

# 타이어 슬립과 조향작동장치의 성능을 고려한 무인자동차 자율주행 제어

## Autonomous Vehicle Driving Control Considering Tire Slip and Steering Actuator Performance

박찬호<sup>1</sup> · 광기성<sup>1</sup> · 정호운<sup>1</sup> · 홍도의<sup>1</sup> · 황성호<sup>1\*</sup>

C. H. Park, G. S. Gwak, H. U. Jeong, D. U. Hong and S. H. Hwang

Received: 30 Jan. 2015, Revised: 12 Jun. 2015, Accepted: 28 Aug. 2015

**Key Words** : Autonomous vehicle(무인자동차), Tire slip angle(타이어 미끄럼각), Side slip angle(차체 미끄럼각), Steering Actuator(조향작동장치), Sensor Noise(센서노이즈)

**Abstract:** An autonomous vehicle control algorithm based on Ackerman Geometry is known to be reliable in low tire slip situation. However, vehicles at high speed make lateral errors due to high tire slip. In this paper, considering the tire slip of vehicles, the steering angle is determined based on the Ackerman Geometry and is supplemented tire slip angle by the Stanley steering algorithm. In addition, to prevent the tire slip, the algorithm, which restricts steering if a certain level of slip occurs, is used to reduce the lateral error. While many studies have been extended to include vehicle slip, studies also need to be carried out on the tire slip depending on hardware performance. The control algorithm of autonomous vehicles is compensated considering the sensor noise and the performance of steering actuator. Through the various simulations, it was found that the performance of steering actuator was the key factor affecting the performance of autonomous driving. Also, it was verified that the usefulness of steering algorithm considering the tire slip and performance of steering actuator.

### 기호 설명

- |   |  |
|---|--|
| <p><math>a</math> : 가속도 (<math>m/s^2</math>)</p> <p><math>a_y</math> : 횡 방향 가속도 (<math>m/s^2</math>)</p> <p><math>b</math> : 앞 차축에서 중심점까지의 거리 (<math>m</math>)</p> <p><math>c</math> : 뒷 차축에서 중심점까지의 거리 (<math>m</math>)</p> <p><math>e_\theta</math> : 차량기준에서 도로의 각도오차 (<math>rad</math>)</p> <p><math>e_y</math> : 목표경로에서 중심점까지의 거리 (<math>m</math>)</p> <p><math>e_m</math> : 예측된 횡 방향 오차 (<math>m</math>)</p> <p><math>v</math> : 주행속도 (<math>m/s</math>)</p> | <p><math>v_x</math> : 종 방향 속도 (<math>m/s</math>)</p> <p><math>v_y</math> : 횡 방향 속도 (<math>m/s</math>)</p> <p><math>L</math> : 차축 사이의 거리 (<math>m</math>)</p> <p><math>R</math> : 회전반경 (<math>m</math>)</p> <p><math>t_{str}</math> : 중립조향으로 복귀하는데 걸리는 시간 (<math>s</math>)</p> <p><math>\alpha_f</math> : 전륜 타이어 미끄럼각 (<math>rad</math>)</p> <p><math>\alpha_r</math> : 후륜 타이어 미끄럼각 (<math>rad</math>)</p> <p><math>\beta</math> : 차체 미끄럼각 (<math>rad</math>)</p> <p><math>\delta</math> : 전륜 조향각 (<math>rad</math>)</p> <p><math>\dot{\psi}</math> : 요레이트 (<math>rad/s</math>)</p> <p><math>act_{max}</math> : 스티어링 구동장치의 최대속도 (<math>rad/s</math>)</p> <p><math>e_{max}</math> : 알고리즘에 반영될 수 있는 최대 횡 방향 오차 (<math>m</math>)</p> |
|---|--|

\* Corresponding author: hsh@me.skku.ac.kr  
 1 Department of Mechanical Engineering, Sungkyunkwan University, Suwon 16419, Korea  
 Copyright © 2015, KSFC

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

### 1. 서론

무인자동차의 제어 기법으로는 DARPA Grand

Challenge의 우승팀인 Stanford University가 제안한 Stanley 조향제어 방식이 많이 사용된다. 해당 방법은 기본적으로 경로와 차량 간의 방향 오차를 보정하며 추가적으로 횡 방향 오차를 일정량 보상하는 방법으로 구성되어 있다.<sup>1)</sup>

하지만 Ackerman Geometry를 기반으로 구성된 Stanley 조향제어는 Ackerman Geometry 자체가 타이어 슬립이 고려되지 않는 저속 영역을 가정함으로 고속 영역이나 혹은 타이어와 도로 간의 마찰계수가 낮은 경우에는 그 신뢰성을 보장할 수 없다.<sup>2)</sup>

