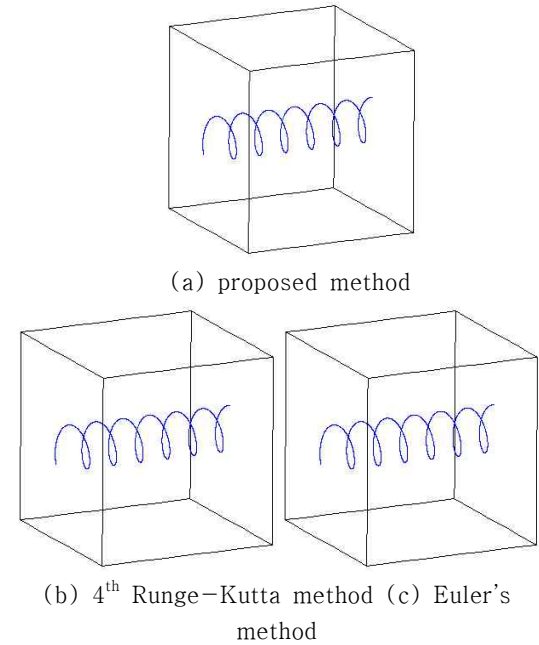
본 논문에서는 제안한 방법의 정확도를 검증하기 위해 테스트 함수를 구현하였다. 또한 기존 방법인 4th Runge-Kutta 방법과 Euler 방법을 구현하여 본 논문에서 제안한 방법과 비교를 하였다. (그림 3)에서 사용한 3D 테스트 함수는 나선형류(helical flow)로 다음과 같다.

$$u = -y, v = x, w = 1$$

이 함수는 10×10×10 셀의 데카르트 좌표 그리드에서 각 그리드 점에서 벡터 값을 생성한다.

(그림 3)에서 보여주는 것과 같이 데카르트 좌표에서 구현된 제안된 방법과 기존의 방법으로 구현된 탄젠트 곡선을 비교해보면 본 논문에서 제안한 방법이 정확하다는 것이 검증된 것이다.

(그림 3) 기존의 방법과 비교를 통해 제안한 방법의 검증

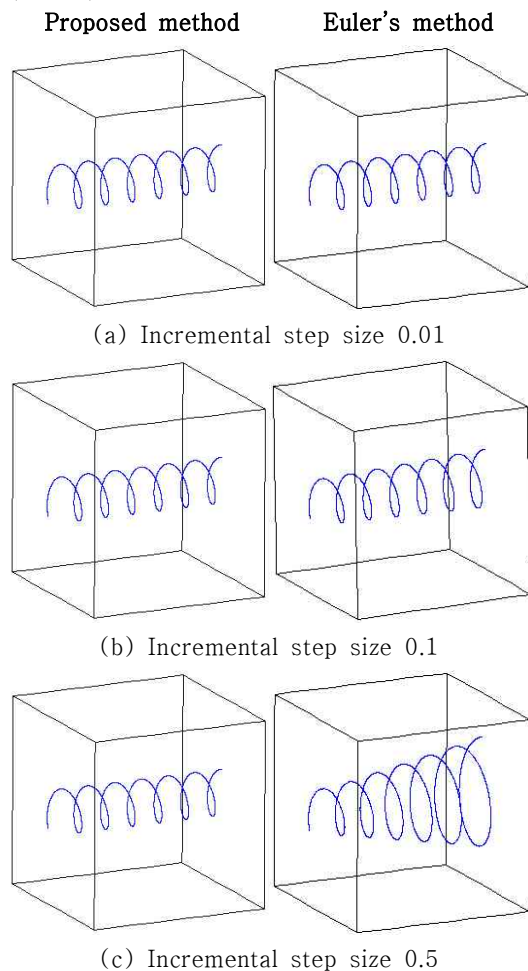


(Figure 3) Validation of proposed method with conventional methods

(그림 4)에서 보는 것 같이 증분크기를 충분

히 작게 하면 본 논문에서 제안한 방법과 기존의 수치 해석적 방법으로 구현한 결과의 차이점을 육안으로 구별할 수 없다. 그러나 본 논문에서 제안한 방법은 다양하게 증분 크기를 변경해도 육안으로 차이점을 발견할 수 없으나 기존의 방법은 증분 크기가 변하면 오류를 발생시킬 수 있다. 이 오류로 정확한 탄젠트 곡선이 아닌 다른 탄젠트 곡선을 생성한다. (그림 4)에서 이 차이점을 보여주고 있다

