

## 주변차량의 안전을 고려한 자동 차선변경 알고리즘 개발

오 태 영<sup>1)</sup> · 손 원 일<sup>1)</sup> · 안 태 원<sup>1)</sup> · 이 용 기<sup>1)</sup> · 박 기 흥<sup>2)</sup>

국민대학교 자동차공학전문대학원<sup>1)</sup> · 국민대학교 자동차IT융합학과<sup>2)</sup>

### Development of Automated Lane Change Algorithm Considering Safety of Surrounding Vehicles

Taeyoung Oh<sup>1)</sup> · Wonil Son<sup>1)</sup> · Taewon Ahn<sup>1)</sup> · Yongki Lee<sup>1)</sup> · Kihong Park<sup>2)</sup>

<sup>1)</sup>Graduate School of Automotive Engineering, Kookmin University, Seoul 02707, Korea

<sup>2)</sup>Department of Automobile and IT Convergence, Kookmin University, Seoul 02707, Korea

(Received 21 December 2020 / Revised 2 January 2021 / Accepted 5 January 2021)

**Abstract** : In this paper, an automated lane change algorithm was developed in accordance with the recent government safety standards for level 3 autonomous vehicles. To this end, driving situations during automated lane change were divided into three control modes, and different control strategies were adopted for each mode. A novel lane change criterion was proposed that considers the safety of surrounding vehicles. When this criterion forbids immediate lane change, a location to initiate safe lane change was computed by solving a constrained optimization problem. For longitudinal control, an LQR-based optimal controller was designed with different state variables for each control mode. For lateral control, a control system consisting of path generation and path tracking was designed. The proposed algorithm was verified on four scenarios involving dangerous lane changing conditions. The results showed that the proposed algorithm could safely perform lane change without threatening the safety of surrounding vehicles. The algorithm could also provide comfort by performing lane change with small velocity changes.

**Key words** : Automated lane change(자동 차선변경), Lane change(차선변경), Autonomous vehicle(자율주행자동차), Inter-vehicle distance control(차간 거리 제어), ACC(적응형 순항제어)

#### 1. 서론

최근 국토교통부는 레벨3 부분자율주행 시스템을 탑재한 차량의 국내 출시 및 판매를 허가하고, 레벨3 자율주행자동차에 대한 안전기준을 공표하였다.<sup>1)</sup> 이는 자율주행 시험운행을 위해 각 차량마다 부여하는 지금까지의 임시허가가 아닌, 양산의 레벨3 자율주행자동차 판매를 허가하는 세계 최초의 법으로, 향후 많은 다른 나라에서 이 법을 벤치마킹하여 자국의 양산 자율주행자동차 판매를 위한 법 제정을 서두를 것으로 예상된다. 도입된 안전기준에 따르면, 레벨3 부분자율주행 시스템은 자동차 전용도로 및 고속도로 등의 지정된 작동영역 내에서 동작하며, 차량 스스로 주행 차선을 유지하는 기능과 더

불어, 방향지시기 작동과 같은 운전자의 차선변경 요청이 있을 때 자동으로 차선변경을 수행하는 기능이 있어야 함을 명시하고 있다.

본 논문에서는 자율주행자동차가 운전자의 지시에 따라 혹은 자율주행 로직의 판단에 따라 차선변경을 반드시 수행해야 하는 경우, 자차량과 주변차량의 안전을 고려해 안전하게 차선을 변경할 수 있는 알고리즘을 제안한다. 차선변경의 기본 메커니즘은 인지 센서를 활용한 주변 환경정보 취득, 취득된 데이터를 기반으로 차선변경 가능 유무 판단, 차선변경을 위한 경로생성, 변경할 차선으로 차량을 이동시키기 위한 종·횡방향 제어 등의 구조로 나타낼 수 있다. 이러한 메커니즘을 바탕으로 한 자동 차선변경 기능은 자율주행자동차가 목적지까지

\*Corresponding author, E-mail: kpark@kookmin.ac.kr

<sup>1)</sup>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.

이동하기 위해 필수적으로 요구되는 기술로 지속적인 연구가 진행되어 왔다. 이와 관련된 연구를 살펴보면 다음과 같다.

오태영 등<sup>2)</sup>은 로봇의 경로생성을 위해 사용되는 Artificial potential field 기법을 이용하여 자율주행자동차 차선변경 위험도 판단 및 경로생성에 관한 연구를 수행하였다. 이들은 자차량과 차선변경 시 목표가 되는 차량과의 상대속도를 고려하여 위험도를 판단하고 이를 기반으로 차선변경을 위한 경로를 생성하였다. 하지만 이 연구에서는 차선변경이 불가한 것으로 판단되는 경우 추가적인 제어 없이 차선변경을 포기하고 현재의 차선을 유지하는 방식을 택하고 있다.

신경식 등<sup>3)</sup>은 영상센서와 가상 차선을 이용한 자동 차선변경 알고리즘에 관한 연구를 수행하였다. 차선변경 위험도 판단을 위해 먼저 차선변경 소요시간을 최대 8초로 정해두고, 이를 고려하였을 때 측진방과 측후방 범위 내에 차량이 없는 경우를 차선변경 가능 상황이라고 판단하였으며, 그렇지 않은 경우에는 차선변경을 시도하지 않는 것으로 설계하고 있다.

김용주 등<sup>4)</sup>은 고속도로 공사구간에서 자율협력주행 차량이 동적합류와 차로변경을 하기 위한 통합제어전략에 관한 연구를 수행한 바 있다. 이 연구에서는 기본적으로 자차량을 포함한 주변의 모든 차량이 V2X를 통해서 모든 거동에 관한 정보를 공유한다는 가정을 하고 있어, 레벨3 부분자율주행 시스템의 양산 관점에서 주변 차량이 수동운전 차량인 경우를 다루지 못하고 있으며 아직까지는 곧바로 실행이 어려운 선행기술로 판단된다.

나원빈 등<sup>5)</sup>은 적응형 순항제어 상황에서 위험도 기반의 차선변경 최적 제어 알고리즘에 대한 연구를 수행하였다. 차선변경 수행 중 LQR 종방향 제어기의 상태변수의 거리오차에 TTC(Time to Collision)를 반영하고, 이를 반영하지 않는 제어기에 비해 성능을 향상시킨 것을 보이고 있다. 참고로 TTC는 전방차량과의 상대거리를 상대속도로 나눈 값으로 충돌이 임박한 정도를 시간으로 나타내는 물리량이다. 하지만 이 연구에서도 시나리오에 따라 차선변경을 하지 않는 것이 합리적이라고 언급하며 차선변경을 수행하지 않는 것을 볼 수 있다.

이렇듯 자동 차선변경과 관련된 많은 연구들이 있었지만, 대부분이 위험도 측면에서 차선변경을 하는 것이 어렵다고 판단되는 경우 차선변경을 포기하는 전략을 택하고 있어 앞서 언급한 국토교통부의 부분자율주행 시스템에 대한 안전기준을 맞추기 어렵다는 것을 알 수 있다.

심층강화학습을 이용한 차선변경 알고리즘 개발에 관한 연구가 있었다.<sup>6)</sup> 이 연구에서는 자율주행 또는 수동

운전 차량이 차선변경 지령을 줄 경우, 강화학습으로 학습된 인공신경망을 통해 차선변경의 가부를 결정하였으며, 이때 인공신경망의 입력정보는 주변차량의 속도 정보와 위치 정보를 사용하고 있다. 이 연구의 경우 실제 운전자의 성향을 반영할 수 있다는 장점은 가지고 있으나, 만족할만한 성능을 얻기 위해 학습에 장시간이 소요된다는 점과 학습하지 않은 상황이 발생하였을 때 알고리즘의 성능을 보장하지 못한다는 단점을 가지고 있다. 이홍석 등<sup>7)</sup>도 심화강화학습 기반의 자율주행자동차의 주행을 위한 알고리즘을 개발하였으며, 위 논문과 유사한 장단점을 가지고 있다. 따라서 두 논문 모두 실차에 적용하기 위해서는 수많은 학습을 필요로 하는데 이를 모두 학습시키는데는 현실적으로 한계가 있다는 것을 알 수 있다. 실제 운전자의 차선변경 특성을 반영한 연구도 박종철 등<sup>8)</sup>의 논문이 있었다. 이 논문에서는 자동 차선변경을 위한 전체 알고리즘을 제안하기 보다는 차선변경의 위험도 판단을 위해 사용될 수 있는 지표의 계수 값들을 실도로 차량 주행 데이터를 기반으로 추정하였다는 점에서 기여를 한 것으로 판단된다.

관련된 이전 연구 가운데 국토교통부의 안전기준에 가장 부합하는 성능을 낼 수 있는 연구로 채홍석 등<sup>9)</sup>의 논문을 들 수 있다. 이 연구에서는 차선변경을 위해 주변 차량의 정보와 주행모드를 기준으로 안전주행 영역을 정의하고 자차량이 안전주행 영역 내에 있을 때 차선변경을 수행하는 알고리즘을 구현하였다. 차선변경을 위한 종방향 및 횡방향 제어기로 MPC 즉 모델예측기법을 사용하였다. 하지만 이 논문의 경우 안전주행 영역을 판단함에 있어 자차량과 목표차량간의 상대속도만을 고려함으로써 차선변경 상황에 처한 두 차량 가운데 후방에 놓인 차량의 절대속도가 빠른 경우 차선변경을 강행하는 경우가 생길 수 있는 것으로 판단되며 이에 대한 분석은 7.4절에서 언급할 예정이다.

위와 같은 이전 연구에 대한 고찰을 바탕으로, 본 논문에서는 자차량이 운전자의 지시 또는 자율주행 로직의 판단에 따라 차선변경을 수행해야 하는 경우, 주변차량과의 안전을 고려해 차선변경을 수행하는 알고리즘을 제안한다. 본 알고리즘은 전방차량과 변경할 차선에 존재하는 측방차량들에 대한 위험도를 함께 고려하고, 이를 바탕으로 차선변경 가능 여부를 결정한다. 만약 차선변경이 불가능하다고 판단되면, 측방차량들에 대한 안전거리를 확보하기 위해 먼저 주행 차선 내에서 거리제어를 수행한다. 거리제어에서는, 차선변경을 시작할 수 있는 자차량의 위치를 최적화 문제의 해를 통해 도출하고, 이렇게 구해진 최적 위치로 종방향 제어를 통해 자차량을 이동시킨다. 그 후 측방차량들과의 안전거리가 확

보된 것을 확인하고 차선변경을 수행하게 된다.