이러한 환경에서는 차량의 특성에 따라 부족 조향이나 과도 조향이 나타나게 된다. 무인자동차의 자율항법 알고리즘에서 이러한 차량 특성은 고속 선회시에 오차로 발생하게 되며 이러한 문제를 방지하기 위하여 진행된 많은 연구들은 복잡한 알고리즘이 가지는 변수 튜닝의 어려움과 범용적인 적용에 어려움을 겪고 있다.<sup>3)</sup>

더욱이 무인자동차의 연구방향이 선행연구에서 상용화의 단계로 진행되고 있으며 이로 인하여 고정밀 센서를 이용한 무인자동차의 연구들이 상용화된 센서들을 위주로 재구성되고 있으며 상대적으로 저렴한 센서들이 가지는 노이즈를 효과적으로 컨트롤하기 위한 연구들이 많이 진행되고 있다.<sup>4)5)</sup>

본 연구에서는 슬립이 고려되지 않은 자전거 모델을 기반으로 타이어의 슬립 영역까지 확대할 수 있는 수식을 도출하고 도출된 수식을 기반으로 새로운 조향제어 알고리즘을 개발하였다. 조향제어 알고리즘이 센서의 노이즈나 무인자동차를 구동하기 위한 조향 작동장치의 최대 회전 성능에 대하여 어떻게 반응하는지를 분석하였으며, 제시된 알고리즘이 상용화 단계에서 충분히 강건한 제어를 제공할 수 있는지를 검토하였다.

## 2. 조향 기하학

타이어가 미끄럼각이 없이 구르는 저속 상황을 해석 및 수식도출의 편리함을 위해서 자전거 모델로 표현하면 Fig. 1과 같이 선회 중심점이 후륜 구동축의 연장선상에 존재하게 된다.

하지만 미끄럼각이 발생하는 고속 영역에서는 Fig. 2와 같이 선회 중심점이 후륜구동축의 연장선상에 있지 않는다. Fig. 2에서 보이는  $\alpha_f, \alpha_r, \beta$ 는 고속 선회에서 생기는 각 들이다. 실제 전륜은 전륜 타이어 미끄럼각의 영향으로 계산보다 부족하게 이동하게

될 것이다. 차체 미끄럼각은 실제 차량중심이 이동하는 방향을 나타내는데 횡 방향 오차를 보정하기 위한 식을 통해 도출된 값과 다르기에 여기서 또 오차가 발생하게 된다. 전륜 미끄럼각과 차량 미끄럼각을 차량에 기 장착된 센서들을 이용하여 구하고 시뮬레이션 상에서 이를 적용시켜 미끄럼각을 고려하는 것의 타당성을 확인할 필요가 있다.

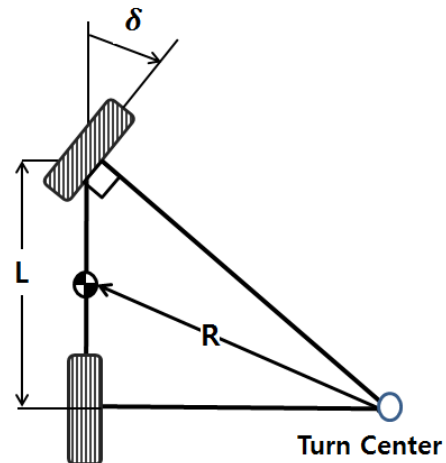


Fig. 1 Bicycle model at low speed

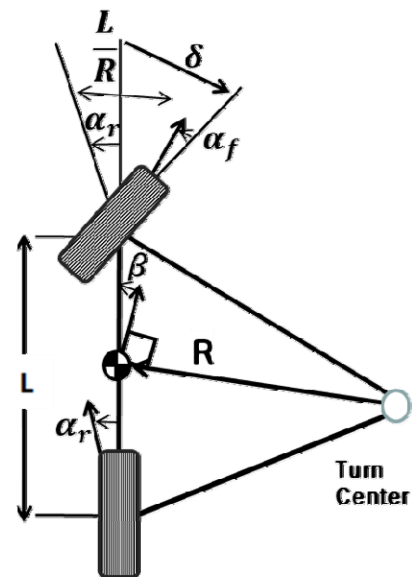


Fig. 2 Vehicle slip at high speed

## 3. 타이어 슬립을 고려한 제어 알고리즘

우선 차량시스템의 좌표와 중요 오차는 Fig. 3과 같다. 횡 방향 오차인  $e_y$ 는 차량의 무게중심으로부터 목표경로까지의 오차를 나타내며 방향오차인  $e_\theta$ 는 목표 경로의 방향과 현재 차량의 방향사이의 오차를

나타낸다. Stanley 조향제어의 기본적인 수식은 식 (1)과 같다. 차량과 목표 경로가 같은 방향을 바라보도록 하는 것이 지배적인 컨트롤이며 추가적으로 횡 방향 오차를 보상하는 형태로 구성되어 있다. 횡 방향 오차를 보상하는 식의  $k$ 는 실험적으로 도출되는 횡방향 오차의 보정이득 값이다.

