

# LAB MEETING

국민대학교 지능형 차량 신호 처리 연구실 학부연구생 김지원

2024.11.29(금)



국민대학교  
KOOKMIN UNIVERSITY

# Sliding Model Control(SMC) 횡방향 제어기 구현 및 성능 분석



국민대학교  
KOOKMIN UNIVERSITY

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### Sliding Mode Control(SMC)

1. 비선형 시스템에서 시스템의 동적 특성을 효과적으로 제어하기 위한 강건 제어 기법.
2. 준수한 제어 정확도, 강건성, 쉬운 튜닝과 구현이 특징.
3. 시스템의 상태가 상태 공간 상의 특정 평면(Sliding surface)에서 움직이도록 설계.
4. 크게 Sliding surface 설계 단계와 Control input 설계 단계로 나뉨.

#### Sliding Surface 설계

$S_1 = e(t)$  : 1차 Sliding surface 예시

$S_2 = \dot{e}(t) + \lambda e(t)$  : 2차 Sliding surface 예시

$e(t)$ : 상태 오류

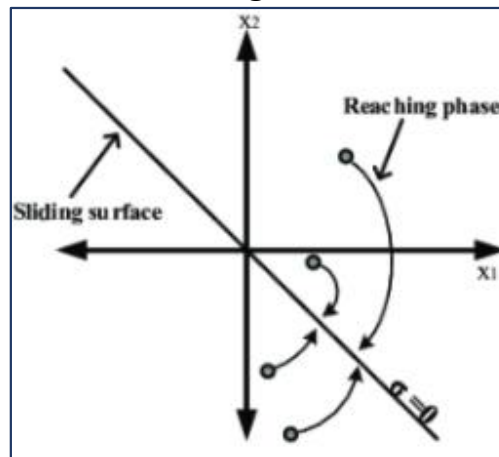
$e(t)_{\text{dot}}$ : 상태 오류 미분

$\lambda$ : Sliding surface 수렴율

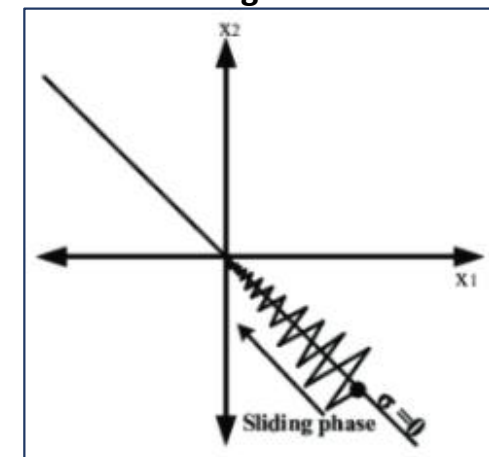
#### Control input 설계

$u(t) = -KS(t)\text{sign}(S(t))$  : 스위칭 제어 입력 방식 예시

Reaching Phase

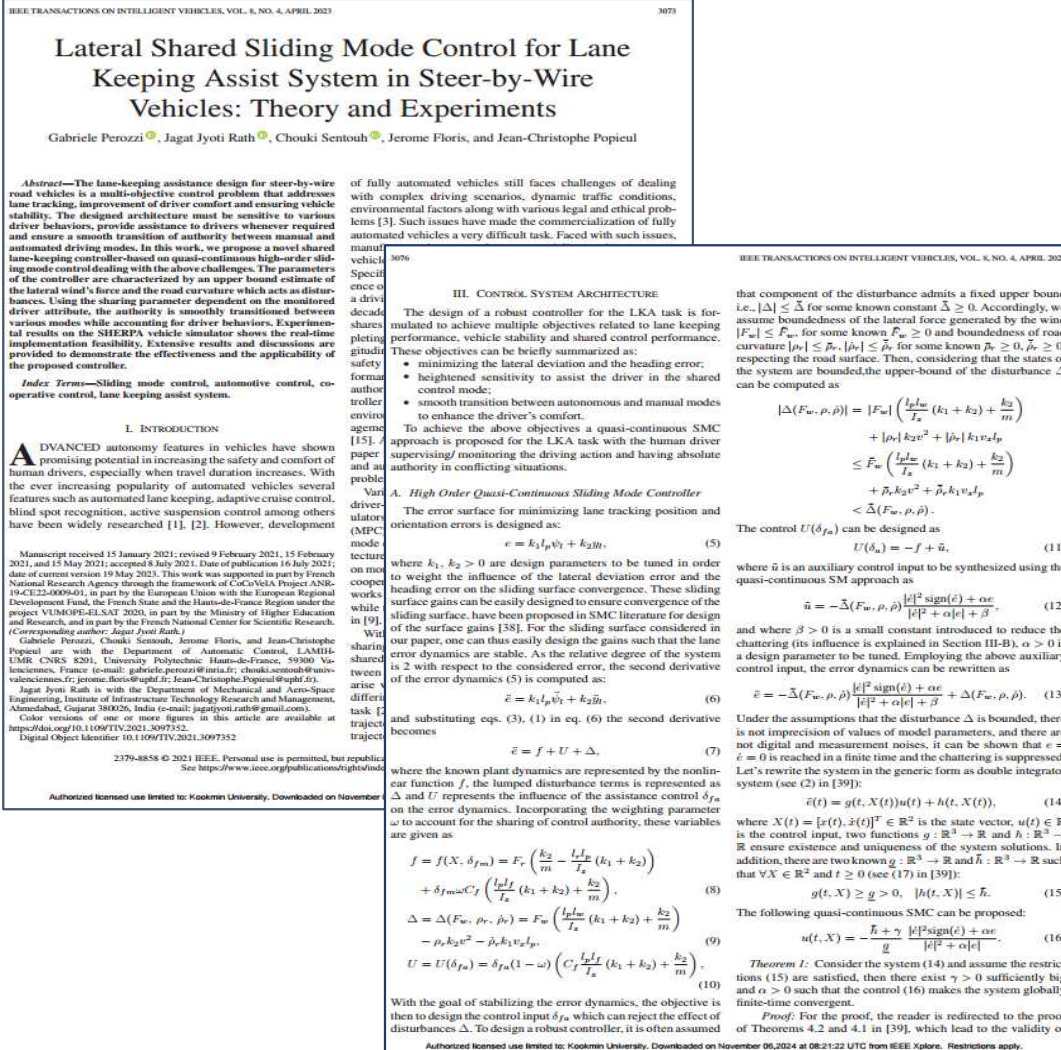


Sliding Phase



## Sliding Model Control(SMC)

## SMC 참고 논문



## 논문 아키텍처

1. 운전자-제어기 간의 매끄러운 권한 전환 및 공유 제어에 초점을 맞춘 아키텍처 제안.
2. 운전자 모델 의존성 제거를 위해 스티어 바이 와이어(steer by wire) 기능을 가정.
3. 시스템은 제어와 관련하여 상대 차수가 2가 됨.
4. 시간 가변 이득을 갖춘 고차 준연속 SMC(HOSMC: Higher Order Quasi Continuous Sliding Mode Control)를 제안.

## 논문 주요 핵심

복잡성, 보수성을 줄이기 위해, 외부 방해 요인의 상한값에 기반하여 계산된 시간 가변 이득을 사용.

-운전자-제어기 간의 권한 관리는 운전자로부터 직접적으로 파생되는 연속적인 공유 매개변수를 통해 보장.



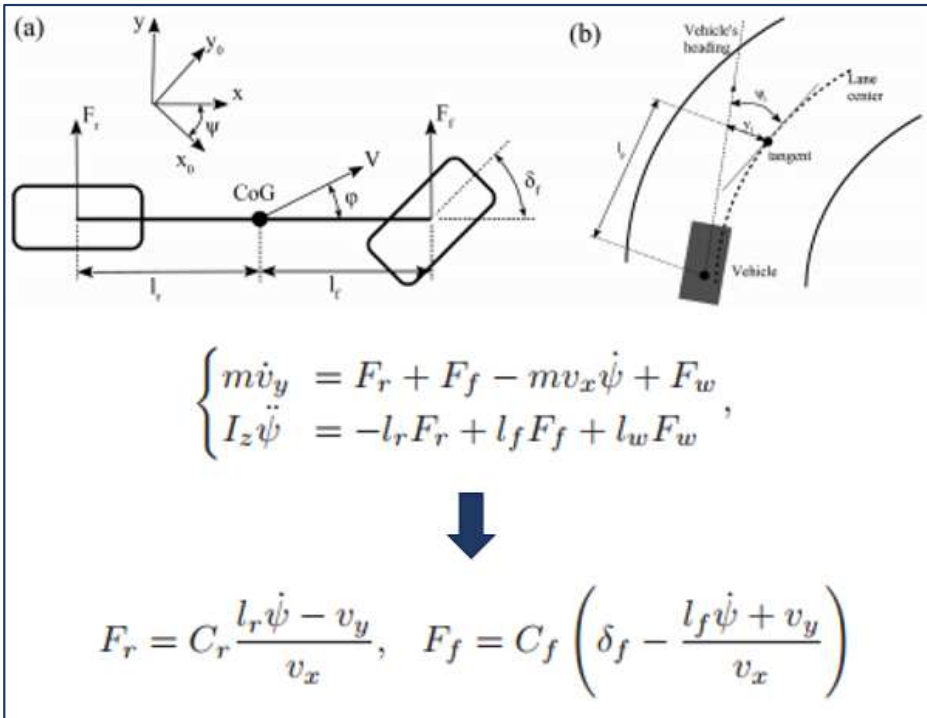
1. 횡방향, 요 각 오차 최소화.
2. 공유 제어 모드에서 운전자를 지원하기 위한 높은 민감도.
3. 자율 모드와 수동 모드 간의 부드러운 전환을 통한 운전자 편안함 향상.