본 논문에서 제안하는 알고리즘의 개발 및 검증에 위해 Matlab/Simulink와 CarMaker를 사용하였으며, 자차량과 변경할 차선에 존재하는 차량과의 상대거리와 상대속도를 조정하여 자차량의 차선변경 위험도를 높이는 주행상황을 시나리오로 구성하여 검증을 수행하였다.

## 2. 시스템 아키텍처

2장에서는 본 연구에서 개발한 자동 차선변경 알고리즘의 전체 구성을 설명한다. 자차량이 자율주행을 하고 있는 상황에서 차선변경 요청이 들어오면, 자차량은 먼저 변경할 차선에 존재하는 차량들과의 안전을 고려하여 주행모드를 결정하고 이에 맞는 제어를 수행해야 한다. 이를 위해 본 연구에서는 차선변경 상황과 관련된 자율주행 모드를 세 가지로 구분하였으며, 이는 ACC (Adaptive Cruise Control) 모드, LC(Lane Change) 모드, DC(Distance Control) 모드이다. Fig. 1은 이들 세 가지 모드로 구성된 본 논문의 자동 차선변경 알고리즘의 아키텍처를 나타낸다.

ACC 모드는 자차량이 현재 차선에서 자율주행을 하도록 해준다. 자차량의 전방에 차량이 없는 경우, 운전자가 설정한 속도로 등속도 제어를 하며, 전방에 근접한 차량이 있는 경우 등간격 제어를 하게 된다. 등간격 제어 시 전방차량과의 차간거리는 전방차량의 현재 속도와 타임갭(Time gap)을 곱한 값으로 정해지는데, 타임갭의 설정에 관해서는 뒤에서 설명하기로 한다. 엄밀한 의미에서 ACC는 자율주행을 위한 종방향 제어만을 의미하지만 본 연구에서의 ACC 모드라 함은 자차량이 현재 차선에서 자율주행을 수행하기 위해 작동하는 종방향과 횡방향 제어기 모두를 포함한다. 이들 로직에 대한 자세

한 내용은 5장과 6장에서 다뤄진다. ACC 모드는 차선변경 요청이 들어오기 직전까지 자차량의 자율주행을 위해 작동되며, 또한 차선변경이 모두 완수되고 나면 다시 ACC 모드가 작동되어 자차량이 변경된 차선에서 자율주행을 재개할 수 있게 해준다.

Fig. 1에서와 같이, ACC 모드에서 자율주행을 하고 있던 자차량은, 운전자의 지령이나 자율주행 로직에 의해 차선변경 요청(LC Request)이 들어오면, 먼저 센서로부터 취득한 자차량과 주변차량의 거동정보를 바탕으로 차선변경 위험도 판단지표(LC Criterion)를 계산한다. 이 판단지표를 통해 현재 시점에서 차선변경이 가능하다고 판단되면, 알고리즘은 ACC 모드에서 곧바로 차선변경을 위한 LC 모드로 천이가 되어, 자차량이 전방차량과 변경할 차선의 차량들에 대해 안전거리를 유지하며 차선을 변경할 수 있도록 종방향 제어와 횡방향 제어를 수행한다. 하지만 반대로 현재 시점에서 차선변경이 불가능하다고 판단되면, 알고리즘은 DC 모드로 천이가 되어, 차선유지를 위한 횡방향 제어와 함께, 현재 차선 내에서 차선변경을 수행하기에 안전한 위치로 자차량을 이동하기 위한 종방향 제어를 수행한다. DC 모드가 ACC 모드와 다른 점을 살펴보면, ACC 모드는 주변차량 가운데 전방차량만을 고려하며 등속도 제어 또는 등간격 제어를 하는 반면, DC 모드는 전방차량, 측전방차량, 측후방차량 모두를 동시에 고려하며 자차량의 위치와 속도제어를 한다는 점이 차이가 있다. LC 모드와 DC 모드의 알고리즘과 관련된 보다 자세한 내용은 4장~6장에서 설명한다.

## 3. 차선변경 위험도 판단

본 연구의 알고리즘은 자차량이 주행하는 동안 센서로부터 취득한 주변차량의 거동정보와 자차량의 거동정보를 바탕으로 차선변경에 대한 위험도 판단지표를 계산하며, 운전자의 지령이나 자율주행 로직에 의해 차선변경 요청이 들어오면 이 판단지표를 기준으로 자차량의 차선변경 여부를 결정한다. 차선변경 위험도 판단지표는 Fig. 1에서 LC criterion으로 표시되어 있으며 이번 장에서는 이 지표에 대해 자세히 설명한다.

본 연구에서는 차선변경 위험도 판단지표(이하  $i_{LC}$ )를 계산하기 위해, 차선변경 요청이 들어온 시점을 기준으로 총 4대 차량의 거동정보를 고려한다. 이 차량은 자차량, 전방차량, 측전방차량, 측후방차량이다. 전방차량은 자차량과 동일한 차선에서 자차량 바로 앞에 위치한 차량을 의미하며, 측전방차량은 변경할 차선에 있으면서 자차량의 앞쪽에 위치한 차량, 측후방차량은 변경할 차선에 있으면서 자차량의 뒤쪽에 위치한 차량을 의미한다.

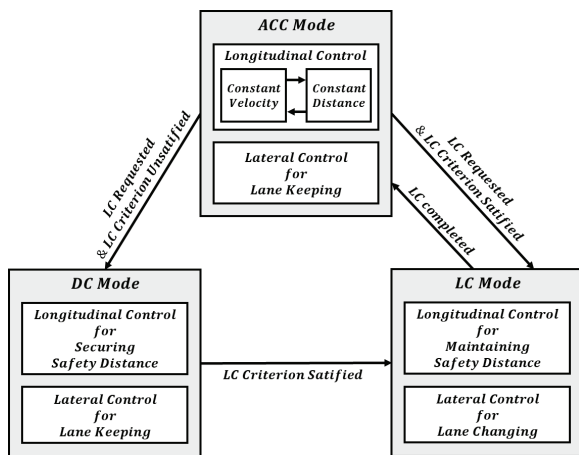


Fig. 1 Architecture of automated lane change system

차선변경 위험도를 도출하기 위해 자차량은 먼저 변경할 차선에 존재하는 차량들의 안전거리를 계산한다. 안전거리란 자차량이 안전하게 차선변경을 수행하기 위해 변경할 차선의 차량들에 대해 확보되어야 할 거리를 의미하며, 자차량의 속도와 측방차량의 속도를 반영하여, 식 (1)과 같이 결정하였다.<sup>8)</sup>

$$c_{safety,sf} = \tau_{abs} v_e + \tau_{rel}(v_e - v_{sf}) \quad (1a)$$

$$c_{safety,sr} = \tau_{abs} v_{sr} + \tau_{rel}(v_{sr} - v_e) \quad (1b)$$

식 (1)의  $c_{safety,sf}$ 와  $c_{safety,sr}$ 는 각각 측전방차량과 측후방차량에 대한 안전거리를 나타낸다.  $v_e$ 는 자차량(Ego vehicle)의 속도이고,  $v_{sf}$ 와  $v_{sr}$ 는 각각 측전방차량과 측후방차량의 속도이다.

식 (1a)의 측전방차량에 대한 안전거리는 두 개의 항으로 이루어져 있는 것을 볼 수 있다. 첫째 항은 자차량의 절대속도를 반영한 항으로, 자차량의 속도가 빠를수록 이 안전거리는 커지게 된다. 두 번째 항은 자차량과 측전방차량 간의 상대속도를 반영한 항으로, 두 차량 간의 상대속도가 0인 경우 이 안전거리도 0이 되고, 두 차량의 간격이 좁혀지는 상황에서는 이 안전거리가 커지게 되며, 두 차량의 간격이 넓어지는 상황에서는 이 안전거리는 줄어들게 된다. 식 (1b)의 측후방차량에 대한 안전거리는 선행차량이 자차량이고 추종차량이 측후방차량인 점만 차이가 나고 안전거리가 계산되는 방법은 식 (1a)와 동일하다.

식 (1)에서  $\tau_{abs}$ 와  $\tau_{rel}$ 는 위에서 설명한 절대속도를 반영한 안전거리항과 상대속도를 반영한 안전거리항에 각각 곱해지는 타임갭 개념의 가중치로, 차선변경에 있어 적극성과 안전성이라는 상반되는 목표 사이에 대해 절충점을 얻을 수 있게 해주는 튜닝 파라미터이다.  $\tau_{abs}$ 의 경우, 값이 커질수록 절대속도를 반영한 안전거리가 커져 보다 넓은 차간에서 차선변경을 수행하게 되어 안전성을 향상시키고, 작아질수록 안전거리가 작아져 보다 좁은 차간에서 차선변경을 수행하게 되어 적극성을 향상시킨다.  $\tau_{rel}$ 의 경우, 값이 커질수록 상대속도를 반영한 안전거리항의 증가비율 또는 감소비율이 극대화된다. 다시 말해, 측방차량이 속도차를 두어 자차량에 접근하는 경우를 예로 들면  $\tau_{rel}$ 이 커질수록 보다 방어적(안전성)으로 차선변경을 수행하게 되며, 작아질수록 보다 공격적(적극성)으로 차선변경을 수행하게 된다. 본 논문에서는 차선변경을 수행하는 도중 측전방차량이 급제동하여 발생할 수 있는 위험성과 자차량의 무리한 차선변경으로 인한 측후방차량에서 발생할 수 있는 위험성 등을

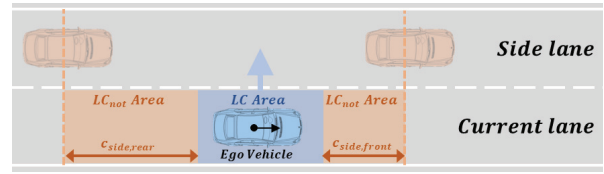


Fig. 2 Schematics of lane change criterion

종합적으로 고려하여 적극성보다는 안전성에 좀 더 비중을 두는 전략을 선택하였으며 다양한 시뮬레이션을 통한 튜닝의 결과로  $\tau_{abs}$ 와  $\tau_{rel}$ 의 값을 각각 1.2초와 0.8초로 설정하였다.

식 (1a)에서 측전방차량에 대한 차선변경을 위한 최종 안전거리가 계산되면, 현재의 측전방차량과의 상대거리가 이 값보다 큰 지를 판단한다. 또한 동일하게 식 (1b)에서 측후방차량에 대한 차선변경을 위한 최종 안전거리가 계산되면, 현재의 측후방차량과의 상대거리가 이 값보다 큰 지를 판단한다. 그런 다음 이 두 가지가 동시에 만족되면 차선변경이 가능한 것으로 판단하고 그렇지 않으면 차선변경이 불가능한 것으로 판단한다. 식 (2)는 이렇게 도출된 차선변경 위험도 판단지표  $i_{LC}$ 를 나타낸다.

$$i_{LC} = i_{sf} \cap i_{sr} \quad (2a)$$

$$\text{where } \begin{cases} i_{sf} := |d_{sf} - c_{safety,sf}| \geq 0 \\ i_{sr} := |d_{sr} - c_{safety,sr}| \geq 0 \end{cases} \quad (2b)$$

Fig. 2는 자차량이 차선변경을 하려는 순간 변경할 차선에 전방차량과 후방차량이 존재하는 경우, 차선변경 위험도 판단지표가 계산되는 과정에서 도출되는 차선변경 안전영역과 불안정영역을 도식적으로 나타낸다.