$$\delta(t) = e_{\theta}(t) + \tan^{-1}\left(\frac{k \cdot e_y}{v_x(t)}\right) \quad (1)$$

현재 연구실에서 운행 중인 무인차에도 동일한 제어 알고리즘이 들어가 있으며 실험적으로 도출한  $k$  값은 0.5로 적용되어 있다.  $k$ 값이 커지면 횡 방향 오차에 대한 수정값이 커짐으로 오차가 줄어들게 되지만 직진주행에서는 차량이 사인파를 그리며 움직이게 되는 원인이 된다. 그래서 평소에는 낮은  $k$ 값을 유지하다가 선회를 할 때  $k$ 값이 증가하는 가변적인 이득값에 대한 알고리즘이 필요하다고 판단되었다. 본 논문에서는 차량의 선회시 발생하는 전륜의 타이어의 미끄럼각에 따라 횡 방향 이득값이 가변적으로 변하는 제어식을 도출하였다.

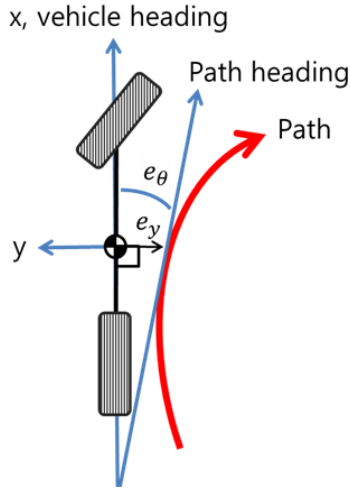


Fig. 3 Vehicle axis and errors

전륜 타이어의 미끄럼 각을 계산하기 위하여 차체 미끄럼각은 다음과 같이 도출할 수 있다.

$$v_x = v \cos \beta \cong v \quad (2)$$

$$v_y = v \sin \beta \cong v\beta \quad (3)$$

$$\beta \cong \frac{v_y}{v} \quad (4)$$

식 (2)와(3)에서 도출된 식 (4)에서 차체 미끄럼각을 계산하기 위해서는 차량 횡 방향에 대한 속도가 필요하다. 일반적으로 Electronic Stability Program (ESP)이 장착된 차량에서는 운전자의 의도를 파악하기 위한 값 중 하나로 횡 방향 가속도에 대한 값을 측정하고 있다. 시뮬레이션에서도 차체 미끄럼각을 계산하기 위해서 노이즈를 포함한 횡 방향 가속도 값을 적분하여 사용하기로 하였다. 누적되는 오차를 방지하기 위해서 중립 조향에 있을 때에는 누적 값을 0으로 초기화하였다.

차체 미끄럼각을 가지고 전륜 타이어의 미끄럼각 역시 구할 수 있으며 상대적으로 회전반경이 차축사이의 거리에 비해 길기 때문에 아래와 같이 단순화시킬 수 있다.

$$\beta = \frac{c}{R} - \alpha_r \cong \frac{L}{R} - \alpha_r \quad (5)$$

식 (5)와 Fig. 2의 기하학적 관계로부터 전륜 타이어의 미끄럼각은 다음과 같이 표현할 수 있다.

$$\alpha_f = \delta_f - \beta - \frac{\dot{\psi}b}{v} \quad (6)$$

식 (6)으로부터 전륜 타이어의 미끄럼각을 계산하여 이를 조향 제어에 적용하고자 한다. Stanley 조향 제어는 고속 선회에서 슬립으로 인하여 목표 경로를 부족하게 추종하는 경향이 있다. 필요한 양보다 횡 방향 오차가 작게 보정되는 이유는 미끄럼각 만큼 적게 보정되기 때문이다. 그래서 보상 값을 계산할 때 목표한 조향 각이 나오기 위하여 계산된 조향 각에서 조향 각과 실제 전륜 코스 각의 비를 곱하여 슬립상황에서 횡 방향 오차에 대하여 더 큰 값이 수정되도록 하여 식 (7)처럼 횡 방향 보정 이득 값이 실험적으로 도출된 상수값  $k$ 가 아닌 가변적으로 적용되는 값  $k(t)$ 를 제안하였다.

$$k(t) = \frac{\delta(t-1)}{2(\delta(t-1) - \alpha_f(t-1))} \quad (7)$$

횡 방향 보정 이득값이 가변적으로 적용되는 제어의 타당성을 확인하기 위해서 Carsim에서 제공하는 차량모델과 기본 제공되는 도로모델 중 다양한 곡률이 연속되어 있는 도로모델에서 시뮬레이션을 진행하였다.

