

# 자율주행 시스템의 장애물 회피 알고리즘에 관한 연구 Researches on Collision Avoidance Algorithms for Autonomous Driving System

안두성 · 박근현 · 최규종 · 전순용

D. S. Ahn, G. H. Park, G. J. Choi, S. Y. Jeon

(접수일 : 2011년 03월 22일, 수정일 : 2011년 10월 06일, 채택확정 : 2011년 10월 12일)

**Key Words** : Irregular obstacles(불규칙 장애물), Obstacle avoidance(장애물 회피), Trajectory generation(궤적생성), Unmanned vehicles(무인차량)

**Abstract** : In unmanned vehicles' navigation, the shapes of obstacles are generally irregular and complex. The motion of vehicles based on the range sensor system such as ultrasonic sensors or laser sensors can be unstable due to the irregular shape of the obstacles. In this case, to generate stable trajectory of unmanned vehicles equipped with range sensors, we need an approach that can simplify an obstacle's irregular shape information. In this paper, we propose the trajectory generation algorithm that an vehicle can stably navigate in the environments where irregular shaped obstacles are scattered. The proposed method is verified through the analysis of vehicle's trail and direction data acquired by simulations and implementations.

## 1. 서 론

무인차량(unmanned vehicle)의 주행(navigation)<sup>1,2</sup> 시 감지되는 장애물의 형상은 일반적으로 단순한 형상 보다는 복잡하고 불규칙한 형상이 대부분이다. 이것은 장애물까지의 거리를 측정하는 초음파센서(ultrasonic sensor)나 레이저 센서(laser sensor)를 이용하여 주행하는 무인차량에게는 장애물의 불규칙한 형상으로 인하여 차량의 움직임을 불안정하게 만드는 원인이 될 수 있다.

현재 다양한 장애물 회피 알고리즘<sup>3,4,5,6</sup>들이 연구되고 있으며, GVG(Generalized Voronoi Graph), EPF(Electro Potential Field) 등을 응용하는 방식도 그러한 알고리즘들 중에 하나이다. 그러나 이것은 장애물의 형상을 직접적으로 이용하여 회피궤적을 생성하는 방식이기 때문에 장애물의 형상이 복잡할 경우 불필요한 차량의 움직임을 발생시킬 수 있다. 따라서 본 논문에서는 이러한 단점을 해결하기 위해

복잡한 장애물의 형상을 단순화 시켜 장애물들을 안정적으로 회피할 수 있는 차량의 이동방향을 결정하는 알고리즘을 제시하고자 한다.

먼저 장애물의 불규칙한 형상정보를 단순화하여 처리하기 위해 레이저 센서로 획득한 거리값들을 이용하여 차량이 이동할 수 있는 영역을 타원형 형상으로 생성한다. 그리고 생성된 타원형의 장축(major axis)을 이용하여 차량의 속도(velocity)를 조절하고, 단축(minor axis)을 이용하여 최소 안전영역을 설정한다. 제시한 알고리즘의 타당성은 시뮬레이션과 실험을 통해 생성된 궤적들을 분석하여 검증한다.

논문의 구성은 2장에서 본 연구에서 제시한 장애물 회피 알고리즘과 궤적생성 알고리즘을 설명하고, 3장에서는 실험에 사용된 모바일 플랫폼과 전체 시스템 구성도를 설명한다. 4장에서는 본 논문에서 제시한 알고리즘을 시뮬레이션과 실험을 통해 검증하였으며, 5장에서 논문의 결론을 도출하였다.