현재의 위치에서는 측전방차량과 측후방차량에 대한 안전거리가 확보되어 차선변경을 곧바로 수행할 수 있지만, 자차량이 현재위치보다 전방이나 후방에 위치하게 되면 측전방차량과 측후방차량에 대한 안전거리 중 하나가 만족되지 않아, 차선변경을 곧바로 수행할 수 없다는 것을 알 수 있게 된다.

#### 4. 차선변경 최적 위치 결정

자차량이 차선변경 요청을 받은 시점에서 차선변경 판단지표가 차선변경 불가 판정을 내리게 되면, LC 모드가 바로 수행되지 못하고 이에 앞서 DC 모드가 수행된다. DC 모드에서는 먼저 현재 주행 차선 내에서 차선변경을 안전하게 수행할 수 있는 최적의 위치를 계산하는데, 본 논문에서는 식 (3)과 같은 구속조건이 있는 최적

화 문제를 세우고 이에 대한 최적해를 구하여 이를 차선 변경을 시작할 수 있는 현재 차선 내의 좌표로 사용하였다. 식 (3)의  $f(x,y)$ 는 최적 위치를 결정하기 위한 목적함수로 자차량의 지역좌표계에 대한 좌표값  $(x,y)$ 로 표현되고 있으며, 현재 자차량의 위치에서 최소의 거리에 있는 위치를 찾게 해준다.

$$\text{Minimize } f(x,y) = x^2 + y^2 \quad (3a)$$

subject to :

$$g_1 = d_{sf} - c_{safety,sf} \geq 0 \quad (3b)$$

$$g_2 = d_{sr} - c_{safety,sr} \geq 0 \quad (3c)$$

$$g_3 = d_p - c_p \geq 0 \quad (3d)$$

$$h = y - \alpha_0 - \alpha_1 x - \alpha_2 x^2 - \alpha_3 x^3 = 0 \quad (3e)$$

식 (3b)-(3e)는 네 개의 구속조건을 나타낸다. 식 (3b)와 (3c)는 차선변경이 가능한 영역 내부의 최적 위치를 찾기 위한 부등식의 구속조건으로, 식 (2)에서 이미 설명된 바 있다. 식 (3d)는 자차량의 최적 위치가 전방차량보다 앞에 위치하는 것을 방지하기 위한 부등식으로,  $d_p$ 는  $(x,y)$ 에서 전방차량까지의 거리이며,  $c_p$ 는 자차량에서 전방차량까지의 거리이다. 식 (3e)는 현재 주행 중인 자차량의 경로를 3차 다항식으로 나타낸 것으로, 최적 위치가 자차량의 현재 주행 차선 위에 놓이게 하기 위한 구속조건이다. 현재 주행 중인 도로가 직선인 경우에는 식 (3e)의 계수들은 모두 0이 되고 식 (3a)의 목적함수에서도  $y$ 를 제외할 수 있지만, 최적 위치로의 이동이 곡선도로에서 이루어져야 하는 경우는  $y$ 를 포함하는 것이 필요하다.

앞서 정의한 목적함수와 구속조건들을 통해 라그랑주 함수  $L$ 을 식 (4)와 같이 정의하였다.<sup>10)</sup>

$$L = f(x,y) + \sum_{i=1}^3 \lambda_i (g_i + s_i^2) + \lambda_4 h \quad (4)$$

위 식에서  $\lambda_i$ 는 라그랑주 승수(Lagrange multiplier)를 나타내며,  $s_i$ 는 부등식의 구속조건에 대한 슬랙변수(slack variable)를 나타낸다. 식 (4)의 라그랑주 함수를 최적화 변수, 라그랑주 승수, 슬랙변수에 대해 편미분을 수행하여 식 (5)의 방정식의 해를 구하면 차선변경이 가능한 영역 내부의 최적 위치를 도출할 수 있다.

$$\nabla L = \left( \frac{\partial L}{\partial x}, \frac{\partial L}{\partial y}, \frac{\partial L}{\partial \lambda_i}, \frac{\partial L}{\partial s_j} \right) = 0 \quad (5)$$

where  $i = 1,2,3,4$  &  $j = 1,2,3$

최적 위치는 앞서 3장에서 도출한 차선변경 안전영역 내부에 위치하며, 전방차량과 측방차량에 대해 안전거리가 확보된 위치이다. 또한, 자차량의 최소 거동만으로 안전하게 차선변경을 수행할 수 있는 위치이다.

### 5. 종방향 제어 알고리즘

자차량은 앞서 소개한 세 가지 모드 중 ACC 모드의 등속도 제어 모드를 제외한 나머지 모든 모드에 대해 주변차량 가운데 하나를 목표차량으로 설정하고 자신의 차속을 제어한다. 주행모드에 따라 목표차량은 자차량과 동일한 차선에 있는 전방차량이 될 수도 있고 또는 변경할 차선에 있는 측방차량이 될 수도 있다. 이번 절에서는 본 연구에서 설계한 LQR 최적제어 기반의 종방향 제어 알고리즘을 소개한다.

식 (6)은 자차량과 목표차량 사이의 관계를 나타내는 상태방정식이다.

$$\begin{aligned} \dot{\chi} &= A\chi + Bu \\ &= \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} \chi + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u \end{aligned} \quad (6a)$$

where

$$\chi = [(c_d - c_{target}), (v_{target} - v_e)]^T \quad (6b)$$

위 식에서  $u$ 는 자차량의 가속도이다.  $c_d$ 는 목표차량에 대한 제어요구거리를,  $c_{target}$ 는 목표차량과의 상대거리를,  $v_{target}$ 은 목표차량의 속도를,  $v_e$ 는 자차량의 속도를 나타낸다. Fig. 3은 이들 사이의 관계를 보여준다.

식 (6b)의 상태변수  $\chi$ 는 2개의 변수로 구성되는데 첫째 상태변수는 거리의 목표치에 대한 오차를 의미하며, 두 번째 상태변수는 속도의 목표치에 대한 오차를 의미한다. 본 연구에서는 ACC 등간격 제어 모드, DC 모드, LC 모드의 세 가지 주행모드에 대해 동일한 LQR 제어를 통해 자차량의 목표가속도를 계산하였다. 하지만 상태변수를 구성하는 값들을 각 모드 마다 다르게 구성함으로써 세가지 주행모드 마다 그 특성에 부합하는 최적

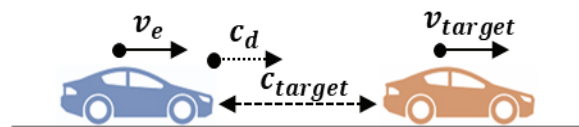


Fig. 3 Relationship between ego vehicle and target vehicle

의 종방향 제어가 수행될 수 있도록 알고리즘을 구성하였다.

먼저 자차량의 목표가속도 계산을 위해 식 (7)과 같은 형태의 비용함수를 정의하였다. 이 식에서  $Q$ 와  $R$ 은 설계인자로, 자차량의 거리 목표치에 대한 오차에 대한 가중치, 자차량의 속도 목표치에 대한 오차에 대한 가중치, 자차량의 가속도 변화에 대한 가중치의 값들로 구성된다. 본 연구에서는 자차량과 주변차량의 안전을 고려하며 많은 튜닝 과정을 거쳐 이들 가중치의 값을 선정하였다.

$$J = \int_0^{\infty} (\chi^T Q \chi + u^T R u) dt \quad (7a)$$

$$\text{where } Q = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \quad (7b)$$

행렬형 Ricatti 방정식으로부터 식 (7)의 비용함수를 최소화하는 최적계인을 찾아, 식 (8)의 상태변수 피드백을 통해 자차량의 종방향 목표가속도를 계산하였다.

$$a_{des} = -K\chi \quad (8a)$$

$$a_{des}^* = \begin{cases} a_{\max} & \text{if } a_{des} > a_{\max} \\ a_{\min} & \text{if } a_{des} < a_{\min} \\ a_{des} & \text{else} \end{cases} \quad (8b)$$

식 (8b)는 비록 식 (7)의 비용함수 속에 자차량의 가속도 크기를 제한하기 위한 항이 포함되어 있지만 식 (8a)의 상태변수 피드백을 통해 계산된 목표가속도 크기가 여전히 큰 값을 가질 수 있기 때문에 이를 적절한 최대값 이내의 값으로 제한하기 위한 식을 나타낸다. 또한, 식 (8b)는 운전자의 승차감을 저해하는 급가속, 급감속과 같은 거동을 방지하기 위해 필요하다. 가속도의 제한으로 인해 안전거리의 확보를 위한 종방향 제어 성능이 떨어지고 이로 인해 차선변경 시 위험이 초래되는 상황이 우려될 수 있다. 하지만 차선변경의 최종 결정은 실시간으로 판단되는 주변차량들과의 안전거리를 기준으로 내려지기 때문에, 가속도의 제한은 차선변경 착수 시점을 조금 지연시킬 수는 있으나 자차량과 주변차량의 안전을 저해하는 요인으로 작용하지는 않는다.

위와 같이 LQR 제어를 통해 자차량의 목표가속도가 계산되면, 이를 가감속 액츄에이터의 입력값으로 인가하여 차량의 종방향 거동을 제어하게 된다. 실제 차량의 경우 목표가속도를 차량 사시 CAN에 직접 입력하는 방식을 많이 사용하여 종방향 제어를 수행하기 때문에, 본 논문에서도 액츄에이터 제어를 위한 하위제어기를

별도로 구성하지 않고, 가감속 액츄에이터 제어를 포함하는 CarMaker에 목표가속도를 바로 인가하여 시뮬레이션을 수행하였다.

본 연구에서는 ACC 등간격 제어 모드, DC 모드, LC 모드의 세 가지 주행모드에 대해 위에서 소개한 LQR 제어를 동일하게 사용한다. 하지만 상태변수를 구성하는 값들을 각 모드에 대해 다르게 구성함으로써 세 가지 주행모드마다 그 특성에 부합하는 최적의 종방향제어가 되도록 알고리즘을 구성하였다. 아래에서는 각 모드에 대한 상태변수의 구성을 설명한다.