시뮬레이션에서 시속 50km/h의 일정한 속도로 도로모델의 곡률 0.0164, 0.0045, 0.013, 0.03, 0.0182, 0.0165의 도로를 주행하도록 하였다. 목표 경로는 도로의 정 중앙을 목표경로로 설정하였다. 제어 알고리즘은 MATLAB/Simulink로 작성되었고 CarSim과 연동하여 시뮬레이션이 진행되었다. 전륜 타이어 미끄럼각을 적용하지 않은 알고리즘에서는 k값을 현재 연구실 실제 무인차량에 적용된 이득 값 0.5로 두었고 전륜 타이어 미끄럼각을 고려한 알고리즘에서는 이득값을 식 (7)로 대체하였다. 다만, 식 (7)에서는 미끄럼 각이 없을 경우에는 기존의 무인자동차와의 비교를 위해서 0.5가 나올 수 있도록 분모에 2를 곱해두었다. 하지만 실험적으로 얻어진 값이 아닌 일반적인 이득 값 1인 경우와 비교를 하는 것이 더 옳다고 판단되어 최소 이득값이 1이 나오도록 식 (7)을 식(8)과 같이 수정하였다.

$$k(t) = \frac{1}{2} \max\left(2, \frac{\delta(t-1)}{(\delta(t-1) - \alpha_f(t-1))}\right) \quad (8)$$

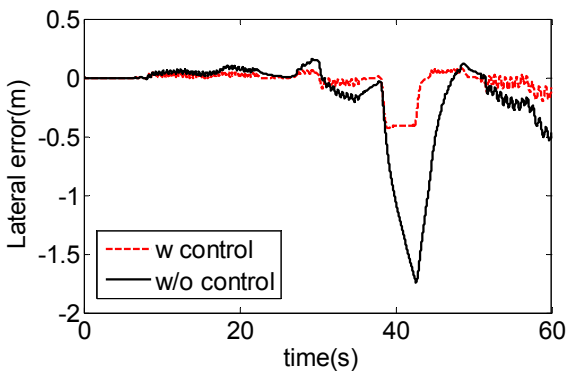


Fig. 4 Control of vehicle considering the tire slip (k=0.5)

주행결과 Fig. 4와 같이 목표 경로로부터 횡 방향 오차가 최대 75.23% 개선 효과가 있는 것을 확인할 수 있다.

이번에는 동일한 조건을 이득 값만 기존의 Stanley 대학에서와 동일하게 1로 두고 시뮬레이션 하였다. 미끄럼각을 고려한 경우는 동일하게 최소 이득 값 이 1이 나오도록 식 (8)을 적용하였다.

이득보정을 한 후의 비교결과는 Fig. 5와 같다. 최대 58%의 오차가 개선되었다. 오차 개선 효과는 이득 값을 높일수록 줄어드는 것을 확인할 수 있었다.

Fig. 4의 결과는 단순히 k값이 상대적으로 높아서 오차가 적은 것 일 수 있지만 Fig. 5는 낮은 곡률의 도로를 선회할 때 예는 거의 동일한 오차율을 보이

며 전륜 미끄럼각이 많은 급격한 곡률의 선회에서는 본 알고리즘이 타당한 효과가 있다고 판단되는 결과가 나온 것을 확인할 수 있었다.

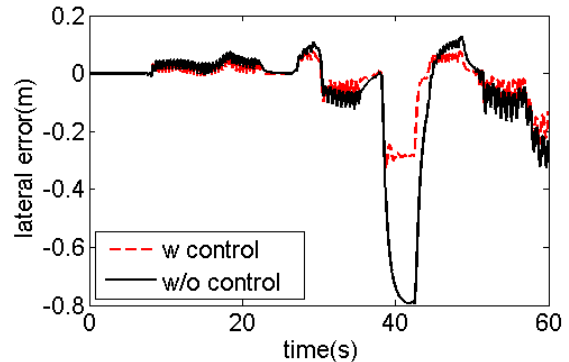


Fig. 5 Control of vehicle considering the tire slip (k=1)

#### 4. 타이어의 슬립제한을 통한 제어 알고리즘

하드웨어 성능 및 센서 오차를 포함한 시뮬레이션을 진행하기 위하여 실제 하드웨어의 특성을 반영하도록 하였다. 현재 성균관대학교에서 무인자동차 연구에 사용되고 있는 무인차량은 Fig 6와 같다 해당 무인차량을 대상으로 한 센서 노이즈 및 조향작동장치 성능은 Table 1과 같으며 실제 센서의 데이터가 아닌 후처리 된 센서의 결과를 기준으로 작성하였다.



Fig. 6 SKKU autonomous vehicle

Table 1 SKKU autonomous vehicle parameters

종방향 오차	max 2.0m
횡방향 오차	rms 0.2m
방위각 오차	rms 1degree
조향 반응속도	400degree/sec

해당 오차를 포함해서 슬립이 큰 시나리오를 만들기 위해서 식 (8)을 적용하여 전륜 타이어 슬립 고려한 제어방식을 CarSim의 차량모델을 사용한 시뮬레이션에서 차량 종 방향 속도 55km/h로 고정하여 진행하였다. 전륜 타이어 슬립이 적용된 알고리즘에서는 슬립이 발생함으로 인하여 보상 차원에서 더 큰 조향 입력을 주지만 전륜 타이어 슬립이 최대 횡력을 발생시키는 범위를 넘어가게 되면서 오히려 횡방향 오차가 커지는 결과를 초래하였다.<sup>6)</sup> 그 결과 하드웨어 특성을 반영하지 않은 시뮬레이션과 비교하여 더 큰 조향 명령을 주게 되더라도 Fig. 7과 같이 오히려 횡 방향 오차는 더 커지는 것을 확인할 수 있다.