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

## SMC 논문 시스템 수식 정리

## Vehicle Model



$v_y$ : 횡방향 차량 속도  
 $v_x$ : 종방향 차량 속도(일정하다고 가정)  
 $\psi$ : 요(yaw) 각  
 $I_z$ : 요(yaw) 관성 모멘트  
 $l_f$ : 차량 무게중심 ~ 앞바퀴 축 거리  
 $l_r$ : 차량 무게중심 ~ 뒷바퀴 축 거리  
 $l_w$ : 바람이 차량에 영향을 미치는 지점까지 거리  
 $F_w$ : 바람에 의해 발생하는 공기역학적 횡방향 힘

$F_f$ : 앞바퀴에 적용되는 횡방향 힘  
 $F_r$ : 뒷바퀴에 적용되는 횡방향 힘  
 $C_f$ : 앞바퀴 코너링 강성  
 $C_r$ : 뒷바퀴 코너링 강성  
 $\delta_f$ : 타이어 조향각

## Vehicle-Road Model

$$\begin{cases} \dot{\psi}_l = \dot{\psi} - \rho_r v_x \\ \dot{y}_l = v_y + l_p \dot{\psi} + \psi_l v_x \end{cases}$$

$y_l$ : 횡방향 오차  
 $\psi_l$ : 요(yaw) 각 오차  
 $\rho_r$ : 도로 곡률  
 $l_p$ : 차량의 무게중심에서 전방을 바라보는 거리

## Driver-Vehicle-Road Interaction

$$\delta_f = (1 - \omega)\delta_{fa} + \omega\delta_{fm} \quad \delta_{fm} = \frac{\delta_d}{R_s}$$

$\delta_f$ : 최종 출력 조향각  
 $\delta_{fa}$ : 제어 시스템 입력 조향각  
 $\delta_{fm}$ : 최종 운전자 조향각  
 $w$ : 운전자-제어기 간의 조향 가중 매개변수  
 $\delta_d$ : 운전자가 스티어링 휠에 입력한 조향각  
 $R_s$ : 운전자 조향 감소 비율

$w = 1$ :  $\delta_f = \delta_{fm}$ 인 완전 수동 모드 상황  
 $w = 0$ :  $\delta_f = \delta_{fa}$ 인 완전 자율 제어 모드 상황

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### SMC 논문 제어기 수식 정리

##### 고차 준연속 SMC 기반 오차면

$$e = k_1 l_p \psi_l + k_2 y_l \quad \ddot{e} = k_1 l_p \ddot{\psi}_l + k_2 \ddot{y}_l$$

$k_1, k_2 > 0$ : 시스템 상태의 오차면 수렴성을 보장하는 매개변수

##### 묶음 외란 항(바람, 곡률 고려) $\Delta$

$$\Delta = \Delta(F_w, \rho_r, \dot{\rho}_r) = F_w \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) - \rho_r k_2 v^2 - \dot{\rho}_r k_1 v_x l_p$$

##### 강건 제어기 설계를 위한 외란 상한(바람, 곡률 고려) $\bar{\Delta}$

$$\begin{aligned} |\Delta(F_w, \rho, \dot{\rho})| &= |F_w| \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) \\ &\quad + |\rho_r| k_2 v^2 + |\dot{\rho}_r| k_1 v_x l_p \\ &\leq \bar{F}_w \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) \\ &\quad + \bar{\rho}_r k_2 v^2 + \bar{\dot{\rho}}_r k_1 v_x l_p \\ &< \bar{\Delta}(F_w, \rho, \dot{\rho}). \end{aligned}$$

##### Plant 동역학 기반 비선형 함수 $f$ (기본 제어 입력)

$$f = f(X, \delta_{fm}) = F_r \left( \frac{k_2}{m} - \frac{l_r l_p}{I_z} (k_1 + k_2) \right) + \delta_{fm} \omega C_f \left( \frac{l_p l_f}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right)$$

##### Driver-Vehicle-Road Interaction

$$U = U(\delta_{fa}) = \delta_{fa} (1 - \omega) \left( C_f \frac{l_p l_f}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right)$$

##### 준연속 SMC 접근법에 의해 합성되는 보조 제어 입력 $\tilde{u}$

$$\tilde{u} = -\Delta(\bar{F}_w, \rho, \dot{\rho}) \frac{|\dot{e}|}{2} \text{sign}(\dot{e}) + \alpha e \frac{|\dot{e}|}{2} + \alpha |e| + \beta,$$

$\beta > 0$ : 체터링을 줄이기 위해 도입되는 양수 파라미터  
 $\alpha > 0$ : 보조 제어 입력 조정을 위한 설계 파라미터

##### 최종 제어 입력 $u$

$$U(\delta_{fa}) = -f + \tilde{u}$$



## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

## 바람 영향 제거 및 완전 자율 모드화 수식

$w = 0 \rightarrow$  완전 자율 제어 모드 적용

$\delta_{fm} = 0 \rightarrow$  운전자 조향 입력 제거( $\delta f = \delta_{fm}$ )

$F_w = 0 \rightarrow$  바람의 영향 및 공기역학적 횡방향 힘 제거

$l_f = 0 \rightarrow$  바람이 차량에 영향을 미치는 거리 제거

꺾임 외란 항  $\Delta$ 

$$\Delta = \Delta(F_w, \rho, \dot{\rho}) = F_w \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) + \rho_r k_2 v^2 + \dot{\rho}_r k_1 v_x l_p$$

## 강건 제어기 설계를 위한 외란 상한(바람, 곡률 고려)

$$\begin{aligned} |\Delta(F_w, \rho, \dot{\rho})| &= |F_w| \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) \\ &\quad + |\rho_r| k_2 v^2 + |\dot{\rho}_r| k_1 v_x l_p \\ &\leq \bar{F}_w \left( \frac{l_p l_w}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right) \\ &\quad + \bar{\rho}_r k_2 v^2 + \bar{\dot{\rho}}_r k_1 v_x l_p \\ &< \bar{\Delta}(F_w, \rho, \dot{\rho}) \end{aligned}$$

곡률을 고려한 외란 보상을 위한 상한 항  
(강건 제어를 위해 상한값 사용)

## 고차 준연속 SMC 기반 오차면

$$e = k_1 l_p \psi_l + k_2 y_l \quad \ddot{e} = k_1 l_p \ddot{\psi}_l + k_2 \ddot{y}_l$$

## 준연속 SMC 접근법에 의해 합성되는 보조 제어 입력

$$\tilde{u} = -\Delta(F_w, \rho, \dot{\rho}) - \frac{|\dot{e}|}{2} \text{sign}(\dot{e}) + \alpha e \frac{|\dot{e}|}{2} + \alpha |e| + \beta,$$

시스템 상태를 오차면에 수렴시키는 오차 기반 제어 항

Plant 동역학 기반 비선형 함수  $f$  (기본 제어 입력)

$$f = f(X, \delta_{fm}) = F_r \left( \frac{k_2}{m} - \frac{l_r l_p}{I_z} (k_1 + k_2) \right) + \delta_{fm} \omega C_f \left( \frac{l_p l_f}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right)$$

차량(플랜트) 동역학 기반 제어 항

## Driver-Vehicle-Road Interaction

$$U = U(\delta_{fa}) = \delta_{fa} (1 - \omega) \left( C_f \frac{l_p l_f}{I_z} (k_1 + k_2) + \frac{k_2}{m} \right)$$

