

이차 선형 시불변 비최소 위상 시스템의 설정값 조정을 위한 새로운 슬라이딩 모드 제어

A New Sliding Mode Control for Set-point Regulation of Second Order LTI Nonminimum Phase Systems

이 하 준*, 박 철 훈
(Hajoon Lee and Cheol Hoon Park)

Abstract : We deal with second order NMP(Non-Minimum Phase) systems which are difficult to control with conventional methods because of their inherent characteristics of undershoot. In such systems, reducing the undesirable undershoot phenomenon makes the response time of the systems much longer. Moreover, it is impossible to control the magnitude of undershoot in a direct way and to predict the response time. In this paper, we propose a novel two sliding mode control scheme which is capable of determining the magnitude of undershoot and thus the response time of NMP systems a priori. To do this, we introduce two sliding lines which are in charge of control in turn. One is used to stabilize the system and achieve asymptotic regulation eventually like the conventional sliding mode methods and the other to stably control the magnitude of undershoot from the beginning of control until the state meets the first sliding line. This control scheme will be proved to have an asymptotic regulation property. The computer simulation shows that the proposed control scheme is very effective and suitable for controlling the second order NMP system because it can decide the magnitude of undershoot in a direct and stable way and reduce the response time compared with the conventional ones.

Keywords : nonminimum phase systems, sliding mode control, undershoot

I. 서론

비최소 위상 시스템(nonminimum phase system)의 제어는 하향초과(undershoot) 현상 때문에 제어 분야에서 가장 어려운 문제들 중 하나로 현재까지 많은 연구가 진행되고 있다. 영점이 s 평면의 우반면에 존재하는 시스템을 비최소 위상 시스템이라고 부르는데 우반면의 영점은 하향초과를 일으켜서 과도응답 특성에 나쁜 영향을 준다[1]. 응답특성에 나쁜 영향을 주는 극점은 되먹임 제어를 사용해서 이동시킬 수 있지만 영점은 되먹임에 의해서는 이동시킬 수 없기 때문에 제어 시스템의 성능의 한계로 작용한다. 다음의 안정한 이차 비최소 위상 시스템을 생각해 보자.

$$P(s) = \frac{-s+b}{s^2+a_1s+a_0}$$

여기에서 $a > 0$, $b > 0$ 이다. 만약 시스템이 불안정하다면 되먹임 제어기로 시스템을 안정하게 만들 수 있다. 단위 계단 입력에 대하여 초기 값 정리에 의해서 $y(0+) = \lim_{s \rightarrow \infty} sP(s) = -1$ 이고 최종 값 정리에 의해서 $y(\infty) = \lim_{s \rightarrow 0} P(s) = b/a_0 > 0$ 이기 때문에 $y(0+)y(\infty) < 0$ 이 되어서 반드시 하향초과가 발생한다[2]. 비록 PID형태의 선형 제어가 간단하기 때문에 널리 사용되고 있지만 비최소 위상 시스템에 적합한 제어 이득을 찾기 힘들고 선형 제어기로서 성능의 한계가 존재한다.

비최소 위상 시스템의 제어에 관한 연구는 두 가지로 나눌 수 있다. 첫째는 가능한 정착 시간을 짧게 유지시키면서 하향초과의 크기를 줄이는 연구이다. 둘째는 하향초과의 크기를 사전에 결정하는 방법에 관한 연구이다. 이는 첫 번째를 포함하는 좀 더 일반화된 주제로 아직 이에 관한 연구가 전무하다. 비록 비최소 위상 시스템에 관한 연구가 많이 진행되고 있지만 대부분은 첫 번째에 관한 연구에 집중하고 있다[3-9]. 점근적인 조정 문제는 Francis와 Wonham에 의해 일반적인 구조 안에서 연구되어 왔다[3]. 하지만 이러한 접근방식은 하향초과의 크기를 직접적으로 줄이는 방법을 제공하지는 않는다. 응답 시간을 크게 늘리지 않으면서 하향초과의 크기를 줄이는 방법에 관해 많은 연구들이 진행되어 왔다[6-9]. Widrow와 Stearns[6]는 하향초과 현상을 극복하고 제어 성능을 향상시키기 위하여 신경회로망을 사용한 비선형 적응역 제어를 제안하였다. 박상봉[7]과 남동경[8]은 더 좋은 성능을 위해 PI 제어기와 진화 연산으로 학습되는 신경회로망 제어를 결합한 새로운 제어를 제안하였다. 비록 이 방법들이 어느 정도 하향초과 현상을 완화시켜주지만 기본적으로 오프라인 학습이 필요한 문제점이 있다. Antić과 Dimitrijević[9]은 퍼지 규칙을 사용한 퍼지 슬라이딩 모드 제어를 제안하였다. 하지만 퍼지 규칙에서 변수 조정이 쉽지 않은 문제점이 있다. 게다가 이러한 방법들은 모두 직접적으로 하향초과의 크기를 제어하지 못하는 문제점이 있다.

본 논문에서는 이차 비최소 위상 시스템에 대해 하향초과의 크기와 응답시간을 사전에 결정하기 위하여 두 개의 슬라이딩 직선을 이용한 새로운 제어 방법을 제안한다. 슬라이딩 모드 제어를 비최소 위상 시스템에 적용하려는 많은 연구가 되어 왔는데[10-13] 기존의 연구들은 하향초과의 크기를 직접적으로 제어하지 못하는 문제점이 존재한다. 우리는 추가적인 슬라이딩 직선을 사용하여 이 문제를 해결하고자 한다.

* 책임저자(Corresponding Author)

논문접수 : 2007. 4. 15., 채택확정 : 2007. 7. 20.

이하준, 박철훈 : 한국과학기술원 전자전산학부 전기및전자공학전공
(inewth@gmail.com/chpark@kaist.ac.kr)

※ 본 연구는 2007년도 한국과학기술원 BK21 정보기술사업단에 의하여 지원되었음.

기존의 슬라이딩 직선은 시스템을 안정화시키는데 사용하고 추가적인 슬라이딩 직선은 하향초과의 크기를 직접적으로 결정하는데 사용한다. 하향초과와 정착 시간 사이의 이론적인 제어 성능의 한계를 나타내는 Middleton[14,15]의 결과와 비교하여 주어진 하향초과의 크기 아래에서 제안하는 제어기를 통해서 얻어진 정착시간은 이득이 커짐에 따라 이론적인 정착 시간의 최소 값에 가까워짐을 보임으로써 제안하는 제어기가 점근적인 최적임을 보인다. 모의실험을 통해서 비최소 위상 시스템에 대하여 기존의 제어 방법과 비교하여 제안된 제어 방법의 효율성을 보인다.

본 논문의 구성은 다음과 같다. 다음 장에서 전달함수로 주어지는 비최소 위상 시스템을 정규형(normal form)으로 바꾸는 방법에 대해 소개한다. 3장에서는 두 개의 슬라이딩 직선을 이용한 새로운 제어 방법을 제안하고 점근적 안정성을 증명한다. 간단한 문제에 대한 모의실험 결과를 4장에서 보여주고 5장에서 결론 및 추후과제에 대해 논의한다.

II. 선형 시불변 비최소 위상 시스템의 수학적 모델

전달함수가 다음과 같이 주어지는 2차 선형 시불변 비최소 위상 시스템을 생각하자.

$$H(s) = k \frac{s-b}{s^2 + a_1s + a_0} \quad (1)$$

여기에서 $k \neq 0$, b 는 양의 상수이고, a_0, a_1 은 상수이다. 전달함수 (1)은 다음과 같이 상태 방정식으로 표현된다.

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ k \end{bmatrix} u, \quad (2)$$

$$y = [-b \quad 1] \mathbf{x} \quad (3)$$

여기에서 \mathbf{x} 는 상태 벡터, u 는 제어 입력, y 는 시스템 출력이다. 제어 목표는 출력 y 를 설정값 $r(t)$ 로 조정(regulation)하는 것이다. 여기에서 $r(t)$ 는 $0 \leq t < t_0$ 일 때 r_0 , $t_{n-1} \leq t < t_n$ 일 때 r_{n-1} ($n=2, \dots, N$), $t \geq t_N$ 일 때 r_N 이고 r_n ($n=0, \dots, N$) 은 상수이다. $[\eta \quad \xi]^T = \mathbf{T}\mathbf{x}$ 가 되는 변환 행렬 \mathbf{T} 를 $\mathbf{T} = [1, 0; -b, 1]$ 로 정의하자. 그러면 시스템 (2)-(3)은 다음과 같이 정규형[16]으로 변환된다.

$$\dot{\eta} = b\eta + \xi \quad (4)$$

$$\dot{\xi} = -(a_0 + a_1b + b^2)\eta - (b + a_1)\xi + ku \quad (5)$$

$$y = \xi \quad (6)$$

여기에서 η 는 내부 상태이고 ξ 는 외부 상태이다. 내부 동역학에서 출력 ξ 를 0으로 유지시키는 동역학을 영동역학(zero dynamics)[16]이라고 부르고 $\dot{\eta} = b\eta$ 로 주어진다. $b > 0$ 이므로 이 시스템은 불안정한 영동역학을 가지기 때문에 내부동역학을 고려치 않고 단순히 출력 y 를 원하는 출력 r 로 가계 하는 제어 입력 u 를 설계하는 것은 시스템을 발산하게 만든다. 이 문제를 해결하기 위해서 다음 장에서 슬라이딩 모드 제어 방법을 소개한다.