식 (9)는 ACC 등간격 제어 모드의 상태변수 구성요소로, 전방차량을 목표차량으로 설정한 경우이다.

$$\chi = [(c_d - c_{target}), (v_{target} - v_e)]^T \quad (9)$$

$$\text{with } \begin{cases} c_d = c_0 + \tau v_p \\ c_{target} = c_p \\ v_{target} = v_p \end{cases}$$

위 식에서 제어요구거리는 안전제동거리인  $c_0$ , 타임 갭  $\tau$ , 전방차량의 속도  $v_p$ 로 구성되며,  $c_p$ 는 전방차량과의 상대거리이다.

식 (10)은 DC 모드의 상태변수 구성요소로, 변경할 차선의 차량 중 안전거리가 확보되지 않은 차량을 목표차량으로 설정한 경우이다.

$$\chi = [(c_d - c_{target}), (v_{target} - v_e)]^T \quad (10)$$

$$\text{with } \begin{cases} c_d = \sqrt{(x_{side} - x_{opt})^2 + (y_{side} - y_{opt})^2} \\ c_{target} = c_{side} \\ v_{target} = v_{side} \end{cases}$$

위 식에서 제어요구거리는, 변경할 차선의 차량에 대한 안전거리를 확보해야 하므로 변경할 차선의 목표차량의 위치  $(x_{side}, y_{side})$ 와 4장에서 도출한 차선변경 최적 위치  $(x_{opt}, y_{opt})$ 사이의 거리로 구성하였다.  $c_{side}$ 와  $v_{side}$ 는 각각 변경할 차선의 차량과의 상대거리와 이 차량의 속도를 나타낸다.

식 (11)은 LC 모드의 상태변수 구성요소로, 자차량이 전방차량과 변경할 차선의 차량에 대해 안전거리를 유지하며 차선변경을 수행해야 할 것을 고려하여, 두 차량의 위험도를 반영한 하나의 가상차량을 도출하고 이를 목표차량을 설정한 경우이다. 이는, 두 차량의 거동정보를 통해 하나의 가상차량을 도출한 점에서 이전 연구와는 차별성이 있다.<sup>9)</sup>

$$\chi = [(c_d - c_{target}), (v_{target} - v_e)]^T \quad (11)$$

$$\text{with } \begin{cases} c_d = \beta(c_0 + \tau v_{side}) + (1 - \beta)(c_0 + \tau v_p) \\ c_{target} = \beta c_{side} + (1 - \beta) c_p \\ v_{target} = \beta v_{side} + (1 - \beta) v_p \end{cases}$$

먼저 두 차량의 위험도를 고려한 하나의 가상차량을 도출하기 위해 변경할 차선으로의 횡방향 수렴 정도를 반영하는 가중치  $\beta$  를 도입하였고, 식 (11)과 같이 전방 차량과 변경할 차선의 차량에 대해 ACC 등간격 제어 모드에서 사용되었던 상태변수 구성요소에 각각  $1 - \beta$  와  $\beta$  의 가중치를 부여하였다. 이렇게 함으로써 제어요구거리, 상대거리, 절대속도가 변경할 차선으로 자차량이 진입하는 정도에 따라, 전방차량과 변경할 차선의 차량에 대해 반영 비율이 달라지게 하였다. 다시 말해, 자차량은 차선변경을 수행하는 과정 동안, 초기에는 전방차량에 더 높은 비중을 두어 거리제어를 수행하고 후기에는 변경할 차선의 차량에 대해 더 높은 비중을 두어 거리제어를 수행하게 된다. 따라서, 자차량은 차선변경 수행 중 두 차량에 대해 적절한 안전거리를 유지하며 차선을 변경하고, 차선변경을 완료한 후에는 ACC 모드로 부드럽게 전환된다.

## 6. 횡방향 제어 알고리즘

Fig. 1에서 볼 수 있듯이 자동 차선변경 시스템을 구성하는 세 가지 모드는 모두 횡방향 제어를 수행한다. ACC 모드와 DC 모드에서는 현재의 차선을 유지하기 위한 횡방향 제어를 수행하며, LC 모드에서는 차선변경을 위한 횡방향 제어를 수행한다. 이번 장에서는 본 연구에서 사용된 횡방향 제어 알고리즘에 관해 설명한다.

자동 차선변경 시스템의 횡방향 거동 계획은 목표경로 생성과 이를 추종하기 위한 경로추종 제어로 구성된다. 최근 자율주행을 위한 경로생성과 경로추종에 대한 연구는 활발히 진행되어 왔으며, 다양한 방법론을 적용한 알고리즘들이 제안되었다.<sup>11-13)</sup> 본 연구의 목적은 차선변경을 위한 고도화된 횡방향 제어알고리즘의 개발이 아니므로 본 논문에서는 비교적 간단하고 직관적으로 구현할 수 있는 경로생성 기법과 경로추종 기법을 사용하였다.

### 6.1 차선변경을 위한 경로생성

앞서 설명한 것과 같이 ACC 모드와 DC 모드에서는 차선유지를 위한 횡방향 제어를 수행해야 하며, 이때의 목표경로는 자차선의 중심선이 된다. 자차선의 중심선 경로를 생성하는 것은 이미 영상 센서를 이용하거나 HD

Map를 이용한 많은 연구가 진행된 상태이기 때문에 본 연구에서는 따로 다루지 않는다. 이와 달리 LC 모드에서는 차선변경을 위한 횡방향 제어를 수행해야 하는데, 이번 절에서는 본 논문의 LC 모드에서 수행된 차선변경을 위한 경로생성에 대해 설명하고자 한다.

본 논문에서는 Cubic spline curve를 이용하여 차선변경을 위한 경로를 생성하였으며, 식 (12)는 경로를 나타내는 3차 다항식이다.

$$S(x) = b_3x^3 + b_2x^2 + b_1x + b_0 \quad (12)$$

식 (13)은 차선변경 경로 특성을 고려하여, 식 (12)의 3차 다항식의 계수 값을 결정하기 위한 네 가지 조건이다.

$$\begin{aligned} S(x_0) &= y_0 \\ S(x_f) &= y_f, \quad (x_f = d_0 + v_e \tau_s) \\ S'(x_0) &= 0 \\ S'(x_f) &= \tan \theta_f, \quad \left( \theta_f = \tan^{-1} \left( \frac{y_f - y_{f-1}}{x_f - x_{f-1}} \right) \right) \end{aligned} \quad (13)$$

위 식에서  $(x_0, y_0)$ 는 자차량의 현재 위치이고,  $(x_f, y_f)$ 는 변경할 차선의 한 점이다. 차량이 안전하게 차선변경을 수행하기 위해 고려되어야 할 차선변경 경로의 특성에는, 고속 주행에서 급격한 조향 입력을 피해야 하며 변경할 차선에 정확히 수렴하여 수렴 후 차선유지를 위한 경로추종에 지장이 없어야 한다는 점을 들 수 있다. 이를 위해 변경할 차선의 한 점  $(x_f, y_f)$ 을 결정하는 요소로 자차량의 속도를 반영하였다.

### 6.2 경로추종

본 논문의 자동 차선변경 시스템의 세 가지 모드에서 차선유지 또는 차선변경을 위한 경로가 생성되고 나면, 이를 추종하는 횡방향 제어가 수행된다. 본 연구에서는 이를 위해 Pure Pursuit 경로추종 기법<sup>14)</sup>을 적용하였다.

이 기법은 Fig. 4와 같이 목표경로를 추종하기 위해 경로 위의 한 점을 예견점(Look ahead point)으로 설정하고, 예견점을 지나기 위한 자차량의 요구조향각을 생성한다. 예견점은 차량의 경로추종 성능과 조향 안정성을 결정하는 중요한 파라미터로, 본 논문에서는 식 (14)와 같이 자차량의 속도를 반영하여 예견점을 설정하였다.

$$L_d = \kappa v_e \quad (14)$$

$$\delta_{des} = \frac{l}{R} = \frac{2l \sin(\theta_{err})}{L_d} \quad (15)$$

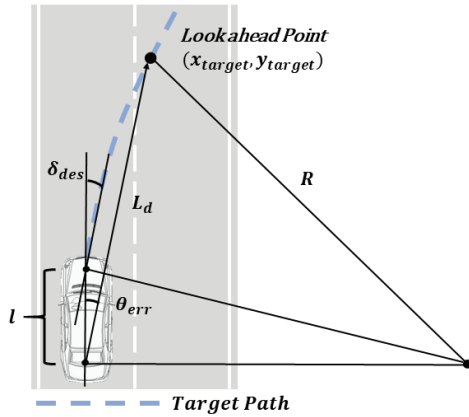


Fig. 4 Pure Pursuit path tracking algorithm

식 (14)에서  $L_d$ 는 자차량의 속도( $v_e$ )를 고려한 목표경로 위의 예견점까지의 거리를 나타내며,  $\kappa$ 는 차선유지와 차선변경의 두 가지 상황에 대해 각각의 적합한 예견점을 설정하여 자차량의 안정적인 요구조향각을 생성하기 위한 파라미터이다. 식 (15)는 목표경로를 추종하기 위한 최종 요구조향각( $\delta_{des}$ )을 나타낸다. 이 식에서  $l$ 은 차량의 축거이고,  $\theta_{err}$ 은 자차량 좌표계 기준의 예견점의 요각도를 나타내며,  $R$ 은 회전 반경이다.

위와 같이 Pure Pursuit 알고리즘을 통해 자차량의 요구조향각이 계산되면, 이를 조향 액츄에이터의 입력값으로 인가하여 차량의 횡방향 거동을 제어하게 된다. 실제 차량의 경우 별도의 액츄에이터를 장착하지 않는 한, 요구조향각을 차량 사시 CAN에 직접 입력하는 방식을 많이 사용하여 제어를 수행한다. 이에 따라 본 논문에서도 조향 액츄에이터 제어를 위한 하위제어를 별도로 구성하지 않고, 조향 액츄에이터 제어를 포함하는 CarMaker에 요구조향각을 바로 인가하여 시뮬레이션을 수행하였다.

### 7. 주행 시나리오 및 시뮬레이션 결과

제한한 알고리즘을 검증하기 위해 본 연구에서는 다양한 차선변경 시나리오에 대한 시뮬레이션을 수행하였으며, 이번 장에서는 그 결과에 대해 설명하고자 한다. 자차량이 차선변경 요청을 받았을 때, 차선변경을 곧바로 실행할 수 있는 상황부터 주변차량으로 인해 차선변경을 곧바로 수행하는 것이 불가능한 상황까지 총 4가지의 주행 시나리오로 구성하여 본 알고리즘의 타당성을 검증하였다.