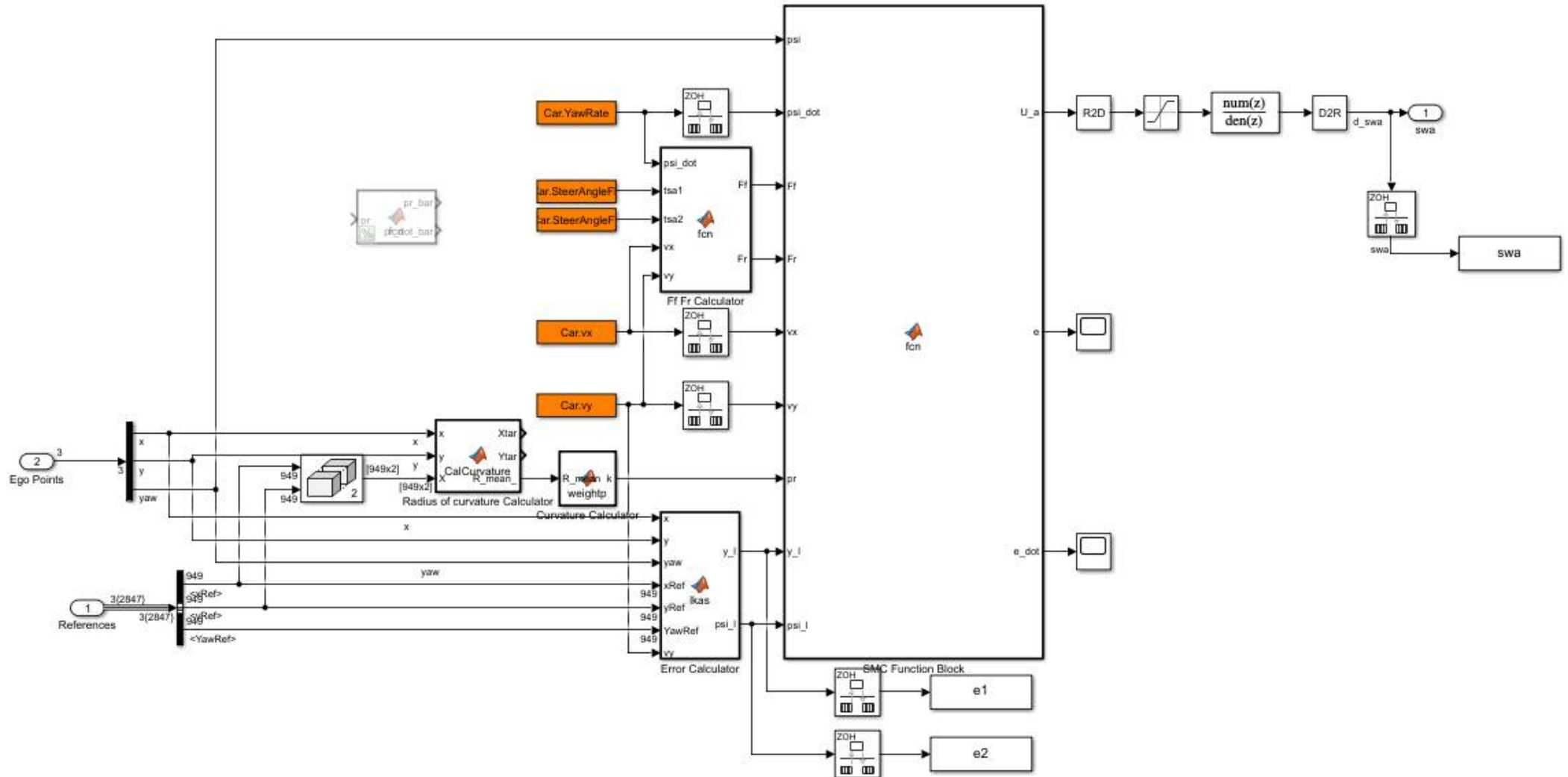
최종 제어 입력  $u$ 

$$U(\delta_{fa}) = -f + \tilde{u}$$

# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

### SMC Simulink 구조도





## Sliding Model Control(SMC)

## SMC Function Block

```

function [U_a, e, e_dot] = fcn(psi, psi_dot, Ff, Fr, vx, vy, pr, y_l, psi_l)
% Ff: 앞바퀴에 적용된 횡방향 힘
% Fr: 뒷바퀴에 적용된 횡방향 힘
% psi_l: 차량 heading 오차
% y_l: 횡방향 편차 오차
% psi 요 각도(yaw)
% pr: 도로 곡률

% lw: 바람이 차량에 영향을 미치는 지점까지의 거리(고려 x)
% Fw: 바람에 의해 발생하는 횡방향 공력 (고려 x)
% w: %운전자 개입 정도(0~1) (고려 x)

lp = 1/(1+12*pr); % 차량 무게 중심에서 앞을 바라보는 거리
m = 2044.2; % 차량 질량
Iz = 3558.1; % 요 관성 모멘트
lf = 1.314; % 앞축 거리
lr = 1.786; % 뒷축 거리
Cf = 110000; % 앞바퀴 코너링 강성
Cr = 98000; % 뒷바퀴 코너링 강성

pr_bar = 0.0148; % 도로 곡률 최대값(외란 상한 파라미터)
pr_dot_bar = 0.1816; % 도로 곡률 변화율 최대값(외란 상한 파라미터)

% k1,k2>0는 횡방향 편차 오차와 heading 오차가 슬라이딩 면의 수렴에
% 미치는 영향을 가중하기 위해 조정할 설계 매개변수
k1 = 1.6;
k2 = 1.6;

psi_l_dot = psi_dot - pr*vx; % psi_l 변화율
y_l_dot = vy + lp*psi_dot + psi_l*vx; % y_l 변화율

% 오차면 설계 공식
e = k1*lp*psi_l + k2*y_l;
e_dot = k1*lp*psi_l_dot + k2*y_l_dot;

% f: 플랜트 동역학을 표현하는 비선형 함수
f = Fr*(k2/m - (lr*lp)/Iz*(k1 + k2)); %+ delta_fm*w*Cf*((lp*lf)/(Iz*(k1 + k2)) + k2/m);

alpha = 5.2; % 조정할 설계 파라미터(alpha > 0)
beta = 5.2; % 채터링을 줄이기 위해 도입된 양수 파라미터(beta > 0)

% trifpp_bar: 방해항 상한
trifpp_bar = abs(pr_bar)*k2*sqrt(vx^2 + vy^2)^2 + abs(pr_dot_bar)*k1*vx*lp;

% u: 준연속 SM 접근법을 사용하여 합성할 보조 제어 입력
u = -trifpp_bar*((abs(e_dot)^2)*sign(e_dot) + (alpha*e))/(abs(e_dot)^2 + alpha*abs(e)+beta);

% 오차 동역학을 안정화하는 것을 목표로 방해항 영향을 배제할 수 있는 제어 입력
U_a = -f + u;

```

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

## Ff, Fr Calculator

```
function [Ff, Fr] = fcn(psi_dot, tsa1, tsa2, vx, vy)
```

```
%tsa: Tire Steer Angle
lf = 1.314; % 앞축 거리
lr = 1.786; % 뒷축 거리
Cf = 110000; % 앞바퀴 코너링 강성
Cr = 98000; % 뒷바퀴 코너링 강성
tsa = (tsa1 + tsa2)/2;
```

$$F_r = C_r \frac{l_r \dot{\psi} - v_y}{v_x}, \quad F_f = C_f \left( \delta_f - \frac{l_f \dot{\psi} + v_y}{v_x} \right)$$

```
Ff = Cf*(tsa - (lf*psi_dot + vy)/vx); % 앞바퀴에 적용된 횡방향 힘
Fr = Cr*(lr*psi_dot - vy)/vx; % 뒷바퀴에 적용된 횡방향 힘
```

## Error Calculator

```
function [y_l, psi_l] = lkas(x, y, yaw, xRef, yRef, YawRef, vx)
```

```
% 가장 가까운 참조 지점을 찾기
distances = sqrt((xRef - x).^2 + (yRef - y).^2);
[~, minIndex] = min(distances);

% 가장 가까운 참조 지점 선택
xRefValue = xRef(minIndex);
yRefValue = yRef(minIndex);
YawRefValue = YawRef(minIndex);

% lateral position error 계산
dx = xRefValue - x;
dy = yRefValue - y;
y_l = dx * sin(YawRefValue) - dy * cos(YawRefValue);
% e1 = sqrt((dx).^2 + (dy).^2);

% lateral position error 변화량 계산
% y_l_dot = vy + 60 * normalizeAngle(yaw - YawRefValue);

% yaw angle error 계산
psi_l = normalizeAngle(yaw - YawRefValue);

% 각도 정규화 함수 (lkas 함수 내에 중첩됨)
function angle = normalizeAngle(angle)
% 각도를 [-pi, pi] 범위로 정규화
angle = atan2(sin(angle), cos(angle));
end
end
```

## Curvature Calculator

```
function [Xtar, Ytar, R_mean_] = CalCurvature(x,y,X)
```

```
%% Tuning Parameters
BACK_num=0;
tar_num=50;
```

```
tempX=zeros(1,tar_num);
tempY=zeros(1,tar_num);
Xm=X(:,1);
Ym=X(:,2);
```

```
%% Path Logic
Xm_first=Xm';
Ym_first=Ym';
endPoint_num = 50;
```

```
dist=sqrt((Xm_first-x
```

```
StartPoint=index-BACK
```

```
endPoint=index-BACK_n
```

```
if StartPoint<1, Star
```

```
if endPoint>length(Xm
```

```
tempX(1,1:endPoint-St
```

```
tempY(1,1:endPoint-St
```

```
Xtar=tempX;
Ytar=tempY;
```

```
%% Curvature Cal
path_L = 50;
Xtar = Xtar(1:path_L)
```

```
Ytar = Ytar(1:path_L)
```

```
XY = [Xtar', Ytar'];
```

```
N = size(XY, 1);
dims = size(XY, 2);
```

```
if dims == 2
XY2 = [XY, zeros(N, 1
```

```
end
R = zeros(path_L, 1);
```

```
R_valid = zeros(path_L-3,
```

```
ii = 0;
```

```
for i = 3:path_L-3
ii = ii+1;
```

```
[R(i), ~] = circumcen
```

```
R_valid(ii) = R(i);
```

```
end
R_valid(R_valid > 600) =
```

```
len=sum(Xtar~=0)-3;
```

```
if len < 1
len = 0;
```

```
end
R_valid(len+1:end,1)=600;
```

```
R_mean = zeros(length(R_v
```

```
for id = 1:length(R_valid
```

```
R_mean(id) = mean(R_v
```

```
end
R_mean_=mean(R_mean);
```

```
end
```

```
function [R1, k1] = circumcenter(A1, B1, C1)
```

```
D1 = cross(B1-A1, C1-A1);
```

```
b1 = norm(A1-C1);
```

```
c1 = norm(A1-B1);
```

```
if nargin == 1
```

```
a1 = norm(B1-C1);
```

```
R1 = a1 * b1 * c1 / 2 / norm(D1);
```

```
if norm(D1) == 0
```

```
R1 = Inf;
```

```
end
```

```
return
```

```
end
```

```
E1 = cross(D1, B1-A1);
```

```
F1 = cross(D1, C1-A1);
```

```
G1 = (b1^2 * E1 - c1^2 * F1) / norm(D1)^2 / 2;
```

```
R1 = norm(G1);
```

```
if R1 == 0
```

```
k1 = G1;
```

```
elseif norm(D1) == 0
```

```
R1 = Inf;
```

```
k1 = D1;
```

```
else
```

```
k1 = G1' / R1^2;
```

```
end
```

```
% Radius Limit 추가
```

```
if abs(R1) >= 1e10 %1e10
```

```
R1 = sign(R1)*1e10; %1e10
```