(그림 4) 다양한 증분 크기에서 두 방법의 비교



(Figure 4) Comparison of two methods at various incremental step sizes

5. 결론

탄젠트 곡선을 계산하기 위해 본 논문에서 제안한 방법은 탄젠트 곡선을 정의하는 상미분 방정식을 풀어 명시해(explicit solution)를 제시한 반면 기존의 방법은 수치 해석적 방법인 Runge-Kutta 방법 또는 Euler 방법[2,3,4,12]을 사용하여 탄젠트 곡선의 근사치를 구했다. 따라서 본 논문에서 제시한 방법은 기존의 방법에서 범할 수 있는 수리적 오류를 줄일 수 있고 탄젠트 곡선을 계산하는데 정확성이 보장된다. 따라서 3차원 벡터 필드를 가시화하는 효율적인 방법으로 사용될 수 있다.

향후 연구 과제로는 3차원 벡터 필드에서 탄젠트 커브를 정의하는 상미분 방정식에 대해 무게 중심 좌표계에서의 명시해(explicit solution)를 마련하는 연구가 진행되어야 할 것이다.

References

- [1] I. -H. Jung, "An Efficient Visualization Method of Two-Dimensional Vector Fields", *Journal of Korea Multimedia Society*, Vol.12, No.11, pp.1623-1628, No v. 2009.
- [2] A. Globus, C. Levit, and T. Lasinski, "A tool for Visualizing the Topology of Three-dimensional Vector Fields," *Proc. Visualization '91, IEEE Computer Society*, pp.33-40, 1991.
- [3] J. L. Helman and L. Hesselink, "Representation and Display of Vector Field Topology in Fluid Data Sets," *IEEE Computer*, Vol.22, No.8, pp.27-36, Aug. 1989.
- [4] J. L. Helman and L. Hesselink, "Surface Presentation of Two- and Three-Dimensional Fluid Flow Topology," *Proc. Visualization '90, IEEE Computer Society*, pp.6-13, 1990.
- [5] J. L. Helman and L. Hesselink, "Visualizing Vector Field Topology in Fluid Flows," *IEEE Computer Graphics and Applications*, Vol.11, pp.36-46, May 1991.
- [6] J. C. Kim, K. O. Kim, E. K. Kim and C. Y. Kim, "3D Visualization System for Realtime Environment

al Data”, Journal of Digital Contents Society, Vol.9, No.4, pp.707-715, Dec. 2008.

[7] V. I. Arnold, “Ordinary Differential Equations”, MIT Press, 1973.

[8] S. Birkhoff and G. Rota, “Ordinary Differential Equations”, fourth edition, Wiley, NewYork, 1989.

[9] D. Kenwright and D. Lane, “Optimization of Time-Dependent Particle Tracing using Tetrahedral Decomposition,” Proc. Visualization '95, IEEE Computer Society, pp.321-328, 1995.

[10] G. M. Nielson, “Visualization in Scientific and Engineering Computation,” IEEE Computer, Vol.24, No.9, pp.58-66, 1991.

[11] G.M. Nielson, I. -J. Jung, N. Srinivasan, J. Sung and J. -B. Yoon, “Tools for Computing Tangent Curves and Topological Graphs for Visualizing Piecewise Linearly Varying Vector Fields over Triangulated Domains,” in: Scientific Visualization: Overview, Methodologies and Techniques, G. Nielson, H. Hagen and H. Muller, editors, IEEE Computer Society Press, Los Alamitos, CA, pp.527-562, 1997.

[12] G. M. Nielson and A. Kaufman, “Visualization Graduates,” IEEE Computer Graphics and Applications, Vol.14, No. 5, pp.17-18, Sep. 1994.

[13] S. -K. Ueng, C. Sikorski, and K.-L. Ma, “Efficient Streamline, Streamribbon, and Streamtube Constructions on Unstructured Grids,” IEEE Transactions of Visualization and Computer Graphics, Vol.2, No.2, pp.100-108, Jun. 1996.



정 일 흥

1993년: 애리조나 주립대학 컴퓨터 공학과 졸업 (공학석사)

1998년: 애리조나 주립대학 컴퓨터 공학과 졸업 (공학박사)

1998년 ~ 현재 : 대전대학교 컴퓨터공학과 교수
 관심분야 : 컴퓨터 그래픽스(Computer Graphics), 멀티미디어(Multimedia), 디지털 콘텐츠(Digital Contents), 가상현실(Virtual Reality), 애니메이션(Animation) 등