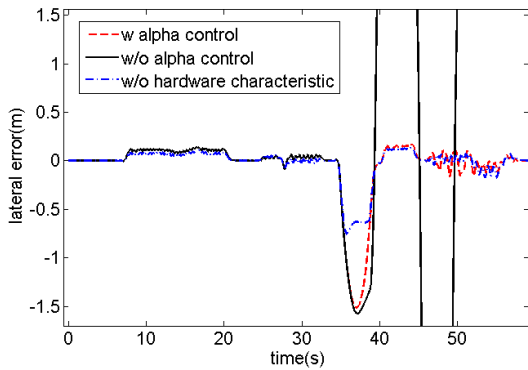


Fig. 7 Control of limit the  $\alpha_f$  (lateral error)

그리고 상대적으로 같은 목표 곡률을 추정하기 위한 조향값이 클수록 차량 거동이 불안정해지며 Table 1의 명기된 실제 실험차량의 노이즈와 특성을 고려하는 것만으로 기존의 알고리즘은 40초 부근에서 발산이 일어나는 것을 볼 수 있다. 발산이 일어난 40초 이후의 비교는 무의미하므로 발산을 야기한 33m 반경의 급격한 곡률부분인 35초에서 39초 부근을 확대해 보면 Fig. 8과 같다.

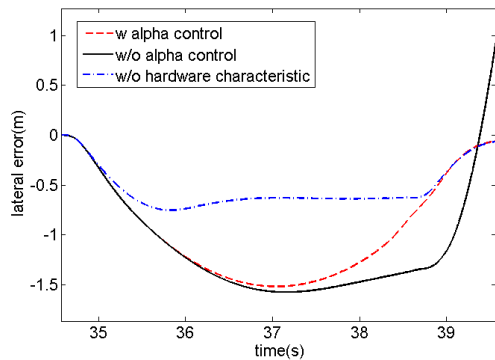


Fig. 8 Control of limit the  $\alpha_f$  (lateral error)

오차를 발생시키는 주요 이유 중 하나는 조향작동장치가 목표 조향까지 이동하는데 하드웨어의 한계로 인하여 지연이 생기기 때문이다. 이러한 지연은 횡 방향 오차를 키우게 되며 지나치게 큰 횡 오차 수정량을 만들게 된다. 이 경우 타이어 미끄럼각이 최대 횡력이 발생하는 미끄럼각을 넘어가기 때문에 오히려 선회가 부족하게 되었다고 판단하여 전륜 타이어 미끄럼각이 12도를 넘어가면 조향의 추가 조작을 방지하는 알고리즘을 추가하였다. 실제 SKKU 무인차의 계측데이터를 확인시  $\psi$ 의 노이즈가 최대 2도 정도 발생하였고 차량의 IMU로부터 획득한 횡방향 가속도를 통한  $\beta$ 계산 결과가 3도 가량 튀는 것을 반영하여 최대  $\pm 5$ 도 정도 노이즈로 인한 계산 오차가 발생할 것을 고려해야 한다. 시뮬레이션에 사용된 차량의 무게는 1,361kg이다. 바퀴당 동일한 수직력이 걸린다고 본다면 바퀴당 3,300N 정도의 수직력이 걸리게 된다. 보통 3,000N의 마찰원을 가지는 타이어가 7~10도 사이에서 최대 횡력 지점이 있다고 본다. 타이어 특성곡선에서 횡력과 미끄럼각의 관계는 낮은 미끄럼각에서는 높은 구배를 가지고 횡력이 증가하지만, 최대 횡력점과 그 이후에는 횡력의 변화가 크지 않다. 그래서 5도의 오차를 반영할 때 해당 오차로 인하여 최대 횡력점 이후의 미끄럼각을 가지는 것은 오차로 인하여 낮은 미끄럼 각에서 제어가 이루어지는 것보다는 부담이 적다고 할 수 있다. 그래서 7~10도 사이가 아닌 12도 부근을 타이어 한계점으로 보았고 최대 계산오차를 통한 전방 타이어의 미끄럼각은 7~17도 사이에서 제한이 일어날 것이다. 그 정도 범위에서는 구배가 낮아서 최대 횡력 값과 근사하게 나타날 것이다. 전륜 타이어 미끄럼각( $\alpha_f$ )를 제어에 반영한 결과는 Fig. 7과 Fig. 8의 w alpha control과 같다. Fig. 9은 Fig. 8의 각 시뮬레이션 당시 계산된 전륜 타이어의 미끄럼 각을 계산한 것이다.