Fig. 5는 네 가지 차선변경 시나리오를 나타낸다. 각각의 시나리오에는 4대의 차량, 즉 자차량, 전방차량, 측전방차량, 측후방차량이 존재한다. 전방차량은 자차량과

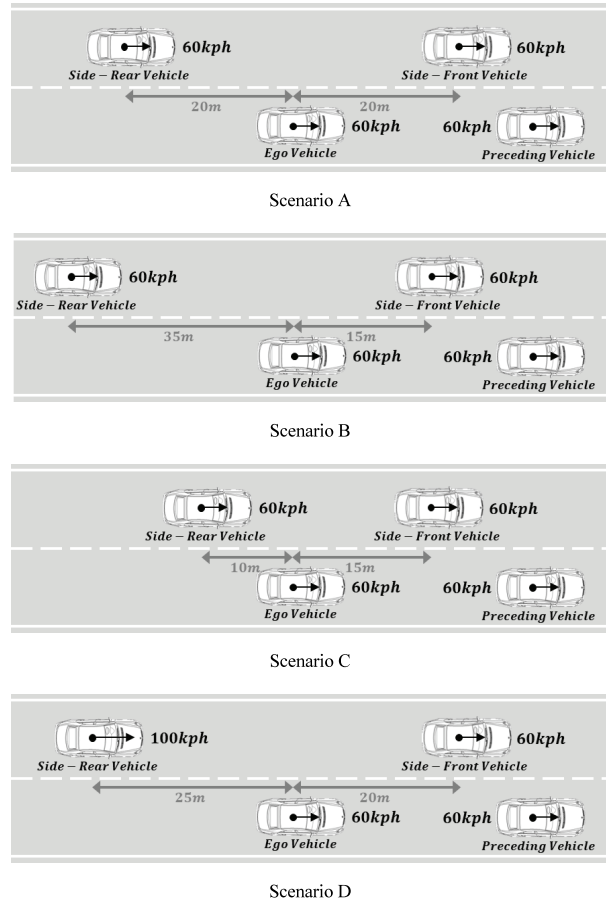


Fig. 5 Scenarios used in lane change simulation

동일한 차선에서 자차량 바로 앞에 위치한 차량을 의미하며, 측전방차량은 변경할 차선에 있으면서 자차량의 앞쪽에 위치한 차량, 측후방차량은 변경할 차선에 있으면서 자차량의 뒤쪽에 위치한 차량을 의미한다.

네 가지 시나리오 모두, 자차량은 60 km/h로 전방차량을 추종 중인 상황을 가정하였다. 시나리오 A, B, C는 측전방차량과 측후방차량이 자차량과 같은 60 km/h의 등속주행을 하는 경우이며, 두 차량이 동일한 속도로 주행하고 있기 때문에 두 차량 사이의 간격은 차선변경 전까지는 초기의 간격으로 유지가 되는 상황이다. 시나리오 D는 측후방차량이 100 km/h의 빠른 속도로 접근한 후 급감속하여 측전방차량을 추종하는 상황이며, 자차량은 측후방차량이 자차량 후방 25 m 지점에 도달한 시점에서 차선변경 지령을 받는 경우이다. 이는 앞의 세 가지 시나리오에 비해 충돌위험이 매우 큰 경우에 해당한다.

Fig. 5의 각각의 개별 시나리오는 특이점을 가지며 이를 비교하여 검증할 수 있다. 시나리오 A, B, C는 측방차량이 자차량과 같은 속도로 등속주행하지만 두 차량 사이의 거리 간격만을 달리한 경우로, 차선변경을 위



해 주어진 측방차량의 사이 간격이 다른 상황에서 본 알고리즘이 어떠한 동작을 수행하는지를 비교할 수 있다. 또한, 시나리오 B, C는 위와 동일한 맥락이지만, 자차량이 측방차량들과 넓은 차간을 둔 경우와 좁은 차간을 둔 경우의 차이를 통해 비교하여 검증할 수 있다. 마지막으로, 시나리오 A, D는 측후방차량이 특정 거리 간격을 두고 자차량과 같은 속도로 등속 주행하는 경우와 급가속하는 경우로 이를 통해 측후방차량이 자차량과 극심한 속도 차를 두고 있을 때, 어떠한 동작을 하는지 비교하여 검증할 수 있다.

차선변경 알고리즘의 검증을 위한 시나리오를 구성함에 있어 적절한 타임갭을 설정하는 것은 중요하다. 타임갭은 자차량이 일정 거리를 유지하며 전방차량을 추종할 때 최소의 안전을 위해 확보해야 하는 거리를 시간으로 나타낸 지표로서, ISO 22179<sup>15)</sup>에서는 이 값을 1.5초~2.2초 사이의 값으로 설정할 것을 규정하고 있다. 본 논문의 4가지 시나리오에서 모두, 자차량은 차선변경 지령 직전까지 이 값 가운데 가장 좁은 간격인 1.5초의 타임갭으로 전방차량을 추종하고 있는 상황을 가정하였다. 또한, 각각의 시나리오 별 주변차량의 경우, ACC 모드를 사용하는 것으로 가정하였다. ACC 등속도 제어 모드의 설정 속도는 Fig. 5의 각각의 시나리오에서 나타내는 속도와 같이 설정하였고 ACC 등간격 제어 모드의 설정 타임갭은 나이와 성별에 따른 운전자의 수동운전 주행 특성을 분석한 이전 연구 결과<sup>16)</sup>를 참조하여 시나리오 A와 B의 주변차량은 각각 1.8초, 시나리오 C, D의 주변차량은 1.5초로 설정하였다.

차선변경 가능 여부에 관한 판단 기준을 정하는데 있어 본 논문에서는 차선변경의 적극성보다는 변경할 차선에 존재하는 차량과의 안전성에 높은 비중을 두었다. 측후방 차량을 기준으로한 자차량에 대한 타임갭과, 자차량을 기준으로한 측전방 차량에 대한 타임갭이, 각각 1.2초 이상 확보가 된 경우를 차선변경이 가능한 상황이라고 보았다. 이는 자차량이 차선변경을 수행하는 상황에서, 측전방 차량이 급제동하여 충돌이 발생할 수 있는 상황을 방지하고, 또한 자차량의 차선변경으로 인해, 변경한 차선의 측후방 차량이 급감속하여 위험에 이르는 상황을 방지하기 위해 정한 수치이다.

아래에서는 네 가지 시나리오에 대한 시뮬레이션 결과를 소개하고 이를 분석한다. 차선변경 알고리즘의 작동 시점은 네 가지 경우 모두 10초에서 운전자의 방향지시등 작동 또는 자율주행 로직의 판단에 의해 차선변경이 시작되는 상황을 가정하였다. 자차량의 주행모드는 ACC, LC, DC로 구분된다. ACC는 차선변경 요청 직전까지 자차량이 전방차량을 일정 간격으로 추종하는 상황

을 나타내며, LC(Lane Change)는 자차량이 차선변경을 하고 있는 과도상태를 나타낸다. DC(Distance Control)는 자차량이 차선변경을 바로 수행할 수 없어 최적의 위치로 이동하고 있는 상태를 나타낸다.

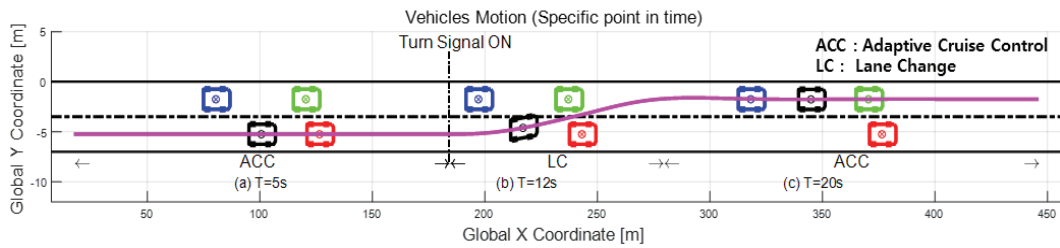
### 7.1 시나리오 A

Fig. 6은 시나리오 A의 시뮬레이션 결과를 나타낸다. Fig. 6(a)는 각각 5초, 12초, 20초 시점에서의 주행상황을 나타낸다. 자차량이 전방차량을 타임갭 1.5초로 추종 중인 상황에서, 10초 또는 190 m 근방에서 차선변경 요청이 들어온 경우이다. 10초 이전까지 측전방차량과 측후방차량은 동일한 60 km/h의 속도로 달리고 있다.

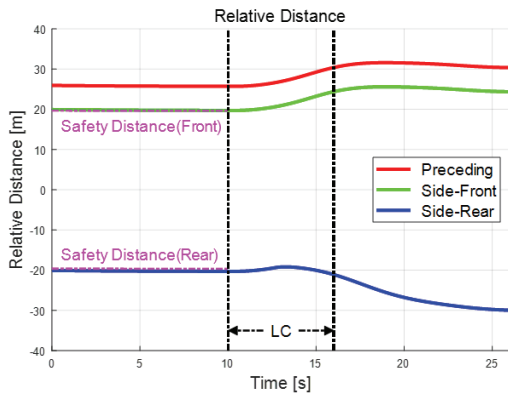
10초 이전까지 측후방차량은 측전방차량과 40 m의 거리 간격으로 달리고 있고 이는 측전방차량의 타임갭 앞서 설정한 1.8초보다 큰 간격이기 때문에 측후방차량의 ACC 등간격 제어 모드는 작동하지 않은 상태라고 볼 수 있다. Fig. 6(b)와 Fig. 6(c)는 자차량에서 본 전방차량과 측방차량들의 상대거리와 상대속도를 보여준다. 차선변경 요청 시점에서 측전방차량과 측후방차량의 상대거리가 각각 20 m와 -20 m이다. 이는 자차량의 속도와 측방차량의 속도를 통해 도출한 식 (1)의 측전방안전거리 및 측후방안전거리 20 m를 만족하므로 자차량은 안전거리 확보를 위한 추가 거동 없이 바로 차선변경을 수행한다.

Fig. 6(c)에서 자차량은 차선변경(LC) 영역에서 측전방차량에 대해 감속하며 최대 5 km/h의 속도차를 만들어 내는 것을 볼 수 있는데, 이는 차선변경 이후 측전방차량을 자신의 새로운 전방차량으로 삼아 타임갭 1.5초로 추종하기 위한 전략으로 볼 수 있다. 차선변경을 보다 적극적으로 하도록 알고리즘을 설계하면 위의 상대속도는 이보다 더욱 커지게 되는데, 본 연구에서는 앞서 설명한 것과 같이 차선변경의 적극성보다는 변경할 차선에 존재하는 차량과의 안전성에 높은 비중을 두어 비교적 부드러운 상대속도 변화를 얻도록 하였다.