III. 두 개의 슬라이딩 직선을 이용한 제어기 설계

1. 기존의 슬라이딩 모드 제어

기존의 방법으로 Utkin[10]과 Shtessel[12]은 비최소 위상 시스템에 대한 슬라이딩 모드 제어 방법을 제안하였다. 이에 대해 간단히 소개하면 다음과 같다.

먼저 다음과 같이 내부 상태의 오차와 외부 상태의 오차를 정의하자.

$$e_1 = \eta - \eta_d, \quad e_2 = \xi - r \quad (7)$$

여기에서 η_d 는 설정값 r 에 따라 우리가 구해야 할 원하는 내부 상태이다. 원하는 내부 상태를 $\eta_d = -r/b$ 로 취하면 (7)에 의해서 시스템 (4)-(6)은 다음과 같이 표현된다.

$$\dot{e}_1 = be_1 + e_2 \quad (8)$$

$$\dot{e}_2 = -(a_0 + a_1b + b^2)\eta - (b + a_1)\xi + ku \quad (9)$$

여기에서 e_1 은 내부 상태 오차이고 e_2 는 출력 오차이다. 슬라이딩 모드 제어기를 설계하기 위하여 슬라이딩 직선 $s_1 = 0$ 을 다음과 같이 정의하자.

$$s_1 = e_2 + (b + \lambda)e_1 \quad (10)$$

여기에서 $\lambda > 0$ 이다. (10)을 미분하면 다음 식을 얻을 수 있다.

$$\dot{s}_1 = -(a_0 + a_1b + b^2)\eta - (b + a_1)\xi + ku + (b + \lambda)(be_1 + e_2) \quad (11)$$

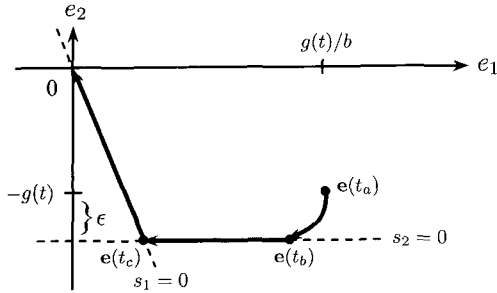
제어 입력 u 는 다음과 같이 표현된다.

$$u = u_{eq_{s_1}} + \Delta u_{s_1} \quad (12)$$

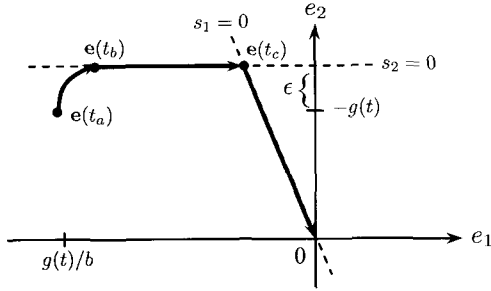
$$u_{eq_{s_1}} = k^{-1} \{ (a_0 + a_1b + b^2)\eta + (b + a_1)\xi - (b + \lambda)(be_1 + e_2) \} \quad (13)$$

$$\Delta u_{s_1} = -k^{-1} \nu_1 \operatorname{sgn}(s_1) \quad (14)$$

여기에서 $u_{eq_{s_1}}$ 은 등가 제어 입력(equivalent control input)이고 Δu_{s_1} 은 떨림 제어 입력(chattering control input)이고 $\nu_1 > 0$ 이다. s_1 에 대한 라푸노프 함수로 $V = s_1^2/2$ 를 잡고 미분을 구하면 $V = s_1 \dot{s}_1 = -\nu_1 |s_1|$ 이 된다. 따라서 오차 궤적은 슬라이딩 직선 $s_1 = 0$ 에 유한 시간 안에 도달하게 되고 그 이후에는 슬라이딩 직선 위에 계속 머무르게 된다. 슬라이딩 직선 위에서 시스템의 동역학은 $\dot{e}_1 = -\lambda e_1$ 으로 표현된다. 이 동역학은 점근적으로 안정하기 때문에 출력 y 는 설정값 r 에 점근적으로 수렴한다. 비록 이 제어 방법은 전체 시스템의 안정성을 보장 할 수 있지만 하향초과 현상을 막을 수는 없다. 이를 보이기 위하여 초기 상태는 원점에 있고 설정 신호로 단위 계단 함수가 인가되었다고 가정하자. (7)로부터 $e_1(0) = 1/b$, $e_2(0) = -1$ 을 얻을 수 있고 이를 (10)에 대입하면 $s_1(0) = \lambda/b$ 가 된다. 이로부터 출력 y 의 미분은 $\dot{y} = \dot{e}_2 = \dot{s}_1 - (b + \lambda)e_1 = -\nu_1 \operatorname{sgn}(s_1) - (b + \lambda)(be_1 + e_2)$ 가 되고 $\dot{y}(0) = -\nu_1 < 0$ 이 된다. 한편, $e_1(\infty) = 0$ 이므로 $y(\infty) = 1$ 이 된다.



(a) The error trajectory when $g(t) > 0$.



(b) The error trajectory when $g(t) < 0$.

그림 1. 제안된 제어 방법의 위상 묘사($\mathbf{e} = [e_1 \ e_2]^T$).

Fig. 1. Phase portrait of the proposed control method ($\mathbf{e} = [e_1 \ e_2]^T$).

따라서 $y(0)y(\infty) < 0$ 이 되므로 하향초과가 반드시 발생한다. 비록 기존의 슬라이딩 모드 제어는 시행착오 과정을 통해 λ 와 ν_1 의 조정으로 하향초과의 크기와 정착 시간을 조절할 수는 있지만 슬라이딩 직선 $s_1 = 0$ 만을 사용해서는 하향초과의 크기를 미리 직접적으로 결정하는 것이 불가능하다. 우리는 하향초과의 크기를 직접적이고 정확한 방법으로 결정하기 위하여 다음 절에서 추가적인 슬라이딩 직선을 소개한다.

2. 두 개의 슬라이딩 직선을 이용한 제어

두 번째 슬라이딩 직선 $s_2 = 0$ 을 다음과 같이 정의하자.

$$s_2 = e_2 + g(t) + \varepsilon \text{sgn}(g(t)) \tag{15}$$

여기에서 $\varepsilon > 0$ 은 허용된 하향초과의 크기이고 $g(t)$ 는 $0 \leq t < t_1$ 일 때 $r_0 - y(0)$, $t_{n-1} \leq t < t_n$ 일 때 $r_{n-1} - y(t_{n-1})$ ($n = 2, \dots, N$), $t \geq t_N$ 일 때 $r_N - y(t_N)$ 으로 정의된다. $g(t) + \varepsilon \text{sgn}(g(t))$ 는 각 시간 간격에서 상수이므로

$$\dot{s}_2 = -(a_0 + a_1 b + b^2)\eta - (b + a_1)\xi + ku \tag{16}$$

가 된다. 제어 입력 u 는 다음과 같이 주어진다.

$$u = u_{eq_{s_2}} + \Delta u_{s_2} \tag{17}$$

$$u_{eq_{s_2}} = k^{-1} \{ (a_0 + a_1 b + b^2)\eta + (b + a_1)\xi \} \tag{18}$$

$$\Delta u_{s_2} = -k^{-1} \nu_2 \text{sgn}(s_2) \tag{19}$$

여기에서 $\nu_2 > 0$ 이다. 제안하는 제어기의 핵심 아이디어는 그림 1에 나타난 것처럼 오차 궤적을 두 번째 슬라이딩 직선에 먼저 도달하게 하고 첫 번째 슬라이딩 직선을 만날 때까

지 두 번째 슬라이딩 직선을 따라서 가게 하는 것이다. 첫 번째 슬라이딩 직선을 만난 뒤에는 첫 번째 슬라이딩 직선을 따라서 원점으로 점근적으로 수렴한다. 우리는 제안하는 제어기가 하향초과의 크기를 결정할 수 있고 점근적으로 안정하다는 것을 보일 것이다.

다음의 보조 정리는 오차 궤적이 두 번째 슬라이딩 직선을 만나고 두 번째 슬라이딩 직선 위에서 궤적이 어느 방향으로 향하는지를 보여준다. 다시 말해서 궤적이 두 번째 슬라이딩 직선을 만난 뒤 두 번째 슬라이딩 직선을 타고 첫 번째 슬라이딩 직선을 향하여 간다는 것을 증명한다.