## 2. 장애물 회피 알고리즘

무인차량이 다양한 형상의 장애물을 안정적(stable)으로 회피한다는 것은 상당히 어려운 문제 중에 하나이다. 인간과 같은 지능을 가지지 못한 차

안두성(교신저자) : 부경대학교 기계자동차공학과  
E-mail : dsahn@pknu.ac.kr, Tel : 051-629-6154  
박근현 : 부경대학교 대학원  
최규종 : (주)에스피시스템스  
전순용 : (주)대원전자 기술연구소

량은 다양한 센서<sup>7</sup>를 통해 획득된 장애물의 정보를 분석하여 이동방향을 매 순간마다 결정해야 한다. 일반적으로 차량이 주행 시 회피해야 할 장애물의 형상은 다양하고 불규칙(irregular)한 것이 대부분이다. 따라서 차량의 이동방향을 결정하기 위해 통상적인 거리센서를 통해 획득된 장애물의 형상 정보를 직접적으로 사용하는 방법은 그 움직임이 다소 불안정할 확률이 높을 것이다. 이것은 장애물의 형상이 불규칙할수록 그리고 차량 주위에 가까이 위치할수록 더 심하게 될 것이다.

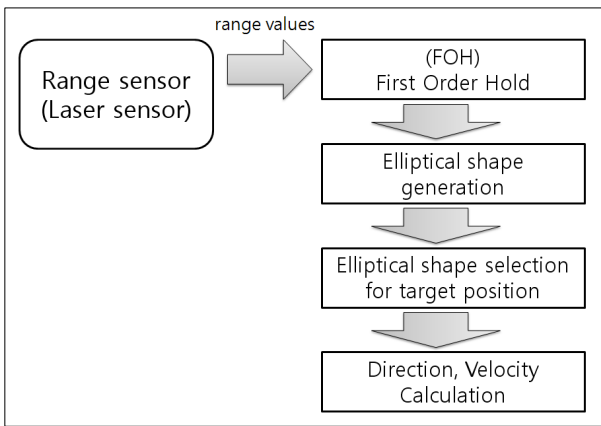


Fig. 1 Collision avoidance algorithm

본 논문에서는 0도에서 180도로 스캔이 가능한 레이저센서를 통해 획득한 장애물 형상정보(각도와 거리값의 쌍으로 이루어지는 측정데이터를 이용하여 계산)와 차량의 현재 위치정보(localization)<sup>8</sup>를 이용하여 차량이 이동 가능한 영역을 타원형(ellipse shape)으로 생성한다. 생성된 영역에서 차량의 움직임은 장축과 단축을 이용하여 제어한다. 즉, 장축방향을 차량의 이동방향으로 결정하고 그것의 길이를 이용하여 차량의 속도를 조절한다. 식(1)은 차량의 속도  $v_r$ 을 계산하는 알고리즘<sup>9</sup>을 보여준다. 첫 번째 항은 차량의 최대속도  $v_{max}$ 에 타원형 장축의 크기에 비례하는 비례계수  $k$ 를 곱하여 차량의 선속도를 계산하는 항이다. 두 번째 항은 식(2)에서 계산된 차량의 각속도  $w_r$ 의 크기에 따라 첫 번째 항의 차량의 선속도를 조절해주는 항이다.

$$v_r = kv_{max} \pm \frac{v_{max}}{w_{max}} w_r \quad \begin{cases} + : w_r \leq 0 \\ - : w_r > 0 \end{cases} \quad (1)$$

$$w_r = k_1 d_e + k_2 r_e + k_3 v_{obs} \quad (2)$$

식(2)에서  $k_1, k_2, k_3$ 는 계인값을 나타내고,

$d_e, r_e, v_{obs}$ 는 차량의 목표방향과 진행방향의 오차값, POIs (Points Of Interest)들의 오차값, 차량을 기준으로 장애물의 상대속도를 각각 나타낸다<sup>9</sup>.

이와 같은 방법은 장애물의 복잡한 형상정보를 직접적으로 이용하지 않기 때문에 차량의 움직임에 대한 불규칙한 장애물 형상의 영향을 줄일 수 있다. Fig. 1은 본 논문에서 제시하는 개략적인 알고리즘의 흐름을 나타낸다.