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### SMC Parameter 설정

##### ● $l_p$ : 차량 무게 중심에서 앞을 바라보는 거리

- 해당 곡률 시나리오 기준  $l_p > 1.6m$ 부터는 경로 이탈 발생.
- 따라서 곡률( $pr$ )을 고려한 수식으로 정의  $\rightarrow l_p = 1/(1 + 12*pr)$

##### ● $k_1, k_2 > 0$ : 시스템 상태의 오차면 수렴성을 보장하는 매개변수

- 값이 크면 Over steering 발생 및 경로 이탈.
- 값이 작으면 Under steering 발생 및 경로 이탈.
- 두 값 모두 적절한 값인 1.6으로 설정.

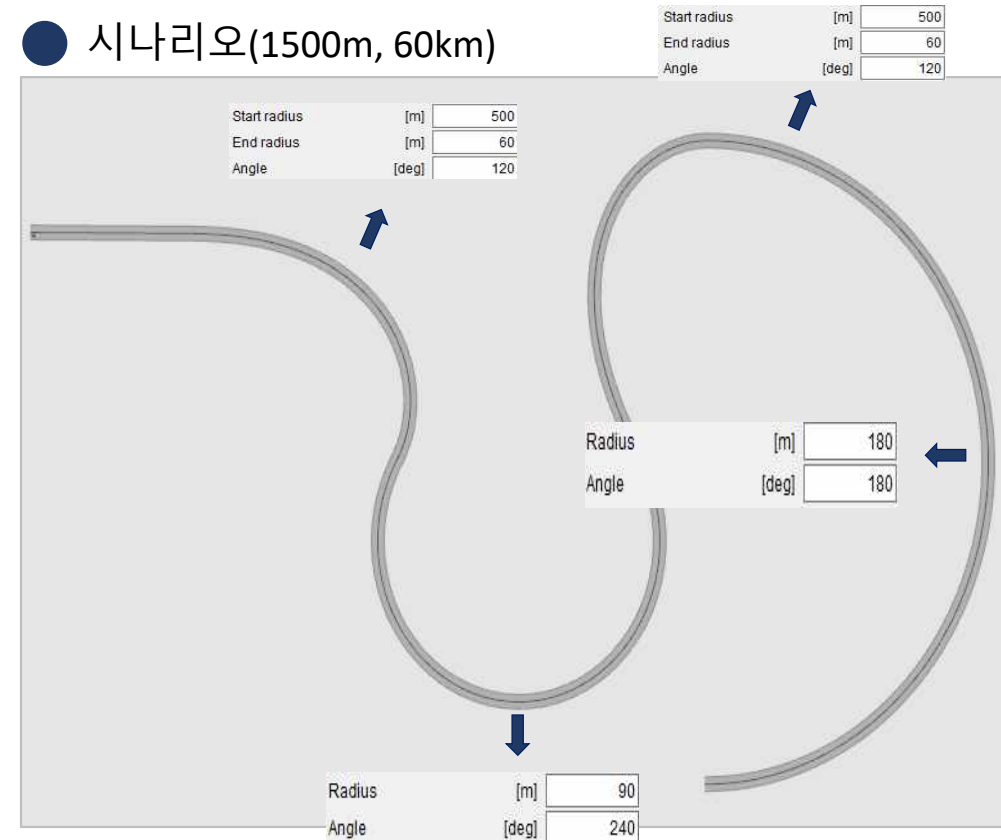
##### ● $\beta > 0$ : 체터링을 줄이기 위해 도입되는 양수 파라미터

##### ● $\alpha > 0$ : 보조 제어 입력 조정을 위한 설계 파라미터

- $\alpha$ 가 작으면  $e_1, e_2$  크게 증가
- $\alpha$ 가 크면  $e_1$ 은 감소하지만  $e_2$  증가, jerk 크게 증가
- $\beta$ 가 작으면 steering wheel 크게 떨림
- $\beta$ 가 크면  $e_1, e_2$  증가
- 두 값 모두 적절한 값인 5.2로 설정.