보조정리 1: 시스템 (1)과 제어 입력 (17)을 생각해 보자. 설정값 신호 $r(t)$ 는 $t < t_a$ 일 때 r_a , $t \geq t_a$ 일 때 r_b 로 주어진다 고 가정하자. 여기에서 r_a 와 r_b 는 상수이다. 그리고 $t < t_a$ 일 때 $e_1(t) = e_2(t) = 0$ 이라고 가정하자. 여기에서 e_1 과 e_2 는 (7)에 정의되어 있다. $t = t_a$ 일 때, $r(t)$ 는 r_a 에서 r_b 로 변하고 $y(t_a) = r_a$ 이기 때문에 내부 상태 오차 $e_1(t_a)$ 와 출력 오차 $e_2(t_a)$ 는 다음과 같이 주어진다.

$$e_1(t = t_a) = g(t)/b, \quad e_2(t = t_a) = -g(t) \tag{20}$$

여기에서 $g(t) = r_b - r_a$ 이다. 그러면 $t > t_a$ 에 대하여 오차 궤적은 슬라이딩 직선 $s_2 = 0$ 에 다음과 같이 주어지는 유한 시간 t_b 안에 도달하게 된다.

$$t_b = t_a + \varepsilon/\nu_2 \tag{21}$$

궤적이 슬라이딩 직선 $s_2 = 0$ 을 만난 뒤에는 직선 $s_2 = 0$ 을 따라서 첫 번째 슬라이딩 직선을 만나게 된다.

증명: 부록 A 참조. ■

전체 제어 입력 u 는 다음과 같이 주어진다.

$$u = \begin{cases} u_{s_1}, & \text{첫 번째 슬라이딩 직선 } s_1 = 0 \text{ 을 만나기 전} \\ u_{s_2}, & \text{첫 번째 슬라이딩 직선 } s_1 = 0 \text{ 을 만난 뒤} \end{cases} \tag{22}$$

여기에서 $u_{s_1} = u_{eq_{s_1}} + \Delta u_{s_1}$, $u_{s_2} = u_{eq_{s_2}} + \Delta u_{s_2}$ 이고 $u_{eq_{s_1}}$, Δu_{s_1} , $u_{eq_{s_2}}$, Δu_{s_2} 는 각각 (13), (14), (18), (19)에 정의되어져 있다. 제어 입력 (22)는 u_{s_2} 에 의해서 $s_2 = 0$ 을 따라서 $s_1 = 0$ 에 도달하게 보장해준다. $s_1 = 0$ 에 도달한 후에는 오차는 그림 1에 보인 것처럼 u_{s_1} 에 의해서 원점으로 간다. 하지만 실제 문제에서는 샘플링 주파수의 한계로 궤적이 슬라이딩 직선 $s_1 = 0$ 에 정확하게 만나지 않을 수 있다. 그래서 실험에서는 제어 입력 (22) 대신에 u 를 $t < t_r$ 일 때 u_{s_2} , $t \geq t_r$ 일 때 u_{s_1} 을 사용한다. 여기에서 t_r 은 $g(t) > 0$ 일 때에는 처음으로 $s_1 < -\Phi$ 를 만족하는 시간이고 $g(t) < 0$ 일 때에는 처음으로 $s_1 > -\Phi$ 를 만족하는 시간이다. Φ 는 양수로 경계층 두께이다. 다음 정리에서 제안된 제어기가 점근적인 안정성을 가짐을 증명한다.

정리 1: 보조정리 1의 가정 하에서 시스템 (1)과 제어 입력 (22)를 생각해 보자. 그러면 $t > t_a$ 에 대하여 오차 궤적은 보조정리 1에 의하여 유한 시간 $t_b = t_a + \varepsilon/\nu_2$ 안에 슬라이딩

직선 $s_2 = 0$ 에 도달한다. 슬라이딩 직선 $s_2 = 0$ 에 도달한 후에는 직선을 따라서 다음과 같이 주어지는 유한시간 t_c 안에 슬라이딩 직선 $s_1 = 0$ 에 도달한다.

$$t_c = t_b + b^{-1} \ln \left\{ \frac{\lambda b (|r_b - r_a| + \varepsilon)}{\nu_2 (b + \lambda) (e^{b\nu_2 t_c} - 1)} \right\} \quad (23)$$

슬라이딩 직선 $s_1 = 0$ 에 도달한 후에는 오차는 원점으로 지수적으로(exponentially) 수렴한다.

증명: 부록 B 참조. ■

따라서 우리는 그림 1에서 보인 것처럼 단지 변수 ε 을 변화시킴으로써 하향초과의 크기를 사전에 제어할 수 있다. 또한 다음의 따름정리를 통해서 제안된 제어 방법의 정착 시간도 구할 수 있다. 먼저 정착 시간을 다음과 같이 정의하자.

정의 1: $\alpha \times 100\%$ 정착 시간 t_s^α 는 다음과 같이 정의된다.

$$t_s^\alpha = \inf_{\tau \in (0, \infty)} \left\{ \tau \mid |y(t) - y(\infty)| \leq \alpha |y(\infty)| \text{ for } t > \tau \right\} \quad (24)$$

여기에서 $y(t)$ 는 시스템 출력이고 $y(\infty) = \lim_{t \rightarrow \infty} y(t)$ 이고 $0 < \alpha < 1$ 이다.

따름정리 1: 시스템 (1)과 제어 입력 (22)를 생각해 보자. 초기 상태는 원점에 있고 원하는 설정값 신호는 단위 계단 함수라고 가정하자. 그러면 정착 시간 t_s^α 는 다음과 같이 주어진다.

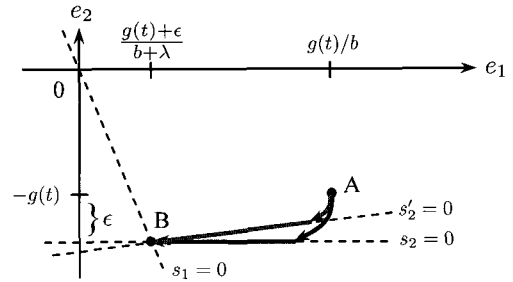
$$t_s^\alpha = \frac{\varepsilon}{\nu_2} + \frac{1}{b} \ln \left\{ \frac{\lambda b (1 + \varepsilon)}{\nu_2 (b + \lambda) (e^{b\nu_2^{-1} \varepsilon} - 1)} \right\} + \frac{1}{\lambda} \ln \frac{1 + \varepsilon}{\alpha} \quad (25)$$

증명: 부록 C 참조. ■

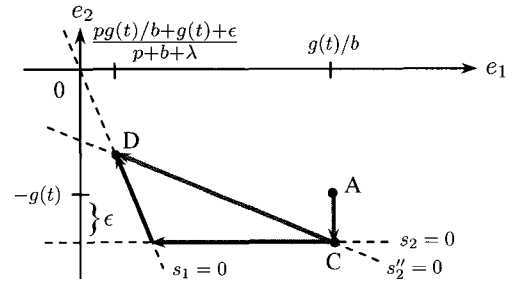
우리는 (25)를 통해 정착 시간 t_s^α 는 시스템의 극점의 위치를 결정하는 a_0, a_1 에는 상관이 없고 영점의 위치에 관련이 있음을 알 수 있다. 제안하는 제어 방법의 정착시간이 주어진 하향초과에 대하여 시간 최적임을 보이기 위하여 하향초과와 정착 시간 사이의 관계를 나타내는 Middleton의 결과 [14,15]와 비교해 본다. Middleton은 단위 계단 입력에 대해 하향초과와 정착 시간 사이에 다음과 같은 트레이드오프 관계가 있음을 증명했다[14].

$$t_{es} \geq b^{-1} \ln(1 + 1/\varepsilon) \quad (26)$$

여기에서 정확한 정착 시간(exact settling time) $t_{es} = \inf_{\tau \in (0, \infty)} \{ \tau \mid y(t) = y(\infty) \text{ for } t > \tau \}$ 로 정의된다. (26)을 살펴보면 우리가 얻을 수 있는 정착 시간의 최소 값은 영점의 위치와 하향초과의 크기와 관련이 있음을 알 수 있다. 여기서 주목할 점은 위의 관계식 (26)은 제어기가 선형이던지 비선형이던지 시불변이던지 시변이던지 무관하다는 점이다. 다시 말해서 우리가 어떠한 제어 법칙을 사용하더라도 (26)의 우변보다 빠른 정착 시간을 구할 수가 없다는 것이다. (25)에서 $\lambda, \nu_2 \rightarrow \infty, \alpha \rightarrow 0$ 면 제안된 제어기의 정확한 정착 시간은 다음과 같이 계산된다.