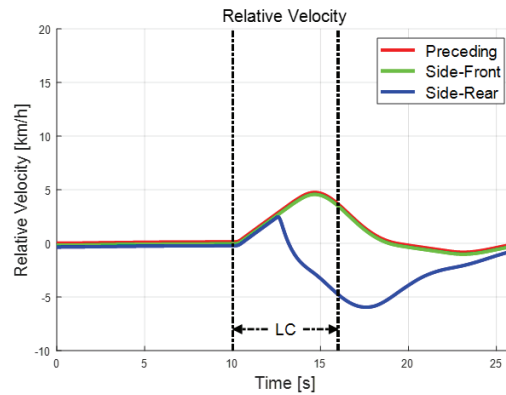
Fig. 6(c)를 보면 자차량은 차선변경 요청이 들어온 10초부터 감속을 하고 있지만, 측후방차량은 약 12.5초부터 감속을 하는 것을 볼 수 있는데, 이는 자차량이 차선변경을 시작하더라도 측후방차량이 자차량을 자신의 차선으로 진입하는 차량으로 인지하는데 일정 시간이 걸리기 때문이다. 측후방차량은 이후 자차량에 비해 최대 6 km/h의 속도차가 나도록 감속을 하는 것을 볼 수 있는데, 이는 측후방차량이 자차량의 차선변경 이후 자차량을 자신의 새로운 전방차량으로 삼아 타임갭 1.8초로 추종하기 위함이다. Fig. 6의 그래프들로부터 자차량의 차선변경은 측후방차량에 위협을 가하지 않는 범위 내에서 수행된 것을 알 수 있다.



(a)



(b)



(c)

Fig. 6 Simulation results of scenario A

Fig. 6(b)로부터, 차선변경 이후에 자차량은 측전방차량에 대해 타임갭 1.5초 즉 차속 60 kph 기준 25 m의 간격을 유지하기 위해 종방향 제어를 하며, 측후방차량은 자차량에 대해 타임갭 1.8초 즉 차속 60 kph 기준 30 m의 간격을 유지하기 위한 종방향 제어를 하는 것을 볼 수 있다. 전방차량과 측전방차량의 경우 자차량의 차선변경과 차속변경 그리고 측후방차량의 차속변경에 전혀 영향을 안받으며 60 km/h의 설정 속도로 계속 주행하는 것을 볼 수 있다.

이 시나리오는 평이한 시나리오로 보일 수 있지만, 본 연구의 차선변경 위험도 판단 알고리즘이 차선변경 요청 시점에서 안전 조건을 만족하였다고 판단하여 즉시 차선을 변경한 결과로, 본 시나리오를 통해 적절한 판단이 수행된 것을 확인할 수 있었다.

## 7.2 시나리오 B

Fig. 7은 시나리오 B의 시뮬레이션 결과를 나타낸다. Fig. 7(a)는 각각 5초, 12초, 18초, 25초 시점에서의 주행 상황을 나타낸다. 자차량이 전방차량을 타임갭 1.5초로 추종하는 도중 10초 지점에서 차선변경요청이 들어왔으며, 하지만 변경할 차선의 차량들과 안전거리가 확보되지 않아, 거리제어(DC)를 수행한 다음 차선변경을 수행하는 경우이다. 10초 이전까지 측후방차량은 측전방차

량과 50 m의 거리 간격으로 달리고 있으며, 이 거리는 측후방차량의 타임갭 1.8초보다 큰 간격이기 때문에 측후방차량의 ACC 등간격 제어 모드는 작동하지 않은 상태라고 볼 수 있다.

차선변경은 자차량을 기준으로 측전방차량과 측후방차량 모두 안전거리가 확보되었을 때 수행될 수 있으나, Fig. 7(b)를 보면 10초 지점에서 측후방차량은 안전거리가 확보되었으나 측전방차량은 안전거리가 확보되지 않은 것을 알 수 있다. 이에 따라 자차량은 측전방차량에 대해 안전거리를 확보하기 위해 약 3.5초 동안 거리제어(DC)를 수행한 다음 그 후 약 6.5초에 걸쳐 차선변경(LC)을 수행한다.

Fig. 7(c)에서 자차량은 거리제어(DC) 영역에서 측전방차량에 대해 최대 6 km/h의 속도차가 나도록 감속을 한다. 자차량의 속도 변화로 인해 Fig. 7(b)에서 보듯이 측전방안전거리는 감소하며 측후방안전거리는 증가(크기에 있어서)한다. 거리제어(DC)를 수행한 지 3.5초가 지난 13.5초 시점에서 자차량에 대한 측전방안전거리와 측후방안전거리가 모두 확보가 되어 그 때부터 자차량의 차선변경(LC)이 시작된다. 차선변경은 약 6.5초에 걸쳐 진행이 되는데 차선변경 시작 후 2초 동안은 측전방차량에 대해 최대 7 km/h의 속도차가 나도록 추가로 감속하며, 그 이후 4.5초 동안은 가속하여 차선변경 이후의 타

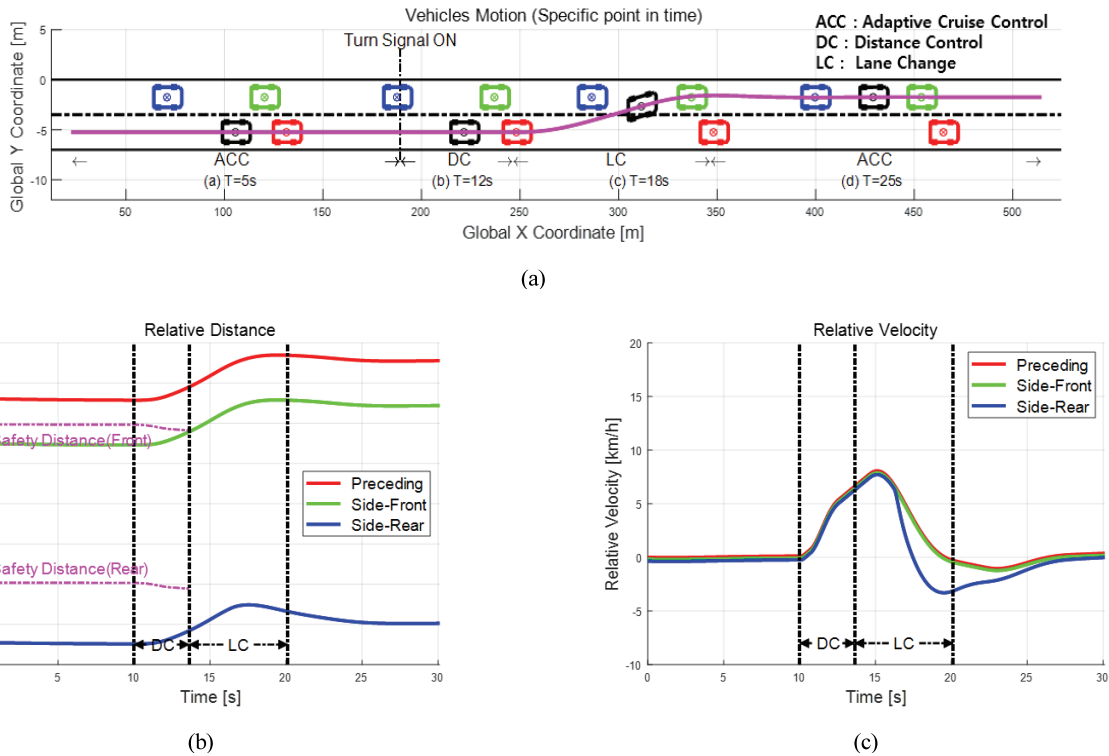


Fig. 7 Simulation results of scenario B

입껐 제어에 대비하는 것을 볼 수 있다.

Fig. 7(b)로부터, 차선변경 이후 자차량은 측전방차량에 대해 타임갭 1.5초의 간격을 유지하기 위해 종방향 제어를 하며, 측후방차량은 자차량에 대해 타임갭 1.8초의 간격을 유지하기 위한 종방향 제어를 하는 것을 볼 수 있다. 전방차량과 측전방차량의 경우 자차량의 차선변경과 차속변경 그리고 측후방차량의 차속변경에 전혀 영향을 안받으며 60 km/h의 설정 속도로 계속 주행한다.

Fig. 7의 그래프들로부터 시나리오 B의 주행상황에 대해 자차량은 주변차량들에 위협을 초래하지 않으며 차선변경을 잘 수행하는 것을 확인할 수 있다.

시나리오 B에서는 시나리오 A와 달리, 측전방차량에 대한 안전거리 미충족으로 인해 즉시 LC 모드로 진입하지 못하고 DC 모드 진입 후 충분한 안전거리가 확보되었다고 판단하였을 때 LC 모드로 천이된 것을 확인할 수 있었다.

### 7.3 시나리오 C

Fig. 8은 시나리오 C의 시뮬레이션 결과를 나타낸다. Fig. 8(a)는 각각 5초, 14초, 23초, 32초 시점에서의 주행 상황을 나타낸다. 자차량이 전방차량을 타임갭 1.5초로 추종하는 도중 10초 지점에서 차선변경 요청이 들어왔으며, 하지만 변경할 차선의 차량들과 안전거리가 확보

되지 않아, 거리제어(DC)를 수행한 다음 차선변경(LC)을 수행하는 경우이다. 10초 이전까지 측후방차량은 측전방차량을 앞서 설정한 타임갭 1.5초로 추종하고 있는 경우이다.

Fig. 8(b)를 보면 10초 지점에서 측전방차량과 측후방차량 모두 차선변경을 위한 안전거리가 확보되지 않은 것을 알 수 있다. 이에 따라 자차량은 측전방차량에 대해 안전거리를 확보하기 위해 약 11초 동안 거리제어(DC)를 수행한 다음 그 후 약 7초에 걸쳐 차선변경(LC)을 수행한다.