### 2.1 타원형 형상의 영역 설정

불규칙한 형상을 가진 장애물들이 복잡하게 흩어져(scattered) 있는 환경에서 무인차량의 이동방향을 생성하기 위해 장애물이 존재하지 않는 영역을 Fig. 2와 같이 타원형 형상을 이용하여 설정하는 방법을 사용한다. 먼저 레이저센서를 이용하여 획득한 거리값들을 1차 보간(First Order Hold)을 수행한다. 그리고 타원형의 단축(minor axis)을 일정하게 고정된 상태에서 사전에 설정한 샘플링 각(sampling angle)만큼 이동하면서 차량이 이동 가능한 영역을 나타내는 타원형을 생성한다. 여기서 타원형을 생성하기 위해 타원형의 중심은 현재 차량의 위치값을 사용하고, 단축길이를 고정시키고 장축 방향의 길이를 장애물이 존재하지 않는 범위 내에서 최대가 되도록 생성한다. 다음으로 생성된 타원형들 중에서 이동하고자 하는 특정 방향을 결정한다.

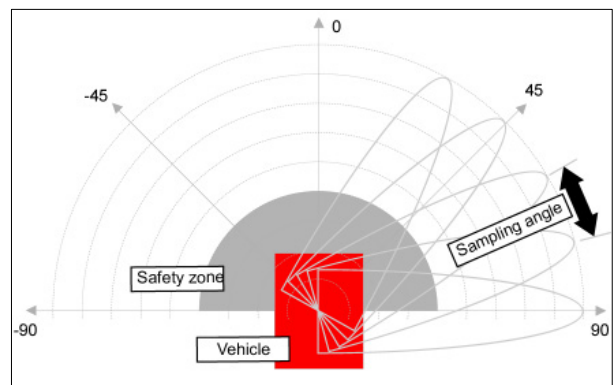


Fig. 2 Elliptical shape area generation method

### 2.2 궤적생성 알고리즘

무인차량의 이동방향 및 속도는 생성된 타원형에서 그것의 장축(major axis)의 정보를 이용해 차량의 이동방향(direction)과 속도(velocity)<sup>10</sup>를 계산할 수 있다. 이것은 장애물의 복잡한 형상정보를 간접적으로 이용하기 때문에 불필요한 차량의 움직임을 감소시킬 수 있다. 또한 타원형의 장축 정보만을 이용하

여 무인차량의 움직임을 간단히 제어할 수 있다는 장점을 가진다. 다음은 차량의 움직임을 제어하는 제어입력(control input)을 계산하는 알고리즘을 설명한다.

**step 1. 기본 설정값 결정**

타원형의 단축(350mm) 및 샘플링 각도(20degree)

**step 2. 레이저센서를 이용한 거리값 획득**

Fig. 3은 각도에 따라 획득한 180개의 거리값을 보여준다.

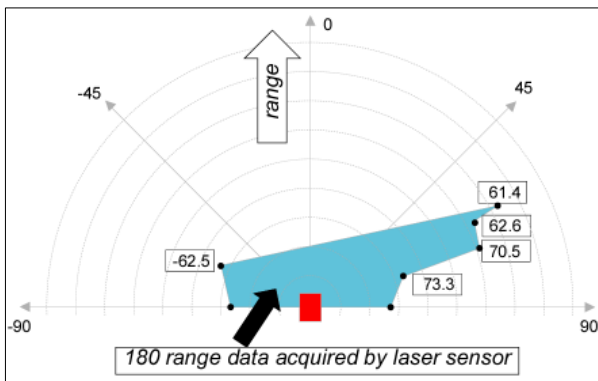


Fig. 3 Range data acquired by laser sensor

**step 3. 거리값 정보들의 1차 보간**

**step 4. 각 방향으로 타원형 생성**

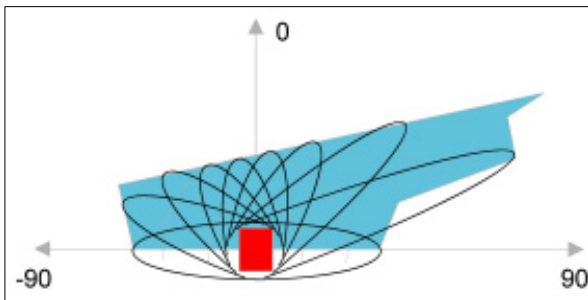


Fig. 4 Generation of elliptical shape

타원형 생성 시 타원형의 단축과 샘플링 각도는 step 1에서 사전에 결정하였다. 따라서 장축의 길이를 결정하면 타원형을 생성할 수 있다. 타원형의 장축방향은 샘플링각도에 의해 결정되기 때문에 그 방향으로 획득된 거리값을 이용하여 장축의 길이를 결정한다. Fig. 4는 샘플링 각도에 따라 생성된 타원형들을 보여준다.