(a) Phase portrait of the proposed control method using $s_2 = 0$ or $s_2' = 0$.



(b) Phase portrait of the proposed control method using $s_2 = 0$ or $s_2'' = 0$.

그림 2. 두 번째 슬라이딩 직선들의 비교.

Fig. 2. Comparison of second sliding lines.

$$\lim_{\substack{\lambda, \nu_2 \rightarrow \infty \\ \alpha \rightarrow 0}} t_s^\alpha = b^{-1} \ln(1 + 1/\varepsilon) \quad (27)$$

여기에서 우리는 제안된 제어기의 정확한 정착 시간 (27)이 이론적인 정착시간 (26)의 최소 값으로 수렴하는 것을 알 수 있다. 다시 말해서 주어진 하향초과에 대해서 제안된 제어 방법의 정착 시간이 이득을 증가시킴으로써 이론적인 최소 값에 가까이 감을 알 수 있다. 하지만 실제 문제에서는 물리적인 한계와 안정성 때문에 이득을 무한대로 키울 수 없기 때문에 적절한 값으로 사용해야 한다.

3. 두 번째 슬라이딩 직선의 시간 최적 설계

우리는 (15)에서 두 번째 슬라이딩 직선을 e_1 축에 평행하게 설계하여 하향초과의 크기를 조절하였다. 하지만 이 방법이 두 번째 슬라이딩 직선을 만드는 유일한 방법은 아니다. 그림 2에 보인 것처럼 동일한 하향초과의 크기를 가진 다음의 슬라이딩 직선을 생각해 볼 수 있다.

$$s_2' = e_2 + g(t) + \varepsilon - p \left(e_1 - \frac{g(t) + \varepsilon}{b + \lambda} \right) \quad (28)$$

$$s_2'' = e_2 + g(t) + \varepsilon - p(e_1 - g(t)/b) \quad (29)$$

여기에서 p 는 슬라이딩 직선의 기울기이고 $g(t)$ 는 양수라고 가정하자. $g(t)$ 가 음수인 경우에도 동일한 결과를 얻을 수 있다. 본 절에서는 슬라이딩 직선 (15)와 (28), (29)를 비교하여 어떤 것을 사용하는 것이 응답 시간이 가장 빠르는지 살펴본다. 먼저, 그림 2(a)에서처럼 슬라이딩 직선 (15)와 (28)을 비교해보자. $s_2 = 0$ 을 사용하여 궤적이 A에서 B까지 가는데

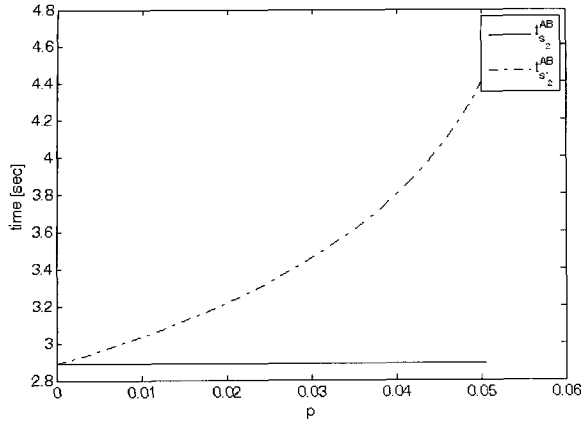


그림 3. $t_{s_2}^{AB}$ 와 $t_{s_2'}^{AB}$ 의 비교.

Fig. 3. Comparison between $t_{s_2}^{AB}$ and $t_{s_2'}^{AB}$.

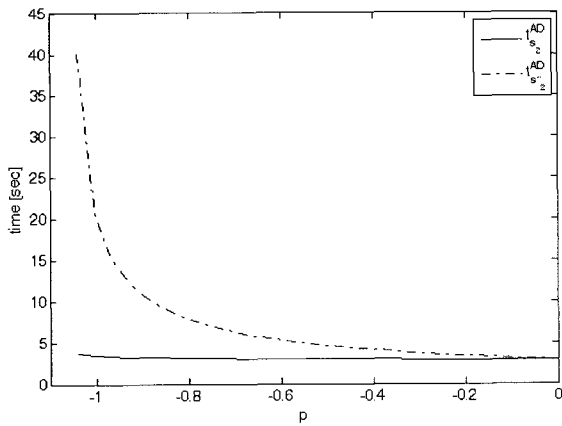


그림 4. $t_{s_2}^{AD}$ 와 $t_{s_2'}^{AD}$ 의 비교.

Fig. 4. Comparison between $t_{s_2}^{AD}$ and $t_{s_2'}^{AD}$.

걸리는 시간을 $t_{s_2}^{AB}$ 로, $s_2' = 0$ 을 사용하여 궤적이 A에서 B까지 가는데 걸리는 시간을 $t_{s_2'}^{AB}$ 로 나타내자. 시간 $t_{s_2}^{AB}$ 는 (23) 으로부터 다음과 같이 주어진다.

$$t_{s_2}^{AB} = \frac{\varepsilon}{v_2} + \frac{1}{b} \ln \left[\frac{\lambda b(g(t) + \varepsilon)}{v_2(b + \lambda)(e^{bv_2^{-1}\varepsilon} - 1)} \right]$$

시간 $t_{s_2'}^{AB}$ 는 다음과 같이 구해진다(부록 D 참조).

$$t_{s_2'}^{AB} = \frac{1}{v_2} \left\{ \varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \right\} + \frac{1}{b + p} \times \ln \frac{\lambda(b + p)(g(t) + \varepsilon)}{v_2(b + \lambda) \left[\exp \left\{ \frac{b + p}{v_2} \left(\varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \right) \right\} - 1 \right]} \quad (30)$$

그림 3은 그림 2(a)에서 $\lambda = 5$, $v_2 = 1$, $g(t) = 1$, $\varepsilon = 0.05$, $b = 1$ 일 때 기울기 p 의 변화에 따라 궤적이 A에서 B까지 가는데 걸리는 시간을 보여준다. 여기에서 시간 $t_{s_2}^{AB}$ 가 $t_{s_2'}^{AB}$ 보다 시간이 적게 걸리는 것을 볼 수 있다. 두 번째로 그림 2(b)에 나타난 것처럼 슬라이딩 직선 (15)와 (29)를 비교해보

자. 두 직선을 사용하여 동일한 하향초과를 얻기 위해서 $v_2 = \infty$ 로 설정하였다. $s_2 = 0$ 을 사용하여 궤적이 A에서 D까지 가는데 걸리는 시간을 $t_{s_2}^{AD}$ 로, $s_2' = 0$ 을 사용하여 궤적이 A에서 D까지 가는데 걸리는 시간을 $t_{s_2'}^{AD}$ 로 나타내자. 시간 $t_{s_2}^{AD}$ 는 다음과 같이 구해진다(부록 E 참조).

$$t_{s_2}^{AD} = \frac{1}{b} \ln \frac{\lambda(g(t) + \varepsilon)}{\varepsilon(b + \lambda)} + \frac{1}{\lambda} \ln \frac{(p + b + \lambda)(g(t) + \varepsilon)}{(b + \lambda)(pg(t)/b + g(t) + \varepsilon)} \quad (31)$$

시간 $t_{s_2'}^{AD}$ 는 다음과 같이 구해진다(부록 F 참조).

$$t_{s_2'}^{AD} = \frac{1}{b + p} \ln \frac{\lambda(pg(t)/b + g(t) + \varepsilon)}{\varepsilon(p + b + \lambda)} \quad (32)$$

그림 4는 그림 2(b)에서 $\lambda = 5$, $g(t) = 1$, $\varepsilon = 0.05$, $b = 1$ 일 때 기울기 p 의 변화에 따라 궤적이 A에서 D까지 가는데 걸리는 시간을 보여준다. 이 경우도 $t_{s_2}^{AD}$ 가 $t_{s_2'}^{AD}$ 보다 시간이 적게 걸리는 것을 알 수 있다. 따라서 두 번째 슬라이딩 직선을 (15)로 설계하는 것이 응답 시간의 관점에서 볼 때 허용된 최대 하향초과 아래에서 최선의 선택임을 알 수 있다.