#### Scenario / Road

##### ● 시나리오(1500m, 60km)

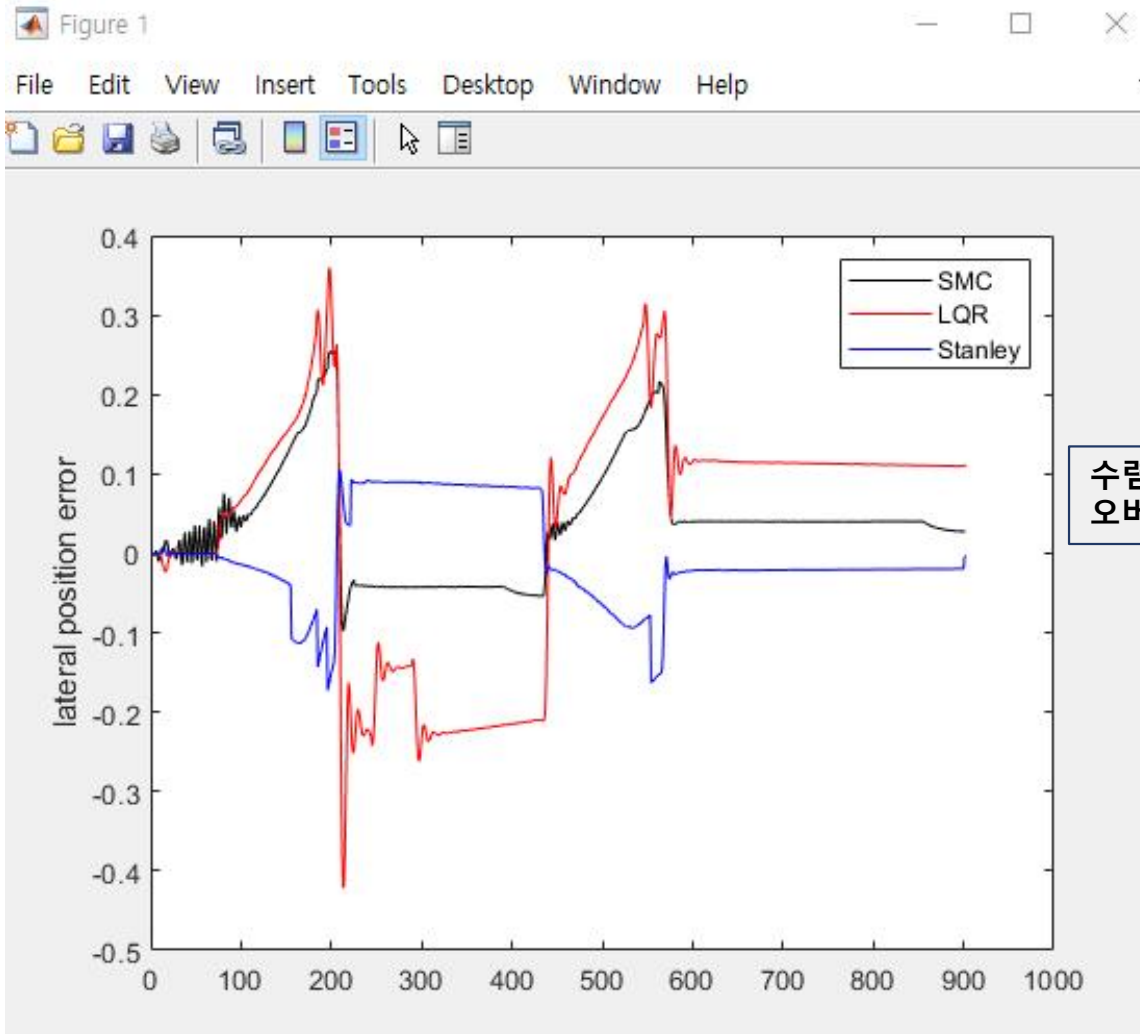


## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

## ■ 횡방향 제어기 성능 비교 분석

## ● e1(lateral position error) 비교분석



# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

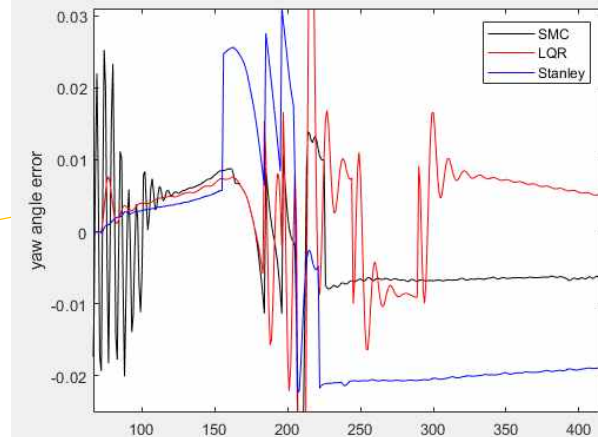
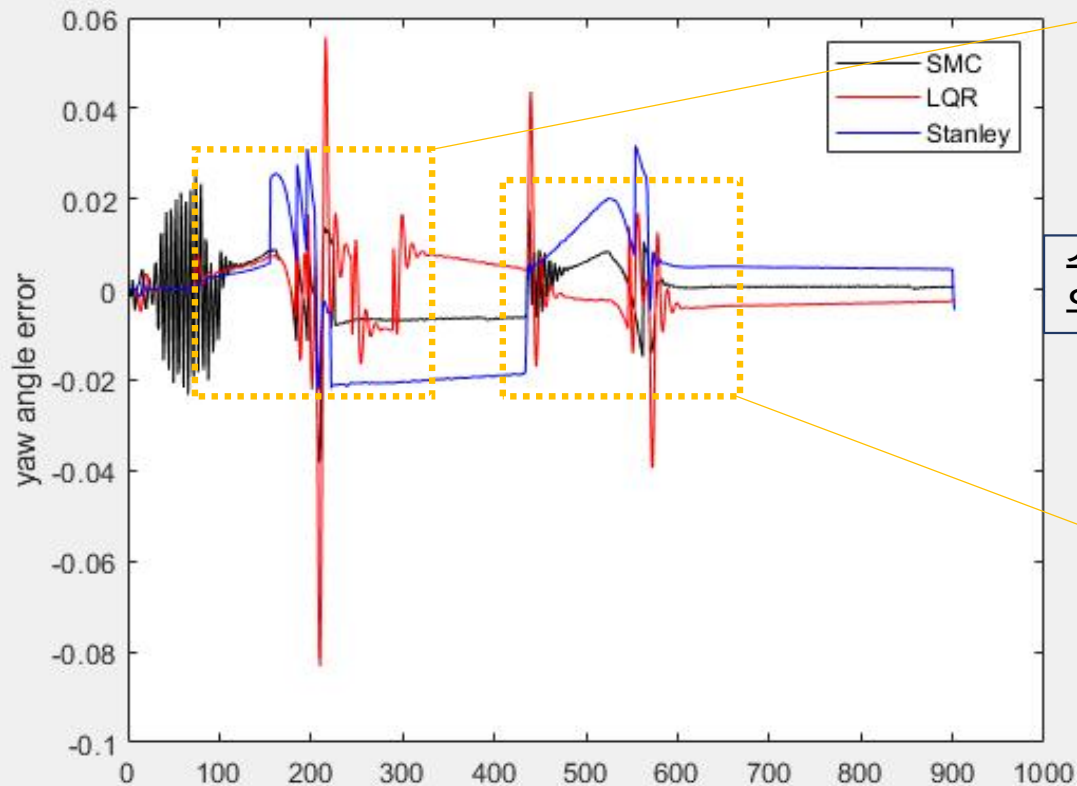
## Sliding Model Control(SMC)

### 횡방향 제어기 성능 비교 분석

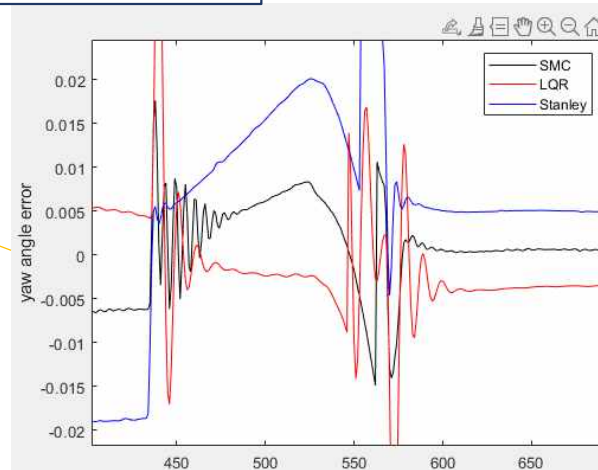
#### e2(yaw angle error) 비교분석

Figure 2

File Edit View Insert Tools Desktop Window Help



수렴율: SMC > LQR > Stanley  
오버슈트: SMC > Stanley > LQR



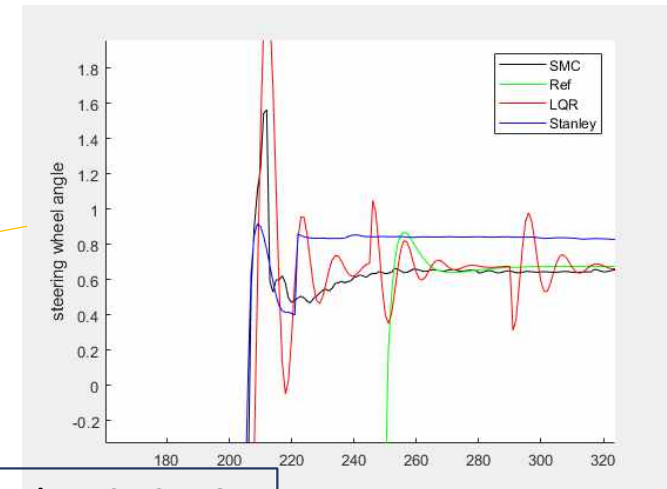
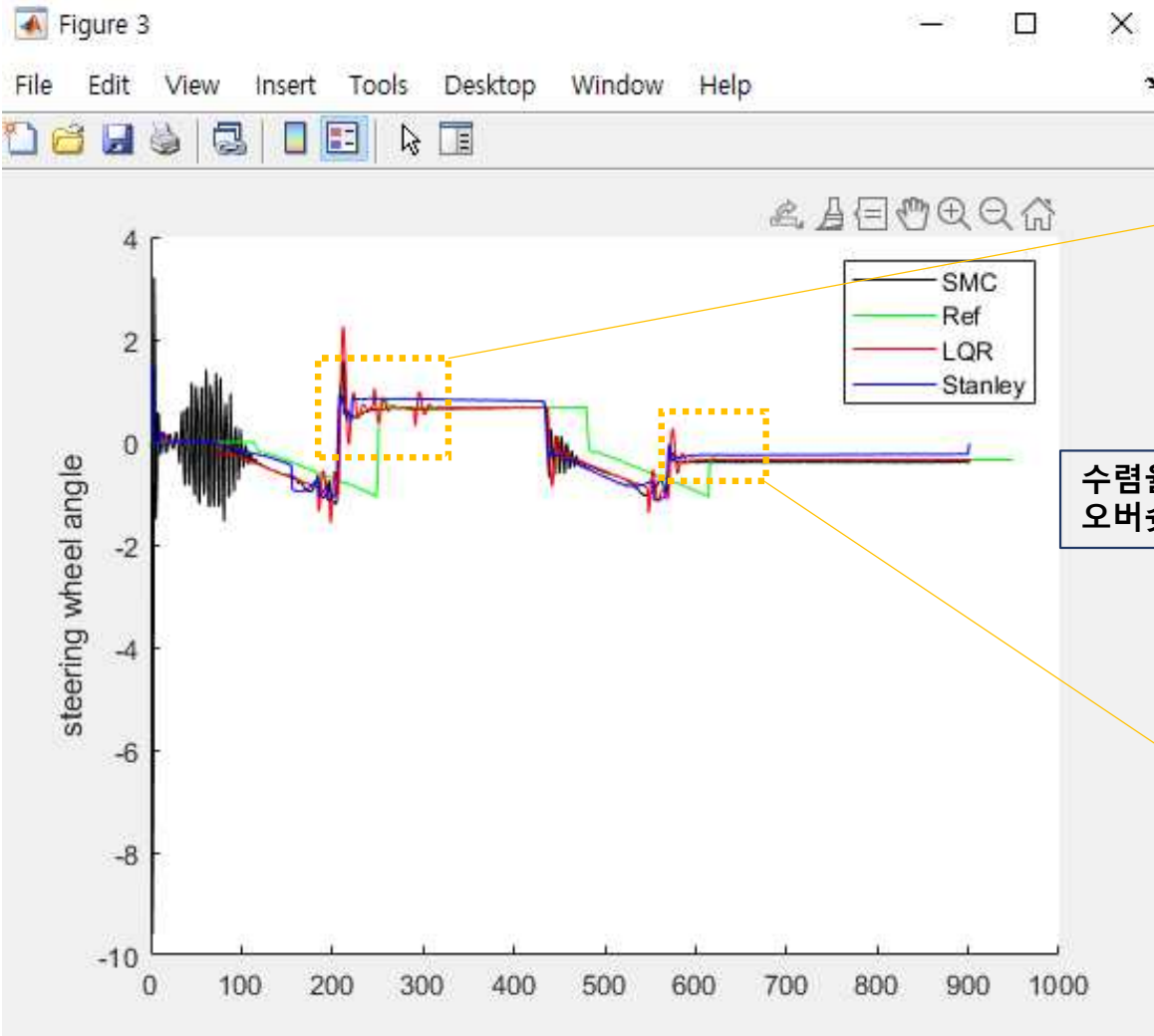


# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

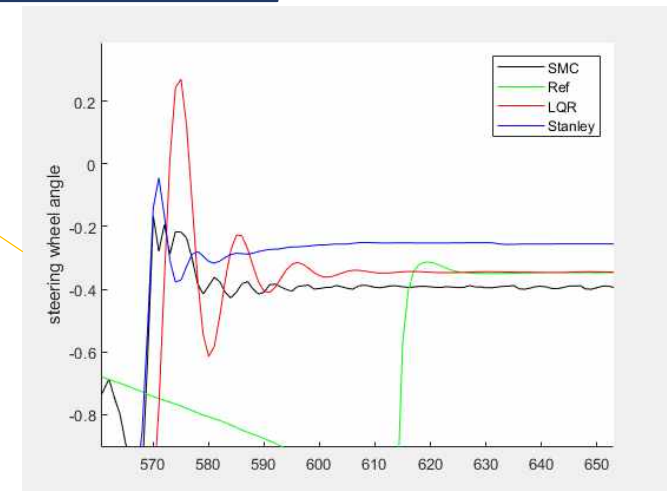
## Sliding Model Control(SMC)

### 횡방향 제어기 성능 비교 분석

#### swa(steering wheel angle) 비교분석



수렴율: Stanley > SMC > LQR  
오버슈트: Stanley > SMC > LQR





## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### ■ 횡방향 제어기 성능 비교 분석

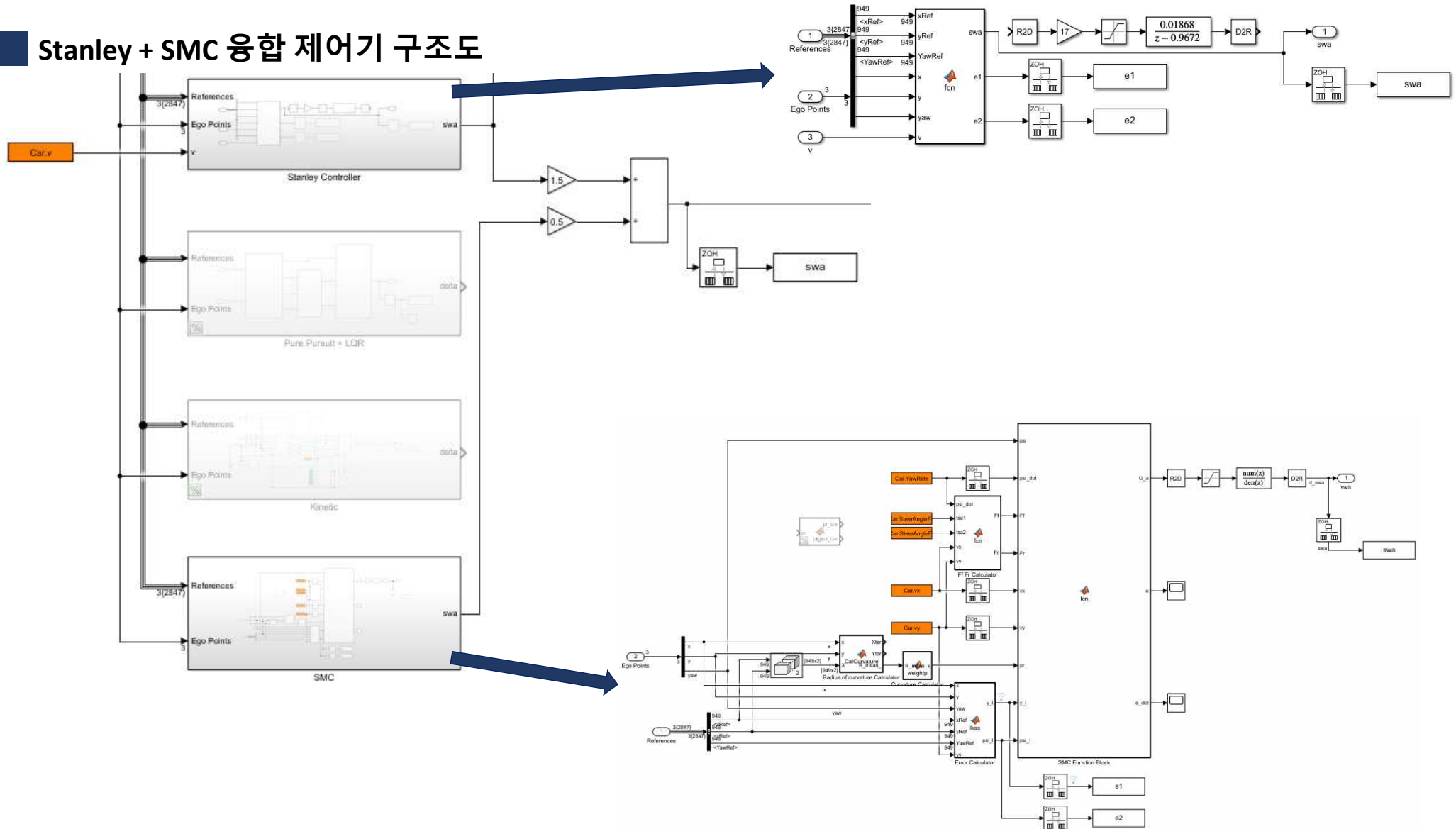
##### ● 횡방향 제어기 성능 비교 분석

	e1(max)	e1(mean)	e2(max)	e2(mean)	SWA(max)	SWA(mean)
SMC	0.2553	0.0791	0.0381	0.0066	9.5859	0.6006
LQR	0.4213	0.1610	0.0832	0.0089	3.1142	0.3988
Stanely	0.1728	0.0609	0.0318	0.0130	1.8982	0.3918

1. LQR은 비선형 시스템을 선형화하여 제어하기때문에 제일 성능이 낮음.
2. LQR은 코너링 강성이 평균값과 큰 차이를 보이는 곡선 시나리오에서 성능이 더 떨어짐.
3. SMC는 차량동역학 기반 비선형 강건 제어 기법이기에 LQR보다 성능이 좋음.
4. SMC의 직선구간 털림 현상을 해결하면 Stanley보다 e1, e2부분에서 높은 성능 예상.