**step 5. 장축길이 계산**

장축방향의 센서값과 이웃해 있는 센서값들이 생성된 타원형의 내부에 존재하지 않는 가장 큰 장축

을 찾는다. 실제 실험에 사용한 이웃한 센서값들은 장축방향을 기준으로 20개의 데이터를 이용하였다.

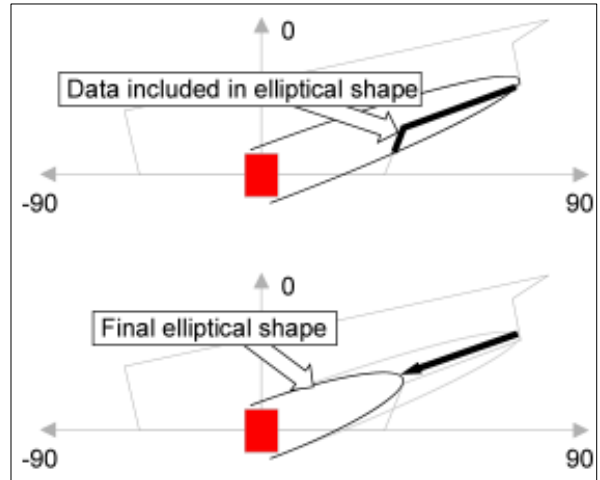


Fig. 5 Scaling process of elliptical shape

Fig. 5는 생성된 타원형의 내부에 존재하는 장애물들을 타원형 내부에서 제외시키기 위해 타원형의 장축의 길이를 스케일링하는 과정을 보여준다.

**step 6. 목표방향의 타원형 선택**

주행목표위치가 정해져 있지 않는 경우 step 5까지의 과정을 거치면서 생성된 타원형들 중에 장축의 길이가 가장 긴 장축의 방향이 장애물을 안정적으로 회피할 수 있는 방향으로 선택할 수 있다.

**step 6. 장축을 이용한 이동 방향/속도 계산**

선택된 타원형의 장축방향이 차량의 이동방향을 나타내며 각속도는 차량의 현재 진행방향과 주행방향의 차이값에 따라 비례식을 사용하고, 속도는 장축이 짧을수록 장애물이 차량 가까이에 존재한다는 것을 나타내므로 역시 장축의 길이에 따라 비례식으로 속도를 제어한다.

**3. 시스템 구성**

본 실험에서 사용된 차량의 이동 플랫폼은 Fig. 6에 보이는 ActivMedia Robotics의 P3DX를 기반으로 하며, 장착되어 있는 센서는 초음파 센서와 SICK사의 LMS200 레이저센서이다. 차량의 내부 상태를 모니터링(Monitoring)하기 위한 GUI (Graphic User Interface)는 Fig. 7에 보이는 것과 같이 랩뷰(LabView)를 사용하여 코딩하였다. GUI는 레이저센서를 통해 수집되는 거리값과 차량의 이동 궤적 및 속도를 실시간 모니터링을 수행할 수 있다.

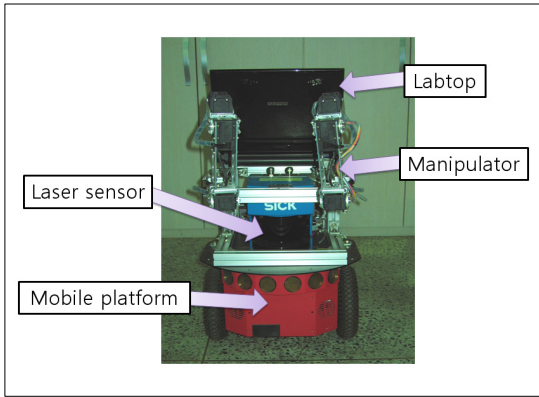


Fig. 6 Mobile platform(P3DX platform)