IV. 컴퓨터 모의실험

제안된 제어기의 성능을 알아보기 위하여 다음의 두 가지의 비최소 위상 시스템을 가지고 컴퓨터 모의실험을 하였다.

$$\text{모델 1: } H_1(s) = \frac{-s + 1}{s^2 + 7s + 12} = \frac{-(s - 1)}{(s + 3)(s + 4)} \quad (33)$$

$$\text{모델 2: } H_2(s) = \frac{-s + 1}{s^2 + 1.872s - 0.256} = \frac{-(s - 1)}{(s + 2)(s - 0.128)} \quad (34)$$

첫 번째 시스템은 안정한 비최소 위상 시스템이고 두 번째 시스템은 불안정한 비최소 위상 시스템이다. 우리는 제안된 제어 방법(TSMC)을 정착 시간과 하향초과의 크기 관점에서 PI 제어 방법(PI), 기존의 슬라이딩 모드 제어 방법(CSMC)과 비교하였다. 실험에서 정착 시간은 2% 정착 시간을 사용하였고 미분 방정식을 풀기 위하여 10 ms 샘플링 시간을 가지고 4차 Runge-Kutta 방법을 사용하였다. 제안된 제어기와 기존의 슬라이딩 모드 제어기에서 제어 입력에 떨림 현상을 막기 위하여 (14), (19)에서 불연속적인 signum 함수 대신에 $\Delta u = -k^{-1}v \text{sat}(s/\Phi)$ 로 주어지는 경계층 방법[16]을 사용하였다. 여기에서 $\text{sat}(s/\Phi)$ 는 $|s| \leq \Phi$ 일 때 s/Φ 이고 $|s| > \Phi$ 일 때 $\text{sgn}(s)$ 이고 Φ 는 경계층 두께이다. 제안된 제어기에서 제어 변수로 $\Phi = 0.01$, $v_1 = v_2 = 1$ 을 사용하였다. 기존의 슬라이딩 모드 제어기에서도 $\Phi = 0.01$ 을 사용하였다. 실험에 사용된 PI 제어기의 제어 이득 K_p, K_i 와 기존의 슬라이딩 모드 제어기의 제어 변수 λ, v_1 은 다목적 진화 알고리즘 중 하나인 NSGA-II[17]로 최적화했다. NSGA-II는 입력이 단

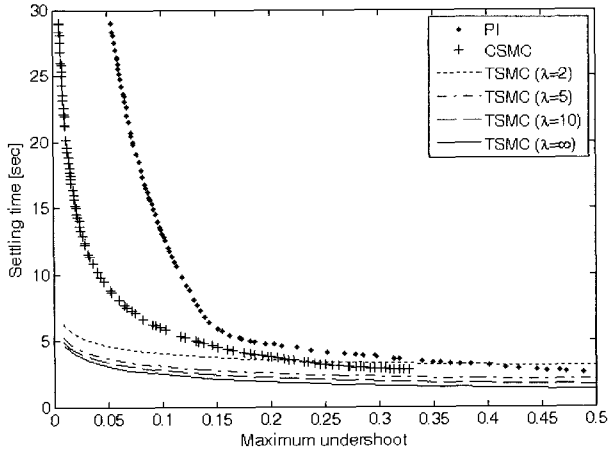


그림 5. 모델 1에 대하여 PI 제어기(PI)와 기존의 슬라이딩 모드 제어기(CSMC)의 얻어진 파레토 해와 λ 의 변화에 따른 제안된 제어기(TSMC)의 제어 결과.

Fig. 5. The obtained pareto solution of the PI controller (PI) and the conventional sliding mode controller(CSMC) and the control result of the proposed two sliding mode controller (TSMC) according to the change of the parameter λ for model 1.

위 계단 함수일 때 최대 하향초과와 정착 시간, 두 가지 목적 함수를 최소화한다. 우리는 다음의 Routh-Hurwitz 안정성 제약 조건하에서 PI 제어기의 변수 K_p, K_I 의 파레토 최적 (Pareto-optimal) 해[18]를 찾았다.

$$\begin{aligned} \text{모델 1: } & K_p < 7, \quad K_I > 0, \\ & (7 - K_p)(12 + K_p - K_I) > K_I \end{aligned} \quad (35)$$

$$\begin{aligned} \text{모델 2: } & K_p < 1.872, \quad K_I > 0, \\ & (1.872 - K_p)(-0.256 + K_p - K_I) > K_I \end{aligned} \quad (36)$$

기존의 슬라이딩 모드 제어기에서는 λ 를 $0 < \lambda \leq 5$, ν_1 을 $0 < \nu_1 \leq 2$ 범위 내에서 NSGA-II로 최적의 해를 찾았다. 더 큰 이득은 떨림 현상을 발생시켜 제어 성능을 떨어뜨리기 때문에 적절한 값으로 설정한 것이다. 실험에서 개체는 100개, 세대는 10,000번을 사용했다. 제안된 제어기는 하향초과의 크기를 결정하는 ε 과 첫 번째 슬라이딩 직선의 기울기 λ 를 변화시켜 가면서 기존의 방법들과 성능을 비교하였다.

1. 모델 1

그림 5는 모델 1에 대해 PI 제어기와 기존의 슬라이딩 모드 제어기의 파레토 해와 제안된 제어기의 λ 값의 변화에 따른 제어 결과를 나타낸다. 그림 5에서 $\lambda = \infty$ 인 경우는 λ 의 변화에 대한 수렴 값으로써 이론적인 한계치를 나타낸다. 우리는 여기에서 λ 값이 증가할수록 제어 성능이 좋아지고 이론적인 한계치 $\lambda = \infty$ 에 수렴해 감을 알 수 있다. 하지만 실제 문제에서는 물리적인 한계치가 존재하고 너무 큰 λ 는 제어 성능을 떨어뜨리고 노이즈나 불확실성에 민감하게 만들어서 시스템을 불안정하게 만들 수 있기 때문에 λ 를 적절한 값으로 설정해야 한다. 그래서 우리는 제안된 제어기의 실험에서 $\lambda = 5$ 를 사용하였다. 비록 그림 5가 하향초과와 정착 시간 사이에 전형적인 트레이드오프 관계를 나타내지

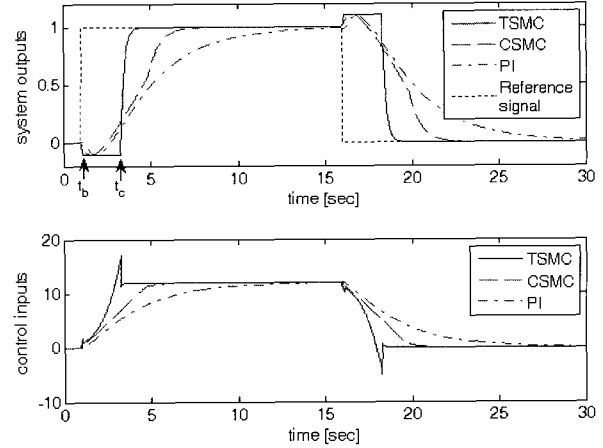


그림 6. 모델 1에 대하여 PI 제어기(PI), 기존의 슬라이딩 모드 제어기(CSMC), 제안된 제어기(TSMC)의 하향초과의 크기가 비슷할 때의 시스템 출력과 제어 입력.

Fig. 6. The system outputs and control inputs of the PI controller, the conventional sliding mode controller and the proposed two sliding mode controller for model 1 when the magnitude of undershoot in the three methods is similar.

만 제안된 제어기의 결과가 PI 제어기와 기존의 슬라이딩 모드 제어기보다 더 적은 하향초과와 정착 시간을 가짐을 볼 수 있다. 특히 제안된 제어기는 작은 하향초과 구간에서 다른 방법들에 비해 월등히 성능이 우수함을 알 수 있다. 그리고 제안된 제어기는 ε 을 변화시켜가면서 직접적으로 하향초과의 크기를 조절할 수 있는 장점이 있다. 하지만 기존의 제어기는 하향초과의 크기를 직접적으로 결정할 방법이 없고 간접적인 방법으로 진화 연산과 같은 시간이 많이 걸리는 작업을 통해서 하향초과의 크기를 결정할 수 있기 때문에 쉽지 않다. 그림 6은 PI 제어기, 기존의 슬라이딩 모드 제어기, 제안된 제어기의 하향초과의 크기가 비슷할 때 시스템 출력과 제어 입력을 나타낸다. 여기에서 PI 제어기의 변수는 $K_p = -0.3395$, $K_I = 2.4266$ 이고 기존의 슬라이딩 모드 제어기의 변수는 $\lambda = 1.5447$, $\nu_1 = 0.4177$ 이고 제안된 제어기의 변수는 $\lambda = 5$, $\varepsilon = 0.1$ 이다. PI 제어기의 변수인 K_p, K_I 와 기존의 슬라이딩 모드 제어기의 변수인 λ, ν_1 은 그림 5에서 하향초과의 크기가 0.1일 때의 변수 값이다. 우리는 그림 6을 통해 제안된 제어기가 PI 제어기와 기존의 슬라이딩 모드 제어기와 비교하여 정착 시간이 빠름을 알 수 있다. 그림 7은 그림 6에 나타난 제안된 제어기의 위상 묘사(phase portrait)를 나타낸다. 그림 6에 나타난 t_b 와 t_c 는 정리 1과 그림 7에 나타난 것처럼 각각 오차 궤적이 두 번째 슬라이딩 직선과 만나는 시간과 첫 번째 슬라이딩 직선과 만나는 시간을 나타낸다. 그림 8은 PI 제어기, 기존의 슬라이딩 모드 제어기, 제안된 제어기의 정착 시간이 비슷할 때의 시스템 출력과 제어 입력을 나타낸다. 여기에서 PI 제어기의 변수는 $K_p = 0.1662$, $K_I = 3.9579$ 이고 기존의 슬라이딩 모드 제어기의 변수는 $\lambda = 2.2422$, $\nu_1 = 0.6134$ 이고 제안된 제어기의 변수는 $\lambda = 5$, $\varepsilon = 0.01$ 이다. 이 경우에도 제안된 제어기가 PI 제어기와 기