Fig. 8(c)에서 자차량은 거리제어(DC) 영역에서 측전방차량에 대해 최대 17 km/h의 속도차가 나도록 감속을 하는데, 이는 앞의 두 시나리오의 경우에 비해 상당히 큰 값임을 알 수 있다. 이에 대한 원인을 찾기 위해, 11초 동안의 거리제어(DC) 영역을 크게 처음 2초와 그 이후의 두 영역으로 나눠서 살펴볼 수 있다. 처음 2초 동안 자차량의 감속으로 측전방차량과의 상대속도가 약 10 km/h로 벌어지면서 측전방차량에 대한 안전거리가 확보된다(Fig. 8(b)). 하지만 이러한 감속으로 인해 측후방차량과의 안전거리는 오히려 악화되고, 그 사이 측후방차량이 자차량의 바로 측면까지 근접하게 된다. 이에 따라 측방차량들의 현재 거동이 계속 유지되는 한 측전방차량과 측후방차량의 사이 공간으로는 차선변경을 수행할 수

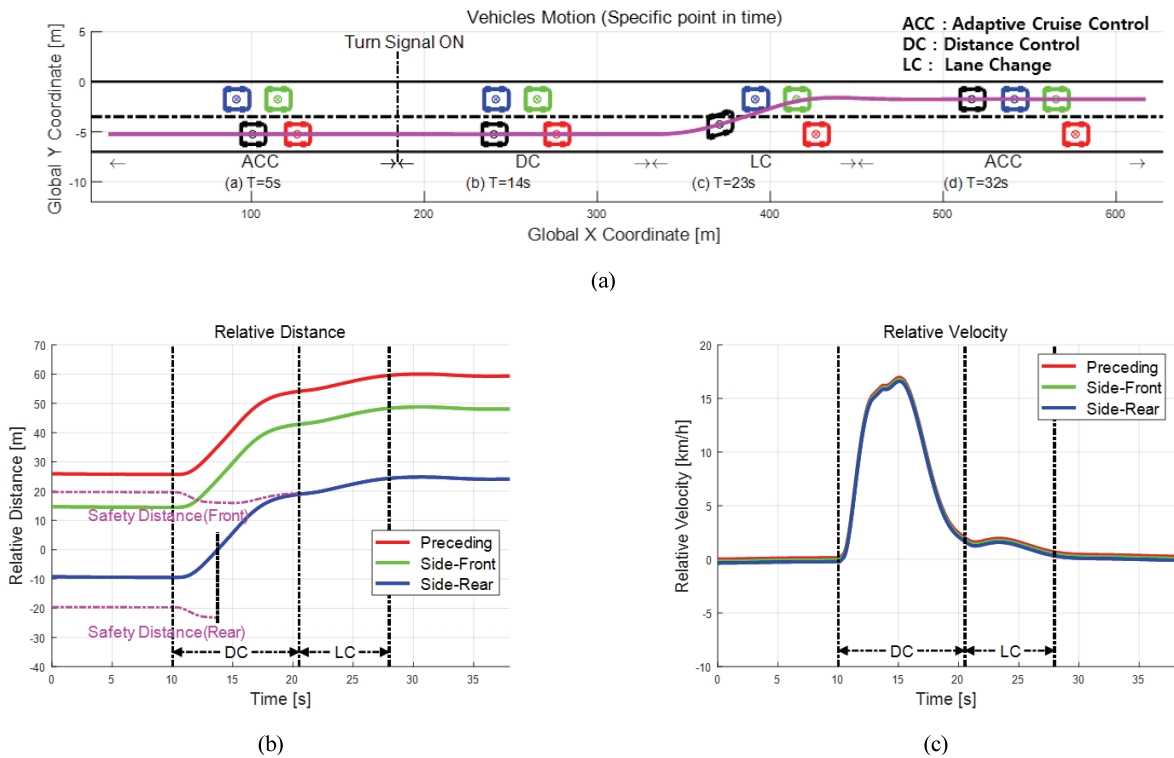


Fig. 8 Simulation results of scenario C

없다고 판단하고, 본 알고리즘은 측후방차량의 후방으로 가서 차선변경을 하는 판단을 내리게 된다. 이에 추가적인 감속을 통해 측방차량들과 최대 17 km/h의 속도 차이를 만들게 되고, 측후방차량의 뒤쪽에서 안전거리를 확보하게 된다.

Fig. 8(b)를 보면, 14초 시점에서 측후방차량의 상대거리가 음수에서 양수로 바뀌는 것을 볼 수 있는데 이는 측후방차량이 자차량을 지나쳐 앞으로 가는 것을 의미한다. 이에 따라 안전거리에 대한 기준도 후방안전거리에서 전방안전거리로 전환된다. 추가로 Fig. 8(c)를 보면 거리제어(DC) 영역의 약 15초 시점부터 자차량은 가속하여 측방차량들과의 속도 차를 줄이는 것을 볼 수 있다. 이는, 이후 차선변경을 수행하는 상황에서 변경할 차선의 차량과 과도한 속도 차를 두는 것이 바람직하지 않기 때문에 수행한 올바른 거동이라 할 수 있으며, 이로 인해 자차량은 차선변경(LC) 영역에서 완만한 감속 즉 측방차량과 속도차를 최대 2 km/h만 발생시키며 타임갭 추종 모드로 진입하는 것을 볼 수 있다.

Fig. 8(c)에서 세 차량의 상대속도가 모두 동일하게 나오는 것은 이들 세 차량이 자차량의 거동에 전혀 영향을 안 받고 동일한 속도로 등속주행을 하기 때문이다.

Fig. 8의 그래프들로부터 숙련된 운전자들에게도 상당히 어려울 만한 시나리오 C의 주행상황에 대해, 본 연

구의 알고리즘은 주변차량들에 위험을 초래하지 않으며 자차량의 차선변경을 잘 수행하도록 하는 것을 확인할 수 있다.

시나리오 C에서는 시나리오 B와 달리, 두 측방차량에 대한 안전거리를 모두 미충족한 상황으로 본 알고리즘은 측방차량 사이의 차선변경 가능 공간이 없다고 판단하여 측후방차량의 후방으로 이동 후 차선변경을 수행한 것을 확인할 수 있었다.

#### 7.4 시나리오 D

Fig. 9는 시나리오 D의 시뮬레이션 결과를 나타낸다. 자차량이 전방차량을 타임갭 1.5초로 추종하는 도중 10초 지점에서 차선변경 요청이 들어왔지만 측후방차량이 고속으로 근접하며 자차량의 차선변경 위험도를 높이고 있어 자차량은 거리제어를 수행한 다음 차선변경을 시도하는 경우이다. Fig. 9(a)는 각각 5초, 14초, 21초, 28초 시점에서의 주행상황을 나타낸다.

Fig. 9(b)를 보면, 처음 5초 동안 측후방차량이 자차량의 80 m 이상 후방에 위치하고 있어 자차량의 센서가 이를 인지하지 못하고 있다. 약 5초 시점부터 측후방차량이 인지되며 측후방차량의 높은 차속이 반영되어 후방 안전거리가 이전의 시나리오에 비해 매우 큰 것을 볼 수 있으며 차선변경 요청이 있는 10초 지점에서 측후방차

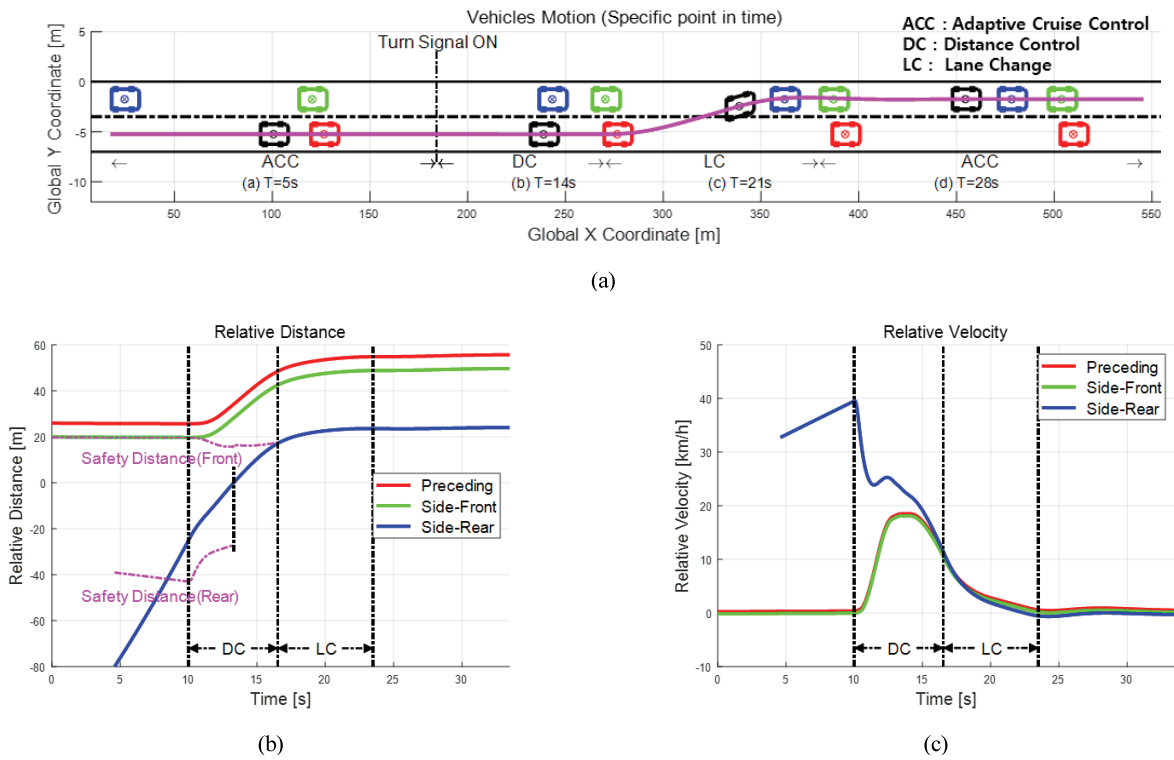


Fig. 9 Simulation results of scenario D

량에 대한 안전거리가 확보되지 않은 것을 볼 수 있다. 본 알고리즘은 측전방차량과 측후방차량에 대한 상대거리와 안전영역을 고려하여 측전방차량과 측후방차량의 사이 공간으로는 차선변경을 수행할 수 없다고 판단하며 이에 따라서 자차량을 측후방차량의 후방으로 이동하기 위한 거리제어를 수행한다. 거리제어(DC)는 약 7초에 걸쳐 수행되며, 그 후 약 6초에 걸쳐 차선변경(LC)이 수행된다.

Fig. 9(c)에서 자차량은 거리제어(DC) 영역에서 측전방차량에 대해 감속하며 최대 19 km/h의 속도차가 나도록 감속을 하는데, 이는 지금까지 시나리오 중에서 가장 큰 값임을 알 수 있다. 이러한 감속을 통해 측후방차량의 상대거리는 급격히 작아지다가 약 12.5초 시점에는 양수로 전환되는데 이는 자차량이 측후방차량의 후방으로 이동한 것을 의미한다. 측후방차량이 자차량의 전방에 위치하게 되면서 본래 측후방차량이 측전방차량으로 전환되는데, 이는 Fig. 9(b)의 12.5초에서 전방안전거리 그래프에 약간의 불연속적 변화로도 확인할 수 있다. 이 불연속 변화의 크기가 크지 않은 것은 Fig. 9(c)에서 볼 수 있듯이 12.5초 시점에서 측후방차량과 측전방차량의 속도의 차이가 4 km/h 정도로 그리 크지 않기 때문이다.