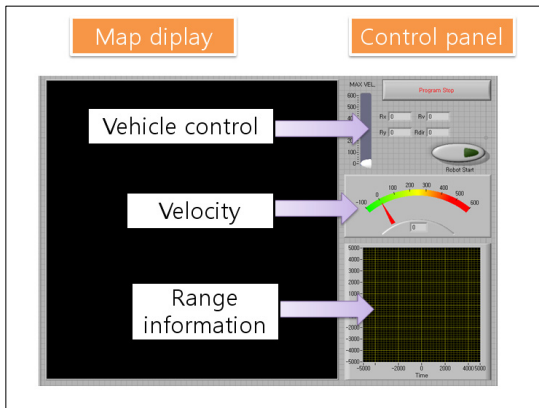


Fig. 7 Monitoring GUI for mobile robot system

전체 시스템은 Fig. 8와 같이 크게 하드웨어(H/W) 부분과 소프트웨어(S/W)로 구분하였으며, 하드웨어는 P3DX 이동 플랫폼을 기반으로 8개의 초음파 센서, 레이저센서, 다이나믹셀(Dynamixel)을 이용해 제작된 2개의 매니퓰레이터(manipulator)와 랩탑컴퓨터(laptop computer)로 이루어져 있다. 실험에서는 장애물의 정보를 정확히 획득하기 위해 레이저센서만을 사용하였다. 소프트웨어(S/W)는 랩뷰(LabView)를 이용하여 GUI를 설계 및 제시된 알고리즘과 차량의 구동 부분을 코딩하였다.

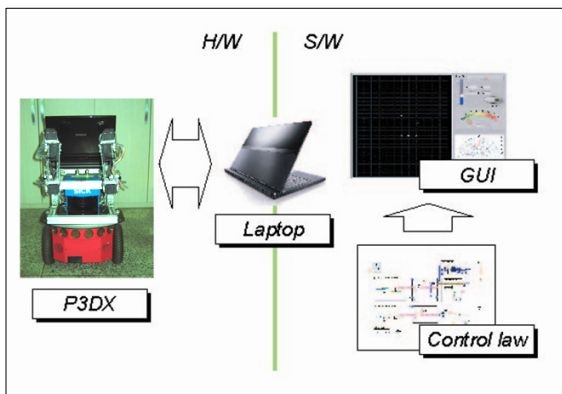


Fig. 8 Overall system architecture

#### 4. 시뮬레이션과 실험

시뮬레이션<sup>11,12</sup>의 초점은 장애물을 통과하는 차량의 궤적이 얼마나 부드럽고(smooth) 안정적(stable)인지를 평가하는데 있다. 또한 특정 목표위치로 무인 차량을 이동시키는 것이 아니라 현재 위치에서 장애물을 회피하기 위해 가장 안전한 방향으로 이동하도록 하였다. 시뮬레이션에 사용된 차량의 구조는 실제 실험장치와 동일하게 구성하였으며, 레이저센서를 이용하여 장애물까지의 거리값을 측정한다.

본 논문에서 제시한 알고리즘을 시뮬레이션과 실험을 통해 상호 비교하여 검증하기 위해 두 환경을 최대한 동일하게 구성하였다. 즉 시뮬레이션에서 설정한 장애물들의 위치와 모양들을 실제 환경에서 구현하기 위해 벽을 인위적으로 설정하여 구현하였다. 그러므로 각각의 결과들 상호 비교함으로써 차량 궤적의 안정성에 대한 분석과 실제 시스템으로의 적용 가능성을 분석할 수 있다.

주위환경에서 일상적으로 볼 수 있는 장애물은 대부분 불규칙한 형상을 가지며, 복잡하게 흩어져 있다. 따라서 시뮬레이션과 실험에서 이와 같은 조건들을 반영하여 불규칙한 형태의 장애물들을 가정하였다. 실험결과로서 랩뷰(LabView)로 코딩한 GUI에 수집된 차량의 궤적을 분석하여 검증하였다. GUI에는 차량의 궤적과 함께 레이저센서로 획득한 거리값들을 이용하여 장애물의 위치정보도 함께 나타나도록 구성되어 있다.