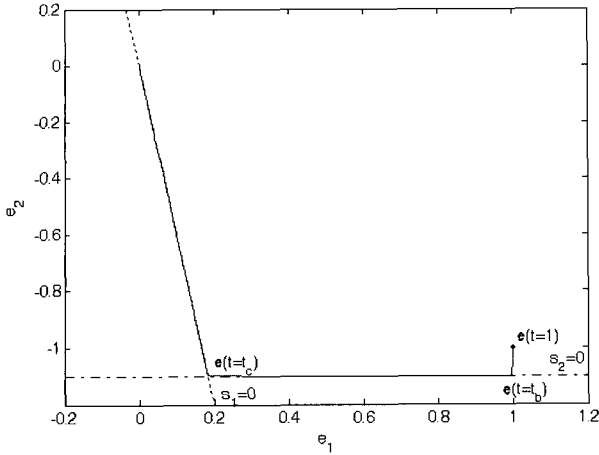


그림 7. 모델 1에 대하여 $\lambda = 5$, $\varepsilon = 0.01$ 일 때 제안된 제어기의 위상 묘사($\mathbf{e} = [e_1, e_2]^T$).

Fig. 7. The phase portrait of the proposed two sliding mode controller for model 1 when $\lambda = 5$, $\varepsilon = 0.01$ ($\mathbf{e} = [e_1, e_2]^T$).

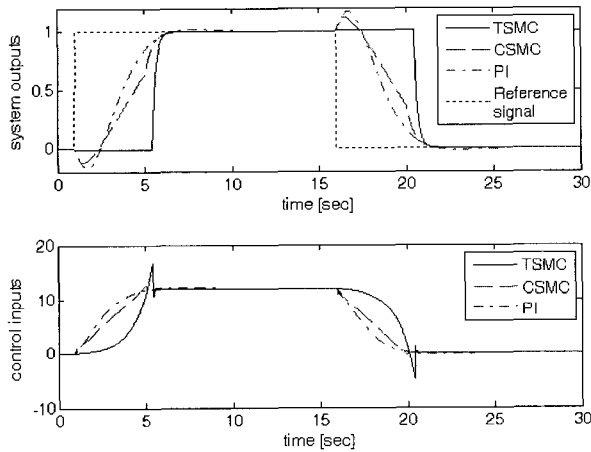


그림 8. 모델 1에 대하여 PI 제어기(PI), 기존의 슬라이딩 모드 제어기(CSMC), 제안된 제어기(TSMC)의 정착 시간이 비슷할 때의 시스템 출력과 제어 입력.

Fig. 8. The system outputs and control inputs of the PI controller, the conventional sliding mode controller and the proposed two sliding mode controller for model 1 when the settling time of the three methods is similar.

존의 슬라이딩 모드 제어기와 비교하여 훨씬 적은 하향초과를 나타냄을 볼 수 있다. 그림 9는 허용된 하향초과의 크기를 나타내는 ε 의 변화에 따른 제안된 제어기의 시스템 출력을 나타낸다. 제안된 제어기는 ε 을 가지고 쉽게 하향초과의 크기를 조절할 수 있음을 알 수 있다.

2. 모델 2

그림 10은 모델 2에 대해 PI 제어기와 기존의 슬라이딩 모드 제어기의 파레토 해와 제안된 제어기의 λ 값의 변화에 따른 제어 결과를 나타낸다. 시스템에 불안정한 극점이 있는 경우 PI 제어기의 성능이 나빠짐을 알 수 있다. 하지만 기존의 슬라이딩 모드 제어기와 제안된 제어기는 등가 제어 입력

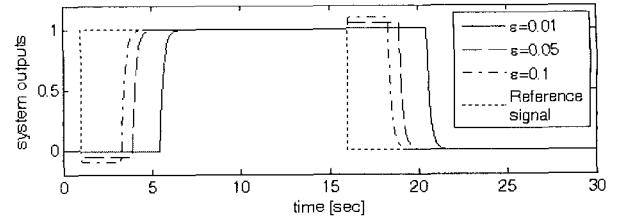


그림 9. 모델 1에 대하여, ε 의 변화에 따른 제안된 제어기 성능($\lambda = 5$).

Fig. 9. Performance of the proposed two sliding mode controller for model 1 according to the change of the parameter ε ($\lambda = 5$).

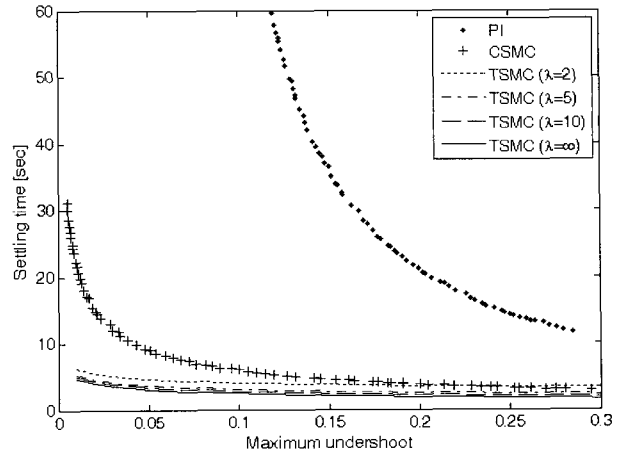


그림 10. 모델 2에 대하여 PI 제어기(PI)와 기존의 슬라이딩 모드 제어기(CSMC)의 얻어진 파레토 해와 λ 의 변화에 따른 제안된 제어기(TSMC)의 제어 결과.

Fig. 10. The obtained pareto solution of the PI controller (PI) and the conventional sliding mode controller(CSMC) and the control result of the proposed two sliding mode controller (TSMC) according to the change of the parameter λ for model 2.

을 사용하기 때문에 시스템의 극점의 위치와는 무관하게 동일한 결과를 나타내었다. 모델 1의 경우와 마찬가지로 제안된 제어기가 다른 제어 방법들 보다 좋은 성능을 나타냄을 알 수 있다. 그림 11은 PI 제어기, 기존의 슬라이딩 모드 제어기, 제안된 제어기의 하향초과의 크기가 비슷할 때의 시스템 출력을 나타낸다. 여기에서 PI 제어기의 변수는 $K_p = 0.5511$, $K_i = 0.0233$ 고 기존의 슬라이딩 모드 제어기의 변수는 $\lambda = 2.2464$, $\nu_1 = 0.7224$ 이고 제안된 제어기의 변수는 $\lambda = 5$, $\varepsilon = 0.15$ 이다. 제안된 제어 방법이 기존의 제어 방법 보다 정착 시간이 빠름을 알 수 있다.

V. 결론 및 추후 과제

본 논문에서는 이차 비최소 위상 시스템의 하향초과 현상을 극복하고자 두 개의 슬라이딩 직선을 이용한 제어 방법을 제안하였다. 제안된 제어 방법은 하향초과의 크기를 직접적으로 조절하면서 전체 시스템의 안정성을 보장한다. 하향초과의 크기는 과도 상태에서 추가적인 슬라이딩 직선을 사용

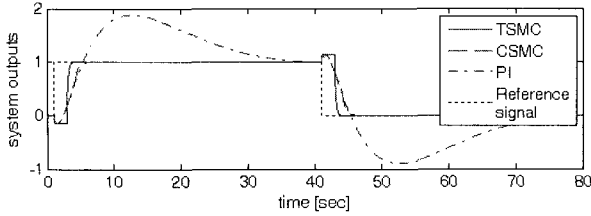


그림 11. 모델 2에 대하여 PI 제어기(PI), 기존의 슬라이딩 모드 제어기(CSMC), 제안된 제어기(TSMC)의 하향초과의 크기가 비슷할 때의 시스템 출력.

Fig. 11. The system outputs of the PI controller, the conventional sliding mode controller and the proposed two sliding mode controller for model 2 when the magnitude of undershoot in the three methods is similar.

하여 원하는 크기로 유지된다. 제안된 제어 방법은 점근적인 안정성을 가짐을 라푸노프 함수를 사용하여 증명하였다. 실험 결과는 제안된 제어 방법이 불안정한 극점을 가지든지 가지 않든지 상관없이 이차 비최소 위상 시스템의 제어에 매우 효율적임을 보여준다. 추후 과제는 다음과 같다. 먼저 불확실성을 가진 비최소 위상 시스템의 제어에 관한 연구가 필요하다. 하나의 양의 영점을 가진 n 차 비최소 위상 시스템과 두 개 이상의 양의 영점을 가진 일반적인 비최소 위상 시스템으로의 확장에 관한 연구가 진행 중이다. 그리고 앞으로 실제 문제에도 적용해 볼 예정이다.