## Sliding Model Control(SMC)

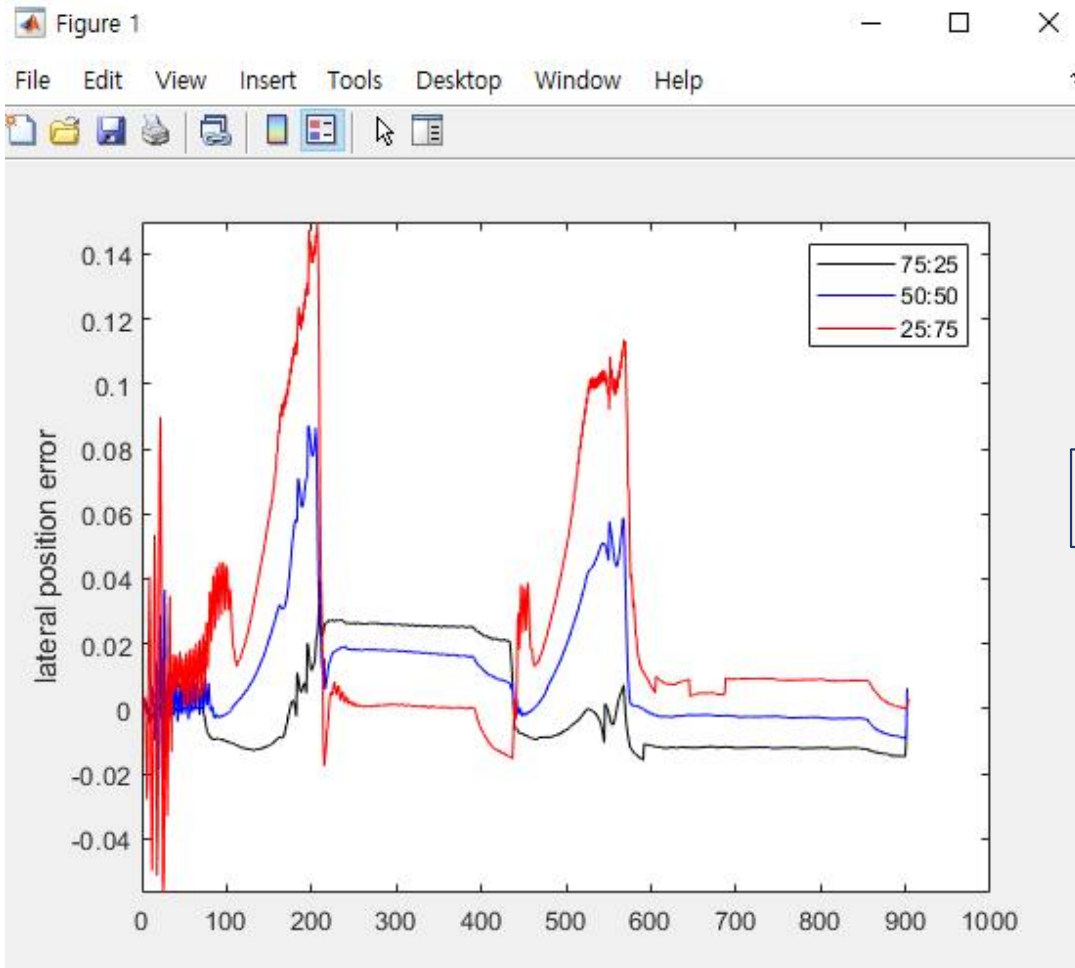
## Stanley + SMC 융합 제어기 구조도



## Sliding Model Control(SMC)

## Stanley + SMC 융합 비율에 따른 성능 비교 분석

## e1(lateral position error) 비교분석



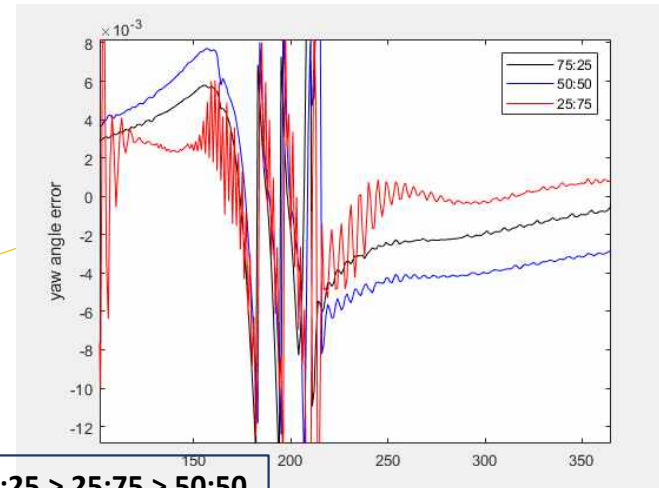
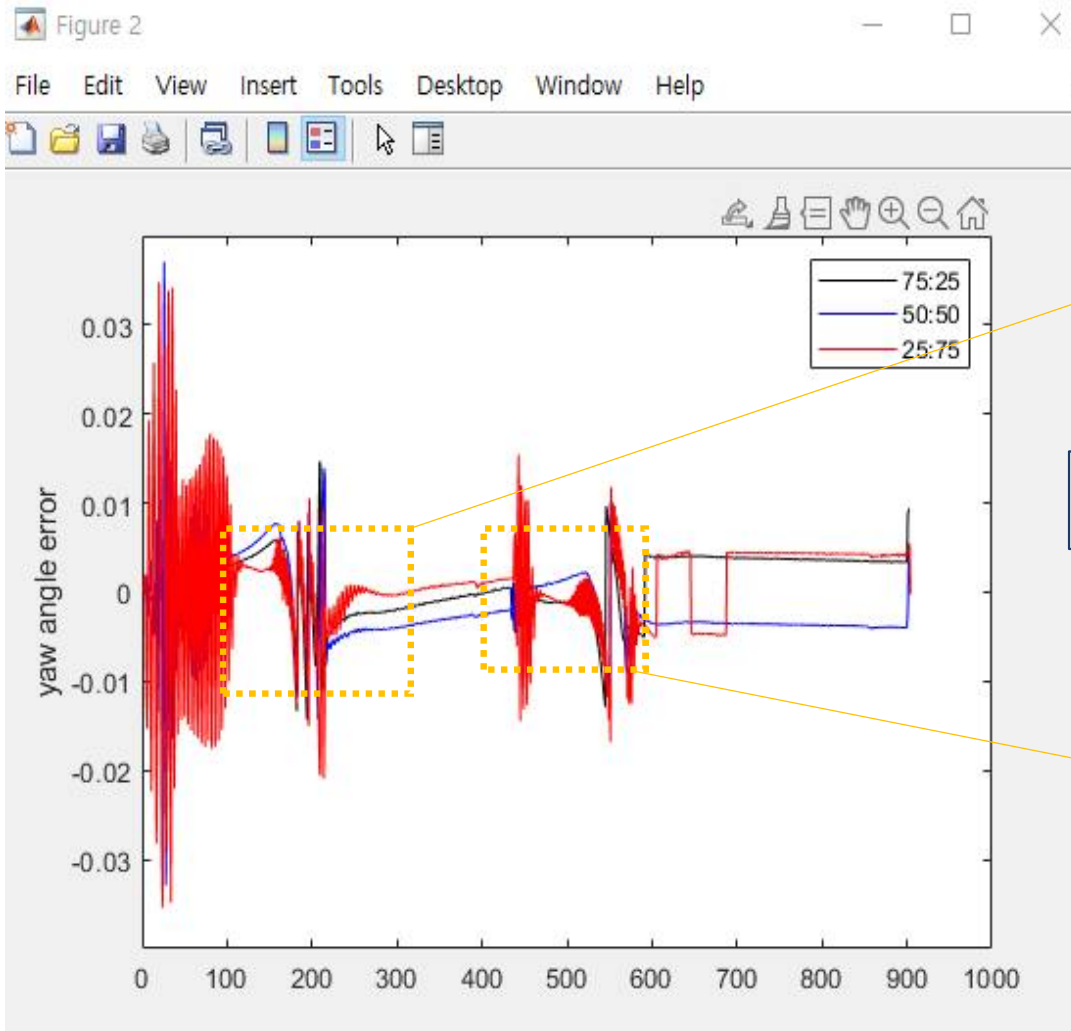
수렴율: 50:50 > 75:25 > 25:75  
오버슈트: 75:25 > 50:50 > 25:75

# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

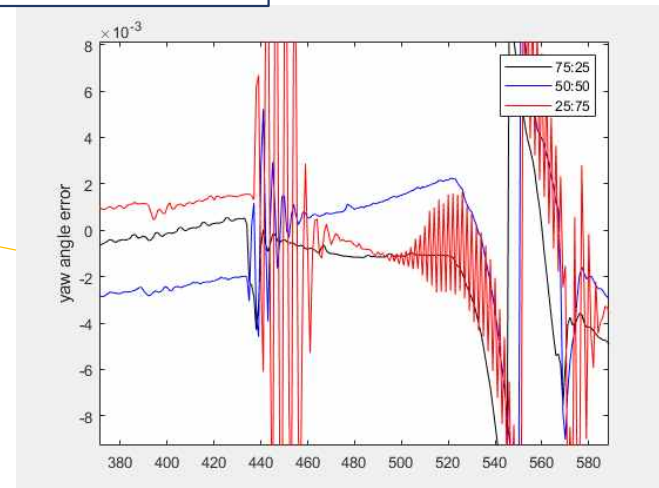
## Sliding Model Control(SMC)

### Stanley + SMC 융합 비율에 따른 성능 비교 분석

#### e2(yaw angle error) 비교분석



수렴율: 75:25 > 25:75 > 50:50  
오버슈트: 75:25 > 50:50 > 25:75

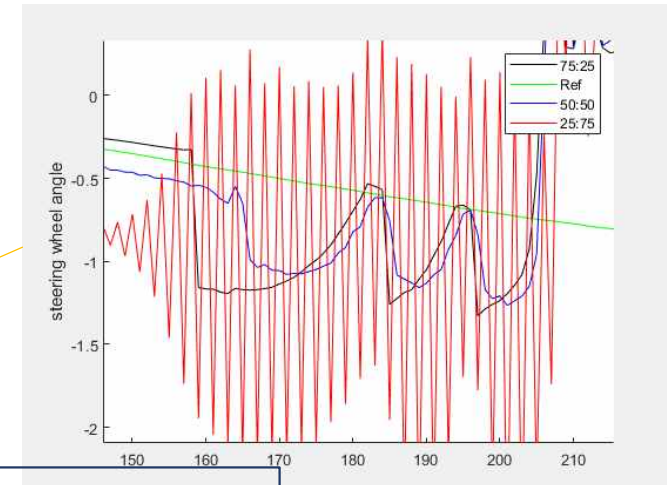
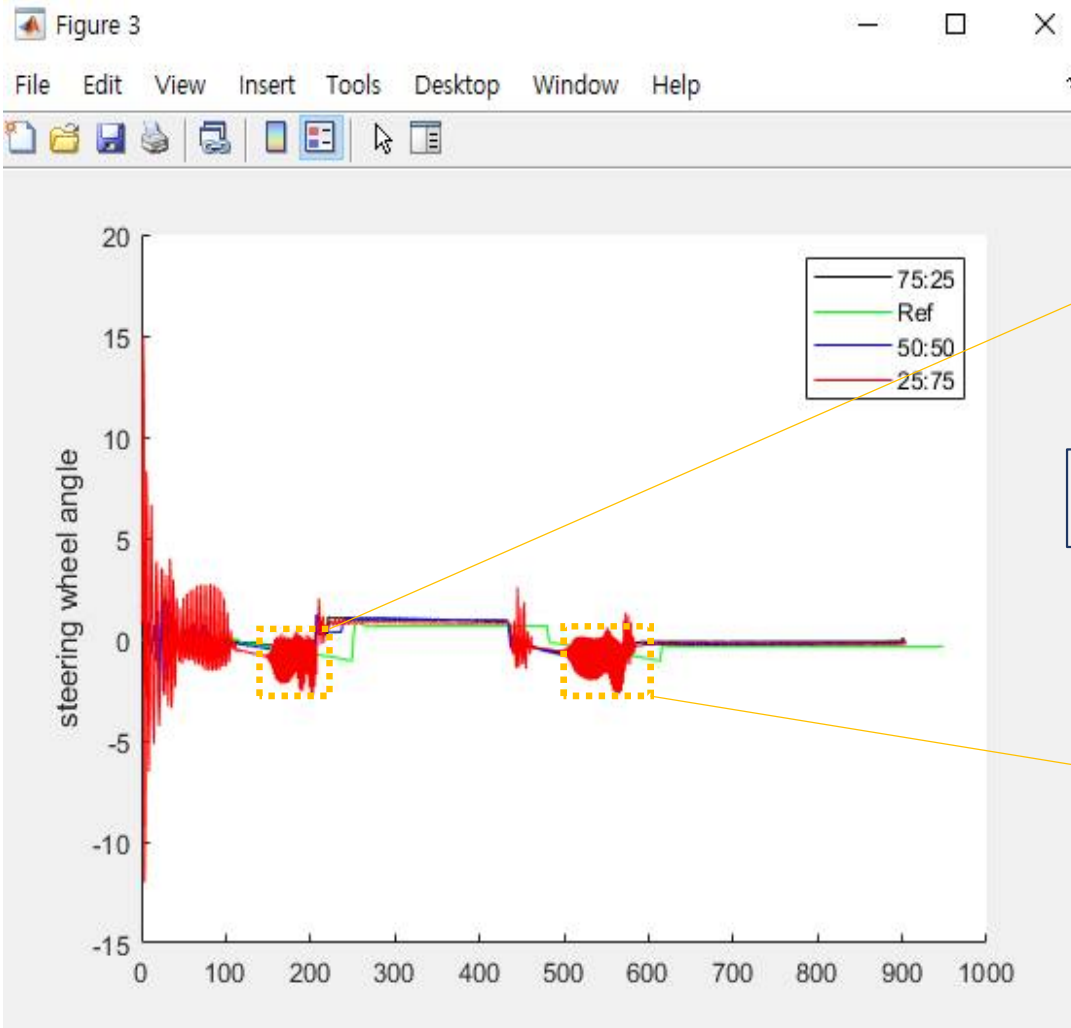


# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

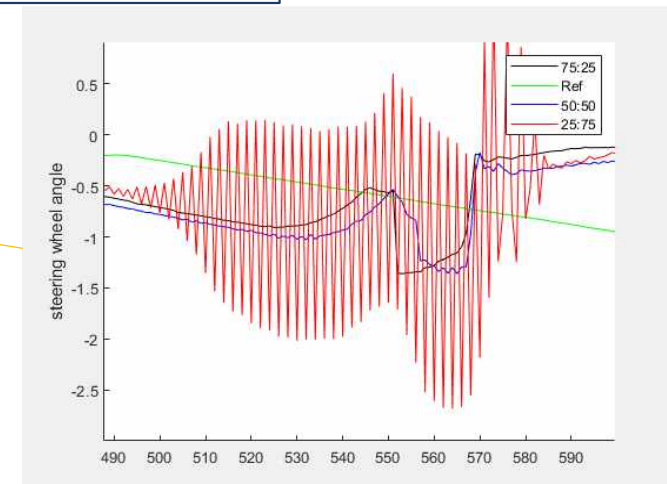
## Sliding Model Control(SMC)

### Stanley + SMC 융합 비율에 따른 성능 비교 분석

#### swa(steering wheel angle) 비교분석



수렴율: 75:25 > 50:50 > 25:75  
오버슈트: 50:50 > 75:25 > 25:75



Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

Sliding Model Control(SMC)

Stanley + SMC 융합 비율에 따른 성능 비교 분석

Stanley + SMC 융합 비율에 따른 성능 비교 분석

S: SMC	e1(max)	e1(mean)	e2(max)	e2(mean)	SWA(max)	SWA(mean)
75: 25	0.0395	0.0155	0.0147	0.0035	6.6176	0.5053
50: 50	0.0872	0.0209	0.0371	0.0050	10.8501	0.6231
25: 75	0.1499	0.0420	0.0354	0.0067	15.0216	1.3040

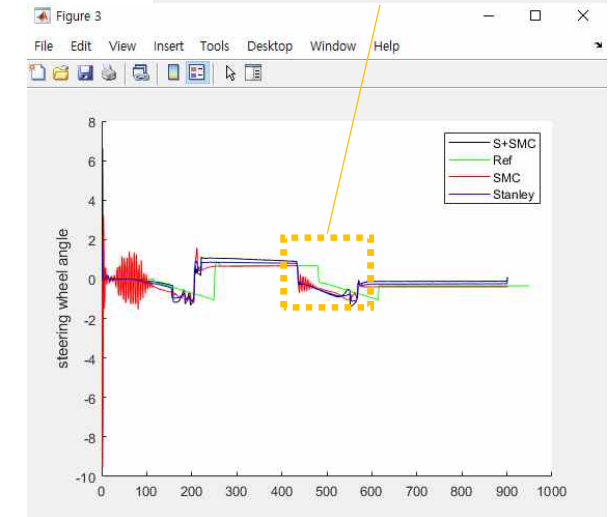
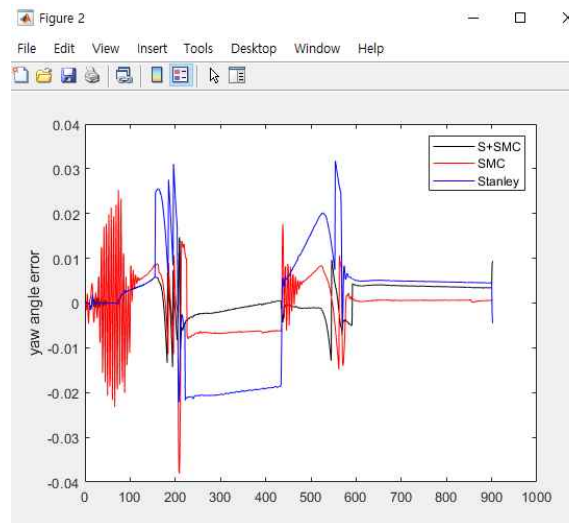
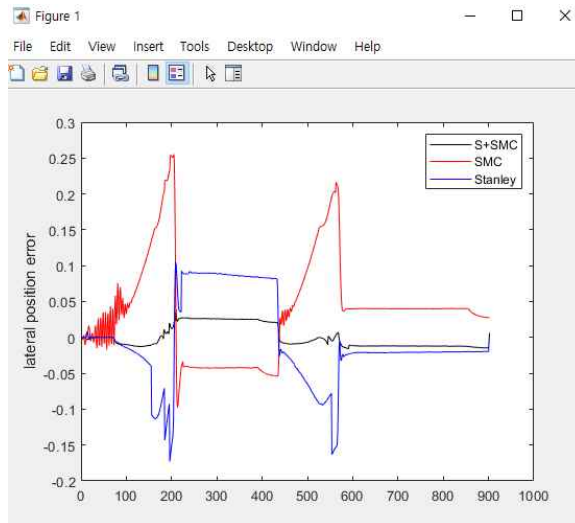


# Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

## Sliding Model Control(SMC)

### Stanley + SMC 및 단독 제어기 성능 비교 분석

### Stanley + SMC 및 단독 제어기 성능 비교 분석



	e1(max)	e1(mean)	e2(max)	e2(mean)	SWA(max)	SWA(mean)
<b>S + SMC (75:25)</b>	<b>0.0395</b>	<b>0.0155</b>	<b>0.0147</b>	<b>0.0035</b>	6.6176	0.5053
<b>SMC</b>	<b>0.2553</b>	<b>0.0791</b>	<b>0.0381</b>	0.0066	<b>9.5859</b>	<b>0.6006</b>
<b>Stanley</b>	0.1728	0.0609	0.0318	<b>0.0130</b>	<b>1.8982</b>	<b>0.3918</b>

## Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### Stanley + SMC 융합 이점

##### ● SMC

- 고차 및 준연속 기반으로써 체터링 및 외란을 효과적으로 억제.
- 비선형성으로써 코너링 강성이 급격히 변하는 곡률 구간에서 적응력 우수.
- 준연속 기반으로써 부드러운 제어 및 강건성 보장.



##### ● Stanley

- 경로 추종에서 단순하면서 효율적인 설계 및 연산 제공.
- Lateral position error, yaw angle error 직접 보정함으로써 추종 성능 우수.



##### ● Stanley + SMC

- 비선형 동역학을 처리하면서도 안정적인 경로 추종 가능.
- 곡률 및 외란이 큰 환경에서도 안정적이고 강건한 제어 가능.
- HOSMC의 고차 연산, Stanley의 실시간성 연산 융합 이점.

## ● Sliding Mode Control(SMC) 횡방향 제어기 구현 및 성능 분석

### Sliding Model Control(SMC)

#### ■ 향후 계획

- HOSMC Ip와 같은 파라미터 설정, Oscillation, Stanley와의 융합 등의 미흡한 부분 보충.
- Super twisting sliding mode control(STSMC) 학습 및 구현.
- HOSMC, STSMC 성능 비교.
- 내년 춘계 학술대회 SMC + Stanley (+ 강화학습).

감사합니다.

국민대학교 지능형 차량 신호 처리 연구실 학부연구생 김지원

2024.11.29(금)



국민대학교  
KOOKMIN UNIVERSITY