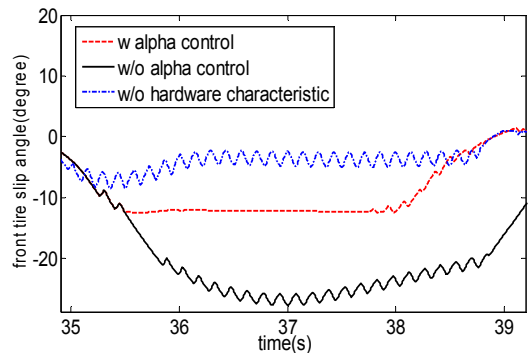


Fig. 9 Control of limit the  $\alpha_f$  (tire slip angle)

Fig. 7과 Fig. 8과 같이 조향작동장치의 지연 요소를 고려하면 지연요소를 고려하지 않은 경우와 비교하여 오차가 상당히 커지는 것을 알 수 있으나 적어도 전륜 타이어의 미끄럼각을 12도로 제한하는 제어로 해당 제어방식을 적용하지 않은 것과 대비하여 조향 입력은 최대 59% 덜 조작하였으며 오차는 발산구간을 제외하고 최대 57.1%가 개선되는 것을 확인할 수 있다.

### 5. 조향 반응속도를 고려한 제어 알고리즘

Fig. 8에서 슬립한계점을 지정하지 않은 차량은 (w/o alpha control) 오차 보정을 위하여 조향 입력으로 최댓값인 540도를 주었지만 오히려 더 많은 횡 오차를 발생시켰다. 해당 37초 부근은 0.0182의 곡률을 가진 도로 부근인데 해당 곡선도로에 대한 추종 오차는 슬립한계점을 통한 제어(w alpha control)에서 크게 이득을 보지 못하였다. 하지만 240도 수준의 작은 조향 입력을 사용함으로써 곡선이 끝나고 직진으로 들어서는 부분에서는 오차를 크게 줄일 수 있었다. 슬립한계점을 통한 제어를 하지 않은 경우 540도까지 틀어둔 조향을 직진도로에 맞게 중립조향으로 다시 돌려놓는데 1.35초가 걸렸고 이 시간 동안 차량은 직진 도로에서 벗어나고 결국 발산을 하게 되었다. 발산을 막는 방법으로는 횡 방향 오차에 대한 이득 값을 낮추는 방법과 횡 방향 오차를 통한 최대 수정량에 제한을 두는 방법이 있는데 이득 값을 낮추면 횡 방향 오차에 대한 수정량이 줄어들어 안정적인 운행이 가능하지만, 전체적인 오차가 커질 수 있으며 최대 수정량에 제한을 두면 안정적인 상황에서도 고정된 속도에서 수렴하게 된다.

요구되는 내용은 안정적인지 못한 상황에서는 횡 방향 오차에 대한 최대 수정량을 낮추고 안정적인 상황에서는 최대 수정량을 높여서 빠르게 수렴해야 한다. 그리고 안정적인 상황과 안정적인지 못한 상황의 결정은 조향작동장치의 속도로 인하여 발생할 오차로 판단한다. 조향작동장치의 속도로 인한 횡 오차는 식 (9)에서 중립조향으로 복귀하는데 걸리는 시간을 계산하여 조향이 중립조향으로 돌아오는데 까지 횡으로 얼마나 이동하는지 식 (10)과 같이 구할 수 있다. 여기에서  $act_{max}$  는 조향작동장치의 최대 속도이다.

$$t_{str} = \frac{\delta}{act_{max}} \tag{9}$$

$$e_s = \frac{t_{str} \delta v}{2} \tag{10}$$

조향작동장치 속도가 느릴수록 차량의 속도가 빠를수록 횡방향 오차는 커지게 된다. 조향으로 인한 횡 오차가 커질수록 불안정한 상황이라고 판단하여 최대 수정량을 낮추도록 알고리즘을 고려하였다. 알고리즘은 식 (11)와 같이 나타낼 수 있다.

$$e_y = \min(e_m, e_{max} - e_s) \tag{11}$$

Fig. 10은 시속 55km/h의 속도에서 0.0182, 0.0165의 서로 다른 곡률의 도로를 연속주행하는 시뮬레이션의 결과이다. 알고리즘의 최대 수정값인  $e_{max}$  는 차량이 정상적으로 차선 안에서 주행을 하고 있다고 가정할 때 차선 폭 3.8m와 실험차량인 소렌토의 차폭 1.88m를 가정하여 차선내의 목표 경로와 최대 1.92m의 오차가 발생할 수 있다고 판단하여 근사한 정숫값인 2m로 설정하였다. 비교대상 알고리즘은 단순히  $e_y$ 의 최댓값을 식 (11)과  $e_{max}$ 가 2m인 제어와의 비교와 최댓값이 줄어들었을 때의 결과를 비교하기 위해서 2m와 1m로 제한을 두었다.