부록 A. 보조정리 1의 증명

$t \geq t_a$ 에 대하여 초기 조건 (20)에서 출발한 오차 궤적은 제어 입력 u (17)을 사용하여 슬라이딩 직선 $s_2 = 0$ 에 도달한다. (16)에 대한 라푸노프 함수로 $V_2 = s_2^2/2$ 를 사용하여 이를 미분하면 $\dot{V}_2 = -v_2|s_2|$ 가 된다. 이로부터 $dV_2/\sqrt{V_2} = -\sqrt{2}v_2 dt$ 를 얻을 수 있고 양변에 t_a 부터 t 까지 적분하여 정리하면 다음 식을 얻을 수 있다.

$$|s_2(t)| = |s_2(t_a)| - v_2(t - t_a) \quad (37)$$

한편, $t = t_a$ 일 때 (20)을 (15)에 대입하면 $s_2(t)$ 의 초기 조건 $s_2(t = t_a) = \varepsilon \operatorname{sgn}(g(t))$ 를 구할 수 있다. 이를 (37)에 대입하면

$$s_2(t) = (\varepsilon - v_2(t - t_a)) \operatorname{sgn}(g(t)), \quad t_a \leq t \leq t_b \quad (38)$$

를 얻는다. 여기에서 $t_b = t_a + \varepsilon/v_2$ 는 $s_2(t = t_b) = 0$ 이 되는 시간이다. (38)을 (15)에 대입하면 $e_2 = -v_2(t - t_a) \operatorname{sgn}(g(t)) - g(t)$ 가 되고 이를 (8)에 대입하면 $\dot{e}_1 = be_1 - v_2(t - t_a) \operatorname{sgn}(g(t)) - g(t)$ 가 된다. 초기 조건 (20)을 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = -b^{-2}v_2(e^{b(t-t_a)} - 1) \operatorname{sgn}(g(t)) + b^{-1}v_2(t - t_a) \operatorname{sgn}(g(t)) + b^{-1}g(t), \quad t_a \leq t \leq t_b \quad (39)$$

$t = t_b$ 일 때 오차 궤적은 그림 1에 나타난 것처럼 슬라이딩 직선 $s_2 = 0$ 에 도달하고 그 위치는 다음과 같이 주어진다.

$$e_1(t = t_b) = -b^{-2}v_2(e^{bv_2^{-1}\varepsilon} - 1) \operatorname{sgn}(g(t)) + b^{-1}\varepsilon \operatorname{sgn}(g(t)) + b^{-1}g(t) \quad (40)$$

$$e_2(t = t_b) = -g(t) - \varepsilon \operatorname{sgn}(g(t)) \quad (41)$$

$t \geq t_b$ 에 대하여 슬라이딩 직선 $s_2 = 0$ 위에서 동역학은 $\dot{e}_1 = be_1 - g(t) - \varepsilon \operatorname{sgn}(g(t))$ 로 표현되고 초기 조건 (40)을 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = -b^{-2}v_2(e^{bv_2^{-1}\varepsilon} - 1)e^{b(t-t_b)} \operatorname{sgn}(g(t)) + b^{-1}(g(t) + \varepsilon \operatorname{sgn}(g(t))), \quad t \geq t_b \quad (42)$$

따라서 $e_1(t)$ 는 $g(t) > 0$ 인 경우에는 $-\infty$ 로 가고 $g(t) < 0$ 인 경우에는 $+\infty$ 로 간다. 즉, 이 두 가지 경우 모두 그림 1에 보인 것처럼 첫 번째 슬라이딩 직선 $s_1 = 0$ 을 만나게 된다.

부록 B. 정리 1의 증명

보조정리 1의 결과로부터 $t_b = t_a + \varepsilon/v_2$ 이다. $s_1(t) = s_2(t)$ 가 되는 시간 t_c 는 (10), (15), (42)를 사용하여 다음과 같이 구할 수 있다.

$$t_c = t_b + b^{-1} \ln \left\{ \frac{\lambda b(|r_b - r_a| + \varepsilon)}{v_2(b + \lambda)(e^{bv_2^{-1}\varepsilon} - 1)} \right\} \quad (43)$$

$t \geq t_c$ 에 대하여 슬라이딩 직선 $s_1 = 0$ 에 도달한 후에는 동역학은 $\dot{e}_1 = -\lambda e_1$, $e_2(t) = -(b + \lambda)e_1(t)$ 로 표현된다. 초기 조건 $e_1(t = t_c) = (g(t) + \varepsilon \operatorname{sgn}(g(t)))/(b + \lambda)$ 를 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = \frac{1}{b + \lambda} (g(t) + \varepsilon \operatorname{sgn}(g(t))) e^{-\lambda(t-t_c)}, \quad t \geq t_c \quad (44)$$

$$e_2(t) = -(g(t) + \varepsilon \operatorname{sgn}(g(t))) e^{-\lambda(t-t_c)}, \quad t \geq t_c \quad (45)$$

따라서 오차 궤적은 그림 1에 보인 것처럼 원점으로 지수적으로 수렴한다.

부록 C. 따름정리 1의 증명

초기 상태는 원점에 있고 설정값 신호는 단위 계단 함수이기 때문에 $g(t) = 1$ 이 된다. 정리 1의 결과를 사용하면 $e_2(t) = -(1 + \varepsilon)e^{-\lambda(t-t_f)}$ 를 얻을 수 있다. 여기에서

$$t_f = \frac{\varepsilon}{v_2} + b^{-1} \ln \left\{ \frac{\lambda b(1 + \varepsilon)}{v_2(b + \lambda)(e^{bv_2^{-1}\varepsilon} - 1)} \right\} \quad (46)$$

이다. 정착 시간의 정의 (24)에 의해서 $|e_2(t)| \leq \alpha$ 이므로 $t \geq t_f + \lambda^{-1} \ln\{(1 + \varepsilon)/\alpha\}$ 가 된다. 따라서 $\alpha \times 100\%$ 정착시간 t_s^α 를 다음과 같이 구할 수 있다.

$$t_s^\alpha = \frac{\varepsilon}{v_2} + \frac{1}{b} \ln \left\{ \frac{\lambda b(1 + \varepsilon)}{v_2(b + \lambda)(e^{bv_2^{-1}\varepsilon} - 1)} \right\} + \frac{1}{\lambda} \ln \frac{1 + \varepsilon}{\alpha} \quad (47)$$

부록 D. t_s^{AB} 의 계산

그림 2(a)에서 A에서 시간을 t_a , 궤적이 $s_2 = 0$ 과 처음

로 만나는 시간을 t'_b , B에서 시간을 t'_c 라고 하자. 그러면 $t = t_a$ 에서 초기 조건은 (20)으로 주어진다. 그리고 $t = t_a$ 에서 $s'_2(t)$ 의 초기 조건은 다음과 같이 주어진다.

$$s'_2(t = t_a) = \varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \quad (48)$$

(28)에 대한 라푸노프 함수로 $W = s_2'^2/2$ 를 사용하면 $\dot{W} = -v_2 |s_2'|$ 이 된다. 이로부터

$$s'_2(t) = \varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) - v_2(t - t_a), \quad t_a \leq t \leq t'_b \quad (49)$$

가 된다. 여기에서 $s'_2(t = t'_b) = 0$ 이 되는 시간 t'_b 는 다음과 같이 주어진다.

$$t'_b = t_a + \frac{1}{v_2} \left\{ \varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \right\} \quad (50)$$

(28)과 (49)를 통해서 $e_2 = p(e_1 - g(t)/b) - g(t) - v_2(t - t_a)$ 를 얻을 수 있고 이를 (8)에 대입하면 $\dot{e}_1 = (b + p)e_1 - g(t) - g(t)p/b - v_2(t - t_a)$ 가 된다. 초기 조건 (20)을 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_2(t) = -(b + p)^{-2} v_2 (e^{(b+p)(t-t_a)} - 1) + (b + p)^{-1} v_2 (t - t_a) + g(t)/b, \quad t_a \leq t \leq t'_b \quad (51)$$

$t = t'_b$ 일 때 오차 궤적은 그림 2(a)에 보인 것처럼 슬라이딩 직선 $s'_2 = 0$ 에 도달하고 그 위치는 다음과 같다.

$$e_1(t = t'_b) = -(b + p)^{-2} v_2 (e^{v_2^{-1}d} - 1) + d + \frac{g(t)}{b} \quad (52)$$

$$d = (b + p) \left(\varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \right)$$

$$e_2(t = t'_b) = -g(t) - \varepsilon + p \left(e_1(t = t'_b) - \frac{g(t) + \varepsilon}{b + \lambda} \right) \quad (53)$$