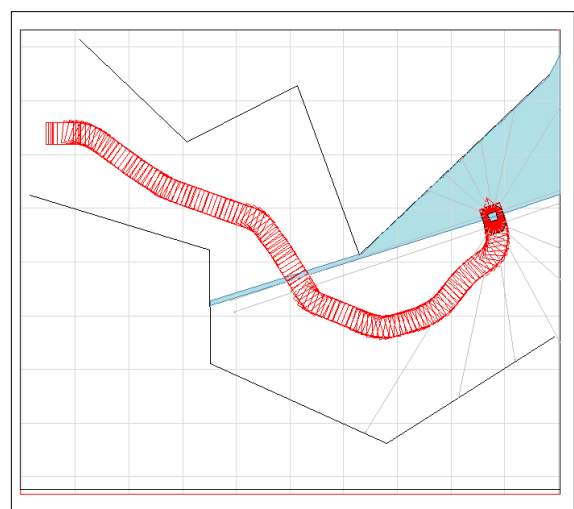


Fig. 9 Simple shaped walls (simulation)

Fig. 9에서는 비교적 단순한 형태의 벽이 존재하는 통로를 지나가는 환경을 가정하여 시뮬레이션을 수

행하였다. 생성된 차량의 궤적을 분석해 볼 때 전반적으로 부드러운 궤적을 나타내고 있다는 것을 알 수 있다. Fig. 10은 시뮬레이션 환경을 실제 환경에서 그대로 설정하여 실제 이동 플랫폼을 가지고 실험한 결과를 GUI창에 모니터링한 결과를 표시하였다. 비교적 선이 두꺼워 보이는 부분이 레이저센서를 이용하여 감지된 장애물이며, 그 사이로 얇은 선이 차량의 이동 궤적을 나타낸다. 시뮬레이션 결과와 유사하게 차량의 움직임이 안정적으로 나타남을 알 수 있다.

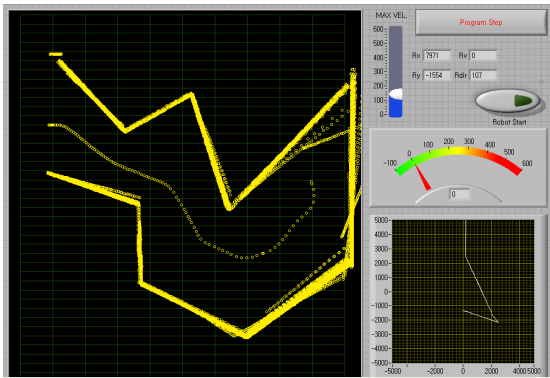


Fig. 10 Simple shaped walls (implementation)

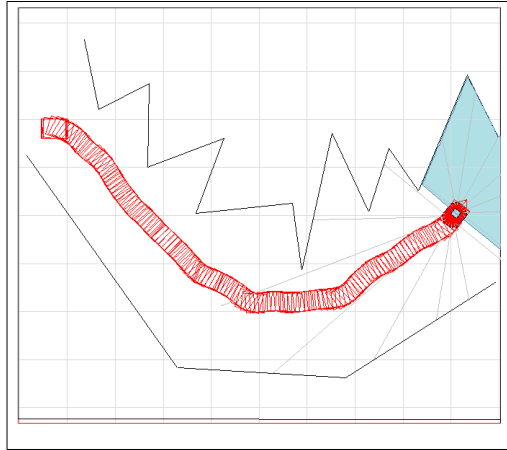


Fig. 11 Complex shaped walls (simulation)