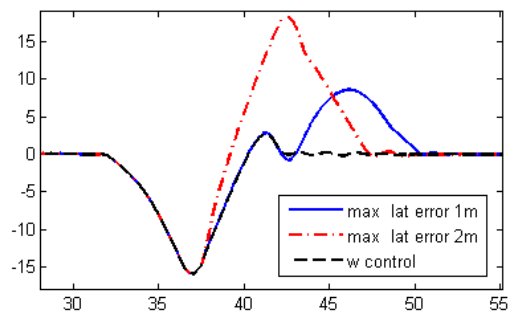


Fig. 10 Result of the maximum limit of steering to reduce lateral error

최대 수정량이 큰 경우 횡 방향 오차를 빠르게 제거할 수 있지만, 곡률이 다른 두 도로가 만나는 지점에서는 조향작동장치가 만든 오차로 인하여 쉽게 목표치에 수렴하지 못하는 것을 볼 수 있다. 반면 최대 수정량이 작은 경우 안정적인 수렴이 가능할 것으로 판단되었으나 연속적인 곡선도로에서 빠른 대응이 불가능한 이유로 횡 방향 오차가 다시 발생하는 것을 확인할 수 있다. 반면에 알고리즘을 적용한 경우에는 차량의 한계로 인하여 발생한 큰 오차에도 안정적으로 목표량에 도달하는 것을 확인할 수 있었다.

## 6. 시뮬레이션 결과

최종적으로 제안하는 알고리즘은 식 (12)와 같으며 전륜자동차의 최대 조향각이 30도 수준임을 고려하여 삼각함수를 무시하여 식을 더 간단하게 표현하였다.  $k(t)$ 는 식 (8)이며  $e_y$ 는 식 (11)의 변환된 내용으로 대체한다. 추가적으로 전륜 타이어 슬립각을 제한하는  $\alpha_f$  제어를 적용하였다. 비교대상인 기존 알고리즘은 기존의 Stanley방식이며 식 (8)과 식 (11) 그리고  $\alpha_f$ 를 제한하는 제어가 도입되지 않았다. Stanley방식의 횡 방향 오차보정식의 이득 값  $k$ 는 1로 고정되어 있다.

$$\delta(t) = e(t) + \frac{k(t) e_y}{v_x(t)} \quad (12)$$

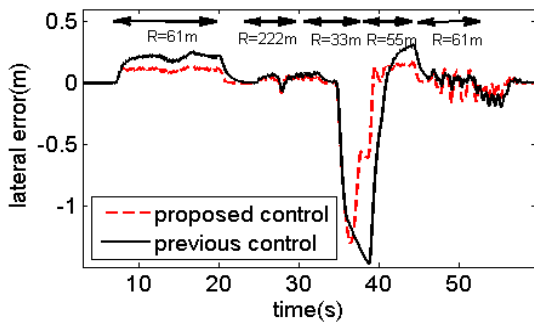


Fig. 11 Compare the proposed algorithm with Stanley method

시뮬레이션에는 Table 1에 제시된 조향 작동장치의 사양과 노이즈를 포함하여 진행되었으며 고속속도(55km/h)로 차량모델 CarSim에서 제공하는 모델을 사용하였으며 Simulink와의 Co-Simulation에서 Simulink로 부터 조향결과를 입력받아 시뮬레이션이 진행되었다. 알고리즘이 적용된 경우와 그렇지 않은 경우의 오차는 Fig. 11과 같다.

회전반경이 61m인 경우에 최대 52%, 거리로는 0.13m의 횡방향 오차 개선 효과가 있었으며 회전반경이 33m인 경우 최대 58.8%, 거리로는 0.86m가 개선된 것을 확인할 수 있었다. 곡률이 큰 지점일수록 오차 개선에 대한 효과가 좋은 것을 확인할 수 있었다. 오차가 최대로 개선된 부분의 애니메이션은 Fig. 12와 같다. Fig. 12의 붉은색 차량이 알고리즘을 적용한 경우이며 파란색 차량이 적용하지 않은 경우이다. 애니메이션에서 나타난 것과 같이 파란색 차량은 목표 경로인 도로의 중앙에서 0.86m 정도 벗어나 있는 것을 확인할 수 있었다.