추가로, Fig. 9(b)에서 10초 이후 후방안전거리가 시나리오 B, C에서와 달리 빠르게 작아지는 것은 측후방차량

이 주행 중 가까워진 측전방차량과 일정 거리를 유지하기 위해 급감속하는 정도가 자차량이 차선변경을 위한 거리제어를 하기 위해 감속하는 정도보다 훨씬 크기 때문이다.

거리제어(DC) 영역의 종료 시점에서는 모든 차량의 상대거리가 전방안전거리의 상단에 위치하여 안전거리가 확보되었음을 확인할 수 있다. 이를 통해, 시나리오 D에서도 측방차량에 대해 안전거리를 확보하기 위한 거동이 잘 수행되었다고 볼 수 있다.

자차량은 차선변경(LC) 영역에서 자신보다 앞서 가는 측후방차량과의 속도 차를 줄이며 차선변경을 수행하고 있으며, 차선변경을 완료한 후에는 측후방차량에 대해 1.5초의 타임갭을 유지하며 추종 중인 것을 확인할 수 있다.

Fig. 9의 그래프들로부터 차선변경에 매우 악의적이며 자칫 큰 사고로 이어질 수 있는 시나리오 D의 주행상황에 대해서도, 본 연구의 알고리즘을 통해 자차량은 주변차량들에 위협을 초래하지 않으며 차선변경을 잘 수행하는 것을 확인할 수 있다.

시나리오 D에서는 시나리오 A와 달리, 측후방차량이 자차량과 큰 속도차를 두고 급가속하여 접근하는 경우로 본 알고리즘은 측방차량 사이의 차선변경 가능 공간이 없다고 판단하여 측후방차량의 후방으로 이동 후 차선변경을 수행한 것을 확인할 수 있다.

추가로 시나리오 D에 대해, 이전 연구 가운데 가장 우수하다고 판단되는 알고리즘<sup>9)</sup>이 적용되었을 때의 결과를 예상해보면, 차선변경 지령이 내려진 10초의 순간에 측전방차량 및 측후방차량과의 안전거리가 충분히 확보된 것으로 판단되어 차선변경을 곧바로 수행할 것으로 예상된다. 하지만 이 순간 자차량과 측후방차량과의 상대거리가 25 m, 상대속도가 40 km/h이고 측후방차량이 자차량의 차선변경을 인지하는데 1~2초 정도의 시간이 걸리는 것을 감안하면 자차량이 차선변경을 완료하고 난 후 혹은 완료하기 전에 측후방차량이 자차량과 충돌을 피하기 위해 급제동을 해야하며 경우에 따라서는 충돌로 이어지는 위험한 상황이 발생할 수 있다는 것을 예상할 수 있다.

## 8. 결론

2020년 국내에서 발효된 양산 레벨3 자율주행자동차에 대한 안전기준을 살펴보면, 자율주행 도중 운전자가 방향지시등과 같은 수단을 통해 차선변경 의지를 보이는 경우, 차선변경이 자동으로 수행되어야 한다고 명시하고 있다.

본 논문에서는 자율주행 도중 운전자의 지령이나 자율주행 로직에 의해 차선변경 요청이 들어왔을 때, 주변 차량과 자차량의 안전을 동시에 고려하며 안전하게 자동으로 차선을 변경할 수 있는 알고리즘을 제안하였다. 이를 위해 차선변경 상황과 관련된 제어 모드를 ACC 모드, DC 모드, LC 모드로 구분하고 각 모드별 제어를 개발하였다. ACC 모드는 자차량이 현재 차선에서 자율주행을 할 수 있게 해주는 모드이며, DC 모드는 차선변경을 곧바로 수행할 수 없을 때 현재 차선 내에서 차선변경이 가능한 위치로 자차량을 이동시키기 위한 거리제어 모드이다. LC 모드는 차선변경을 수행하기 위한 안전조건이 모두 충족되었을 때 차선변경을 수행하는 모드이다.

본 연구에서는 차선변경 시 자차량과 주변차량의 안전을 고려할 수 있는 차선변경 위험도 판단지표를 개발하였다. 자차량의 절대속도와 자차량과 주변차량들 사이의 상대속도를 함께 반영하였으며, 차선변경에 있어 적극성과 안전성이라는 상반되는 두 가지 목표에 대한 튜닝 파라미터를 도입하였다. 차선변경을 바로 수행하기 불가능한 경우 작동하는 DC 모드에서는 구속화된 최적화 문제를 세우고 이에 대한 최적해를 구하여 차선변경을 시작할 수 있는 현재 차선 내의 좌표를 도출하였다.

자동 차선변경 시스템의 종방향 제어를 위해서, 앞서 언급한 세 가지 모드에 공통적으로 적용할 수 있는 LQR

기반의 최적제어를 설계하였으며, 하지만 세 가지 모드에 대해 상태변수를 다르게 정의하여 각 모드별로 최적화된 종방향 제어로직을 구현하였다. 자동 차선변경 시스템의 횡방향 제어를 위해서는, 경로생성과 경로추종으로 구성된 제어시스템을 구축하였다. 차선변경을 위한 경로생성에 있어서는 3차 다항식의 곡선을 사용하였으며, 차량이 차선변경을 하는 동안의 안전성을 고려함과 동시에 차선변경을 마치고 다시 자율주행을 안전하게 재개할 수 있는 점을 고려하여 다항식의 계수를 설정하였다. 경로추종은 자율주행 조향 제어기법으로 잘 알려진 Pure Pursuit 알고리즘을 사용하였다.

본 논문의 자동 차선변경 시스템의 성능을 검증하기 위해 네 가지의 시나리오를 구성하여, 그 결과를 분석하였다. 네 가지의 시나리오는, 차선변경 요청이 들어왔을 때 곧바로 차선변경을 실행할 수 있는 상황부터 변경할 차선 후방에서 고속으로 차량이 다가와 곧바로 차선을 변경해서는 안 되는 차선변경 상황까지 포함하였다. 본 연구의 자동 차선변경 알고리즘은 네 가지의 시나리오에서 모두 적절한 차선변경 가능 위치에서 안전기준을 만족한 상태로 차선변경을 수행한 것을 확인할 수 있었다. 이와 더불어 차선변경 과정 동안 자차량의 차속 변화가 매우 작게 일어나게 제어함으로써, 탑승객의 승차감을 향상시키고 탑승객이 느낄 수 있는 위험을 최소화하는 것을 확인할 수 있었다.

## 후 기

본 연구는 산업통상자원부 산업핵심기술개발 사업 “자율주행자동차 주변환경 인지 모듈의 안전도 평가기법 및 차량 안전 제어(Fail-operation) 원천기술 개발(10062448)”과제의 지원을 받아 수행되었음.

## References

- 1) MOLIT(Ministry of Land, Infrastructure and Transport), Reported News: Level 3. Autonomous Vehicles Safety Standards Enactment, [http://www.molit.go.kr/USR/NEWS/m\\_71/dtl.jsp?id=95083365](http://www.molit.go.kr/USR/NEWS/m_71/dtl.jsp?id=95083365), 2020.
- 2) T. Y. Oh, W. I. Son, T. W. Ahn and K. H. Park, “A Study on Lane Change Risk Assessment and Path Generation for Autonomous Vehicle Using Artificial Potential Field Method,” KSAE Fall Conference Proceedings, p.742, 2019.
- 3) K. S. Shin, J. H. Choi and K. S. Huh, “A Study on the Automated Lane Change Algorithm Using Vision Sensor and Pseudo-lane,” Transactions of KSAE,

- Vol.27, No.12, pp.925-932, 2019.
- 4) Y. J. Kim, D. J. Ka, S. H. Kim and C. W. Lee, "Development of Integrated Traffic Control of Dynamic Merge and Lane Change at Freeway Work Zones in a Connected and Automated Vehicle Environment," *The Journal of The Korea Institute of Intelligent Transport Systems*, Vol.19, No.3, pp.38-51, 2020.
  - 5) W. B. Na, J. W. Kim and H. C. Lee, "Study on TTC-based Optimal Lane Change Algorithm in Adaptive Cruise Control," *Transactions of KSAE*, Vol.27, No.8, pp.627-636, 2019.
  - 6) J. H. Lee and J. W. Choi, "Deep Reinforcement Learning Based Lane Changing System," *KSAE Spring Conference Proceedings*, pp.821-824, 2019.
  - 7) H. S. Lee, E. S. Park and S. I. Kim, "Deep Reinforcement Learning for Autonomous Vehicle Driving," *KIISE Korea Software Conference Proceedings*, pp.784-786, 2017.
  - 8) J. C. Park, H. S. Chae and K. S. Yi, "Lane Change Driving Analysis Based on Road Driving Data," *Journal of Auto-vehicle Safety Association*, Vol.10, No.1, pp.38-44, 2018.
  - 9) H. S. Chae, Y. H. Jeong, K. C. Min, M. S. Lee and K. S. Yi, "Automated Driving Lane Change Algorithm Based on Robust Model Predictive Control for Merge Situations on Highway Inter-sections," *Transactions of the Korean Society of Mechanical Engineers A*, Vol.41, No.7, pp.575-583, 2017.
  - 10) J. Arora, *Introduction to Optimum Design*, 3rd Edn., Elsevier, New York, 2011.
  - 11) S. Y. Shin, *A Study on Lane Change Path Generation Using Lateral Behavior Prediction Information Based on Model Prediction Control*, M. S. Thesis, Kookmin University, Seoul, 2019.
  - 12) J. J. Wang, Q. C. Zhang, D. B. Zhao and Y. R. Chen, "Lane Change Decision-making through Deep Reinforcement Learning with Rule-based Constraints," *International Joint Conference on Neural Networks*, Budapest, 2019.
  - 13) M. H. Song, C. I. Kim, K. S. Lee, J. M. Kim and M. S. Kim, "Sliding Mode Control with Sliding Observer for Lateral Control of Autonomous Vehicle," *KSAE Spring Conference Proceedings*, pp.708-712, 2017.
  - 14) C. Coulter, *Implementation of the Pure Pursuit Path Tracking Algorithm*, Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, CMU-RI-TR-92-01, 1992.
  - 15) ISO 22179, *Intelligent Transport Systems – Full Speed Range Adaptive Cruise Control(FSRA) Systems – Performance Requirements and Test Procedures*, 2009.
  - 16) I. K. Moon and K. S. Yi, "Human Driver's Driving Pattern Analysis and An Adaptive Cruise Control Strategy," *Transactions of KSAE*, Vol.12, No.4, pp.191-197, 2004.