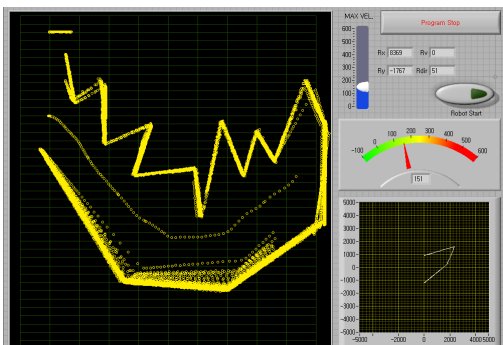


Fig. 12 Complex shaped walls (implementation)

Fig. 11에서는 Fig. 9보다 벽의 형상을 좀 더 불규칙하게 설정하여 제시한 알고리즘으로 차량의 궤적을 생성하여 그 결과를 분석하였다. 차량이 불규칙한 형상이 존재하는 구간을 통과할 때 생성된 궤적은 불규칙한 형상에 대한 영향을 덜 받으면서 부드러운 움직임을 보인다는 것을 알 수 있다. 또한 Fig. 12에서와 같이 실험결과에서도 차량은 레이저 센서를 통해 장애물을 인식하고 인식된 장애물 정보를 이용하여 궤적을 시뮬레이션 결과와 유사하게 생성하고 있음을 알 수 있다.

마지막으로 Fig. 13은 삼각형 형상의 장애물들이 복잡하게 흩어져(scattered) 있는 환경을 가정하여 시뮬레이션을 수행 하였다. 차량의 궤적이 불안정한 움직임을 보이는 구간들이 부분적으로 존재하지만 대체적으로 불규칙한 장애물의 형상에 크게 영향을 받지 않았다는 것을 알 수 있다. Fig. 14의 실험결과도 시뮬레이션 결과와 유사하게 나타남을 알 수 있다.

본 논문에서 제시한 알고리즘을 검증하기 위해 세 가지 시뮬레이션과 실험을 수행하였다. 그 결과 불규칙한 장애물의 형상에 따라 무인차량의 움직임이 다소 불안정해 보이는 구간이 부분적으로 존재한다는 것을 알 수 있었다. 그러나 적용된 알고리즘의 단순성을 감안하고, 실제 시스템에 적용하여 실험한 결과도 시뮬레이션과 유사하다는 것을 볼 때 상당히 만족스러운 결과라고 할 수 있다. 본 연구에서 타원형 형상을 생성하기 위해 단축을 고정시키고 샘플링 각도를 일정하게 유지하여 궤적을 생성하였지만, 향후 이러한 부분들에 대해 알고리즘을 개선하기 위한 연구가 보완되어야 할 것이다.

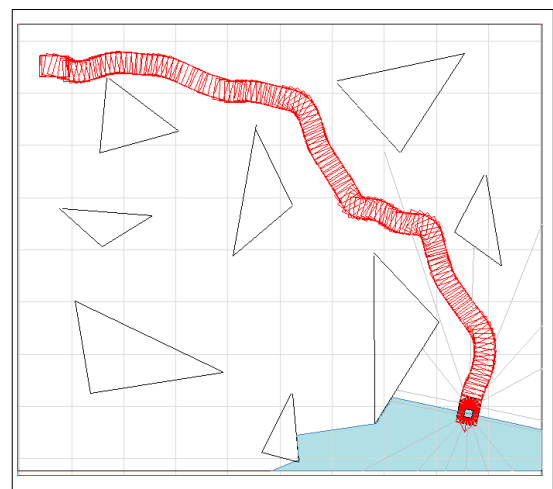


Fig. 13 Scattering triangular obstacles (simulation)

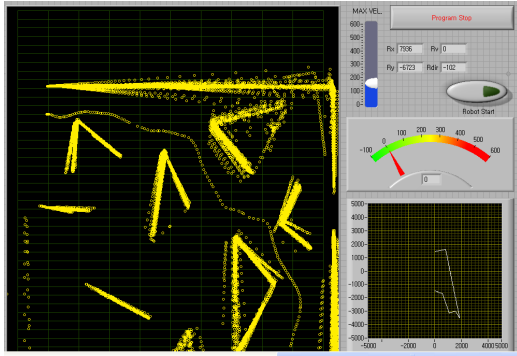


Fig. 14 Scattering triangular obstacles (implementation)

## 5. 결 론

본 논문에서는 불규칙한 형상의 장애물들이 존재하는 환경 내에서 무인차량이 장애물을 안정적으로 회피할 수 있는 궤적을 생성하는 알고리즘을 제시하였다. 제시된 알고리즘에 대한 시뮬레이션 및 실험을 통해 불규칙한 형상의 장애물들이 흩어져있는 환경 조건에서 차량이 얼마나 부드럽고(smooth) 안정적(stable)으로 회피동작을 수행하는지를 확인하였다.