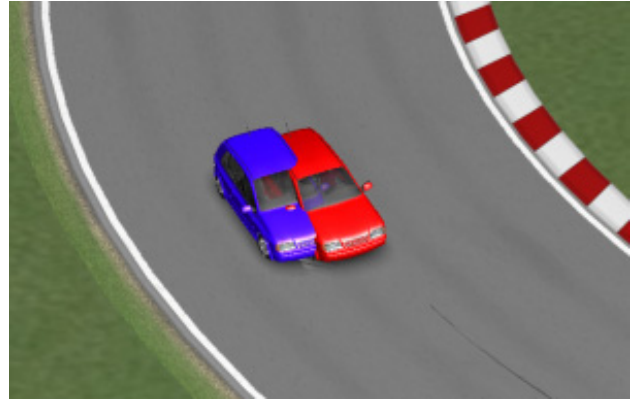


Fig. 12 Animation from computer simulation with CarSim

## 7. 결론

본 논문에서는 무인차량의 조향 제어를 전륜 타이어의 미끄럼각을 고려하여 제어하였다. 시뮬레이션은 MATLAB/Simulink와 CarSim을 통해 진행되었다. 시뮬레이션에 센서노이즈와 조향작동장치 특성을 반영하는 것을 통해 조향작동장치의 반응속도가 제어를 어렵게 만드는 요인임을 확인하였다.

전륜 타이어 미끄럼각을 이용하여 조향을 보상해주는 과정과 타이어의 횡력이 가장 큰 지점의 전륜 타이어 미끄럼각을 사용하는 과정, 마지막으로 조향 작동장치의 성능을 반영하는 과정으로 총 3가지의 추가적인 알고리즘을 제안하였다.

최종적인 결과로는 고속영역에서 가장 효과적인 알고리즘은 횡력이 가장 큰 지점을 사용하는 제어방식이었다. 미끄럼각을 보상하는 제어는 Fig. 7과 같이 차량의 노이즈와 조향작동장치의 특성을 고려하자 발산을 하는 것을 볼 수 있고 조향작동장치의 반응속도를 고려한 알고리즘은 Fig. 10과 같이 채터링의 감소에는 효과가 있지만 횡 방향 오차를 줄여주지는 못하는 것을 확인할 수 있었다. 하드웨어의 성능 부분에서는 우선 센서의 오차 성분은 목표 경로의 곡률과 상관없이 일정수준의 가우시안 분포를 보이게 된다. 하지만 Fig. 7에서 낮은 곡률의 도로에서는 차량의 노이즈를 반영하였음에도 불구하고 횡 방향 오차에는 큰 영향이 없는 것을 볼 수 있다. 또한, 같은 그림에서 곡률이 큰 지점, 즉 조향작동장치의 성능에 따라서 목표 조향까지의 지연이 크게 발생하는 부분에서 오차가 크게 발생한 것은 조향작동장치의 성능이 오차에 큰 영향을 준다는 간접증거며 이를 고려하여 조향 반응속도를 고려한 제어 알고리즘이 횡 방향 오차 성분을 줄이지는 못하였지만, 채터

링 현상을 크게 줄여주는 것을 확인함으로써 조향작 동장치의 성능향상이 횡 방향 오차와 채터링 현상을 모두에 크게 영향을 주는 요인임을 확인할 수 있다.

다만 횡력이 가장 큰 지점을 사용하는 제어방식에서 기본적으로 최대 계산오차를 5도로 예상하여 계산함으로써 인하여 차량은 항상 횡력이 가장 큰 지점을 지난 부근을 사용하게 된다. 최대 점을 지나게 되면 횡력이 떨어지는 구배가 낮아서 5도 수준까지는 횡력이 가장 큰 지점과의 차이가 크지 않을 것으로 예상하고 있으나 차체 미끄럼각을 더 자세하게 구할 수 있어 최대 계산오차를 줄일 수 있으며 가속도와 캠의 마찰원(Kamm's friction circle)을 고려하여 최대 횡력이 발생하는 지점을 실시간으로 계산해서 사용할 수 있으면 더 좋은 결과를 낼 수 있을 것으로 예상된다.

### 후 기

이 논문은 2013년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(NRF-2013R1A1A2005594)

### References

1) S. Thrun et al., "Stanley: The Robot that Won the DARPA Grand Challenge", J. of Field Robotics, pp.661-692, 2006.

2) T. Gillespie, Fundamentals of Vehicle Dynamics, SAE, pp.177-214, 1992.  
 3) S. Lee et al., "Development of Optimal Controller for Unmanned Vehicle", The Korean Society Of Automotive Engineers, pp.1104-1108, 2010  
 4) V. Subramnaian, T. F. Burks and W. E. Dixon, "Sensor Fusion Using Fuzzy Logic Enhanced Kalman Filter for Autonomous Vehicle Guidance in Citrus Groves", American Society of Agricultural and Biological Engineers, Vol. 52, No.4, pp.1411-1422, 2009.  
 5) Q. Li et al., "A Sensor-Fusion Driveable-Region and Lane-Detection System for Autonomous Vehicle Navigation in Challenging Road Scenarios", IEEE Transactions on Vehicular Technology, Vol.63, No.3, pp.540-555, 2014.  
 6) S. L. Koo, H. S. Tan, M. Tomizuka, "Nonlinear Tire Lateral Force Versus Slip Angle Curve Identification", American Control Conference, pp.2128-2133, 2004.