본 연구는 정성적인(qualitative) 방법으로 복잡한 형상의 장애물들을 회피하고, 실제 시스템에 적용할 수 있는 알고리즘을 구현하는 것에 초점을 두었다. 비록 생성된 궤적이 부분적으로 불안정해 보이는 구간들이 존재하여 향후 지속적인 보완연구들이 필요하지만, 알고리즘 구현의 단순성을 볼 때 실제 시스템에 적용 가능성이 높다고 평가할 수 있다.

## 후 기

본 과제(결과물)는 교육과학기술부의 재원으로 한국연구재단의 지원을 받아 수행된 광역경제권 선도산업 인재양성사업의 연구결과입니다.

## 참고 문헌

1. S. H. Park and B. H. Lee, "Practical Environment Modeling Based on a Heuristic Sensor Fusion Method", Proceedings of the 2004 IEEE International Conference on Robotics & Automation, pp. 200-205, April, 2004.
2. Alessandro Correa Victorino, Patrick Rives and Jean-Jacques Borrelly, "Mobile Robot Navigation Using a Sensor-Based Control Strategy", Proceedings of the 2001 IEEE International Conference on Robotics and Automation, Seoul, Korea, vol. 3, pp. 2753-2758, May, 2001.
3. Choset, H., and Burdick, J. W., "Sensor based planning, Part I: The Generalized Voronoi Graph", Proc. IEEE Int. Conf. on Robotics and Automation, Nagoya, Japan, pp.1649-1655, 1995.
4. Roque, W. L. and Doering, D., "Multiple Visiting Stations Trajectory (Re)Planning for Mobile Lab Robots Based on Global Vision and Voronoi Roadmaps", in Proc. of the 33rd International Symposium on Robotics, Stockholm, Sweden, 2002.
5. Nikos C. Tsourveloudis, Kimon P. Valavanis, and Timothy Hebert, "Autonomous Vehicle Navigation Utilizing Electrostatic Potential Fields and Fuzzy Logic", IEEE Trans. Robot. Automat., vol. 17, pp.490-497, 2001.
6. J. Modayil, P. Beeson, and B. Kuipers, "Using the topological skeleton for scalable global metrical map-building", in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Sendai, Japan, pp. 1530-1536, 2004.
7. Zou Yi, Ho Yeong Khing, Chua Chin Seng, and Zhou Xiao Wei, "Multi-ultrasonic Sonic Fusion for Mobile Robots", Proceedings of the IEEE Intelligent Vehicles Symposium 2000, pp. 387-391, October 3-5, 2000.
8. Jennings, C., Murray, D., Little, J.J., "Cooperative robot localization with vision-based mapping", Robotics and Automation, Vol. 4, pp. 2659-2665, 10-15 May 1999.
9. Gyu-Jong Choi, Young-Seok Jung, and Doo-Sung Ahn, "Mobile robot navigation algorithm using a vector-based topological map and virtual Jacobian", Proc. IMechE Vol.221 Part I: J. Systems and Controls Engineering, 2007.
10. K. R. S. Kodagoda, W. S. Wijesoma and E. K. Teoh, "Fuzzy Speed and Steering Control of an

AGV”, IEEE Transaction on Control Systems  
Technology, vol. 10, pp. 112-120, Jan. 2002.

11. Laurene V. Fausett, “Applied numerical  
analysis using Matlab”, Prentice hall, 1999.
12. J.-S. R. Jang, C.-T. Sun and E. Mizutani,  
Neuro-Fuzzy AND Soft-Computing, Prentice  
hall., 1997.