$t \geq t'_b$ 에 대하여 슬라이딩 직선 $s'_2 = 0$ 위에서의 동역학은 $\dot{e}_1(t) = -(b + p)e_1 - g(t) - \varepsilon - p(g(t) + \varepsilon)/(b + \lambda)$ 로 표현된다. 초기 조건 (52)를 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = -(b + p)^{-2} v_2 (e^{v_2^{-1}d} - 1) e^{(b+p)(t-t'_b)} + (b + p)^{-1} \left(g(t) + \varepsilon + \frac{g(t) + \varepsilon}{b + \lambda} p \right), \quad t'_b \leq t \leq t'_c \quad (54)$$

여기에서 $s_1(t) = s'_2(t)$ 가 되는 시간 t'_c 는 (10), (28), (54)를 사용하여 다음과 같이 구할 수 있다.

$$t'_c = t'_b + \frac{1}{b + p} \ln \frac{\lambda(b + p)(g(t) + \varepsilon)}{v_2(b + \lambda) \{ e^{v_2^{-1}d} - 1 \}} \quad (55)$$

따라서 궤적이 $s'_2 = 0$ 을 사용하여 A에서 B까지 가는데 걸리

는 시간 $t_{s_2}^{AB}$ 는 다음과 같다.

$$t_{s_2}^{AB} = t'_c - t_a = \frac{1}{v_2} \left\{ \varepsilon - p \left(\frac{g(t)}{b} - \frac{g(t) + \varepsilon}{b + \lambda} \right) \right\} + \frac{1}{b + p} \ln \frac{\lambda(b + p)(g(t) + \varepsilon)}{v_2(b + \lambda) \{ e^{v_2^{-1}d} - 1 \}} \quad (56)$$

부록 E. $t_{s_2}^{AD}$ 의 계산

그림 2(b)에서 A에서 시간을 t_a , 궤적이 $s_2 = 0$ 과 처음으로 만나는 C에서 시간을 t_b , 두 개의 슬라이딩 직선이 만나는 곳, 즉 $s_1(t) = s_2(t)$ 에서 시간을 t_c , D에서 시간을 t_d 라고 하자. $t = t_a$ 에서 초기 조건은 (20)과 같다. $v_2 = \infty$ 로 두었기 때문에 (21)로부터 $\lim_{v_2 \rightarrow \infty} t_b = t_a$ 가 되고 (40)으로부터 $\lim_{v_2 \rightarrow \infty} e_1(t = t_b) = g(t)/b$ 가 된다. 슬라이딩 직선 $s_2 = 0$ 위에서의 동역학은 $\dot{e}_1 = be_1 - g(t) - \varepsilon$ 이 된다. 초기 조건 $e_1(t = t_b) = g(t)/b$ 를 가지고 이 미분 방정식을 풀면 $e_1(t) = -b^{-1} \varepsilon e^{b(t-t_b)} + b^{-1}(g(t) + \varepsilon)$, $t_b \leq t \leq t_c$ 가 된다. 여기에서 $s_1(t) = s_2(t)$ 가 되는 시간 t_c 는 다음과 같이 구해진다.

$$t_c = t_b + b^{-1} \ln \frac{\lambda(g(t) + \varepsilon)}{\varepsilon(b + \lambda)} \quad (57)$$

$t \geq t_c$ 에 대하여 슬라이딩 직선 $s_1(t) = 0$ 위에서의 동역학은 $\dot{e}_1 = -\lambda e_1$ 이 된다. 초기 조건 $e_1(t = t_c) = (g(t) + \varepsilon)/(b + \lambda)$ 를 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = \frac{g(t) + \varepsilon}{b + \lambda} e^{-\lambda(t-t_c)}, \quad t \geq t_c \quad (58)$$

$s_1(t) = s_2''(t)$ 가 되는 D에서 시간 t_d 는 다음과 같이 구해진다.

$$t_d = t_c + \frac{1}{\lambda} \ln \frac{(p + b + \lambda)(g(t) + \varepsilon)}{(b + \lambda)(pg(t)/b + g(t) + \varepsilon)} \quad (59)$$

따라서 궤적이 $s_2 = 0$ 을 사용하여 A에서 D까지 가는데 걸리는 시간 $t_{s_2}^{AD}$ 는 다음과 같다.

$$t_{s_2}^{AD} = \frac{1}{b} \ln \frac{\lambda(g(t) + \varepsilon)}{\varepsilon(b + \lambda)} + \frac{1}{\lambda} \ln \frac{(p + b + \lambda)(g(t) + \varepsilon)}{(b + \lambda)(pg(t)/b + g(t) + \varepsilon)} \quad (60)$$

부록 F. $t_{s_2}^{AD}$ 의 계산

$t_{s_2}^{AD}$ 의 경우와 같은 초기 조건 하에서 궤적이 $s_2'' = 0$ 과 처음으로 만나는 C에서 시간을 t''_b , $s_1(t) = s_2''(t)$ 가 되는 D에서 시간을 t''_c 라고 하자. $t_{s_2}^{AD}$ 의 경우와 마찬가지로 $\lim_{v_2 \rightarrow \infty} t''_b = t_a$ 와 $\lim_{v_2 \rightarrow \infty} e_1(t = t''_b) = g(t)/b$ 임을 쉽게 보일 수 있다. 슬라이딩 직선 $s_2'' = 0$ 위에서의 동역학은 $\dot{e}_1 = (b + p)e_1 - pg(t)/b - g(t) - \varepsilon$ 으로 표현되고 초기 조건 $e_1(t = t''_b) = g(t)/b$ 를 가지고 이 미분 방정식을 풀면 그 해는 다음과 같다.

$$e_1(t) = \left\{ \frac{g(t)}{b} - \frac{pg(t)/b + g(t) + \varepsilon}{b+p} \right\} e^{(b+p)(t-t_b)} + \frac{pg(t)/b + g(t) + \varepsilon}{b+p}, \quad t_b'' \leq t \leq t_c'' \quad (61)$$

여기에서 $s_1(t) = s_2''(t)$ 가 되는 시간 t_c'' 는 다음과 같이 구해진다.

$$t_c'' = t_b'' + \frac{1}{b+p} \ln \frac{\lambda(pg(t)/b + g(t) + \varepsilon)}{\varepsilon(p+b+\lambda)} \quad (62)$$

따라서 궤적이 $s_2'' = 0$ 을 사용하여 A에서 D까지 가는데 걸리는 시간 $t_{s_2}^{AD}$ 는 다음과 같다.

$$t_{s_2}^{AD} = \frac{1}{b+p} \ln \frac{\lambda(pg(t)/b + g(t) + \varepsilon)}{\varepsilon(p+b+\lambda)} \quad (63)$$

참고문헌

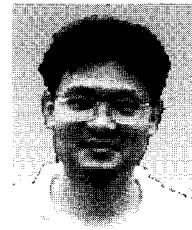
[1] B. A. León de la Barra, "On undershoot in SISO systems," *IEEE Trans. Automat. Contr.*, vol. 39, no. 3, pp. 578-581, Mar. 1994.
 [2] M. Vidyasagar, "On undershoot and nonminimum phase zeros," *IEEE Trans. Automat. Contr.*, vol. 31, no. 5, pp. 440-440, May 1986.
 [3] B. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, pp. 457-465, 1976.
 [4] A. Isidori and C. I. Byrnes, "Output regulation of nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 35, no. 2, pp. 131-140, Feb. 1990.
 [5] S. Devasia, D. Chen, and B. Paden, "Nonlinear inversion-based output tracking," *IEEE Trans. Automat. Contr.*, vol. 41, no. 7, pp. 930-942, July 1996.
 [6] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Englewood Cliffs, NJ: Prentice Hall, 1985.
 [7] S. Park, L.-J. Park, and C. H. Park, "A neuro-genetic controller

for nonminimum phase systems," *IEEE Trans. Neural Networks*, vol. 6, no. 5, pp. 1297-1300, Sept. 1995.
 [8] D. Nam, H. Lee, S. Park, L.-J. Park, and C. H. Park, "A multiobjective evolutionary neuro-controller for nonminimum phase systems," *IEICE Transactions on Information and Systems*, vol. E87-D, no. 11, pp. 2517-2520, Nov. 2004.
 [9] D. Antić and S. Dimitrijević, "Non-minimum phase plant control using fuzzy sliding mode," *Electronic Letters*, vol. 34, no. 11, pp. 1156-1158, May 1998.
 [10] H.-S. Jeong and V. I. Utkin, "Sliding mode tracking control of systems with unstable zero dynamics," in *Variable structure systems: sliding mode and nonlinear control*, K. Young and U. Ozguner, Eds. London: Springer, pp. 303-327, 1999.
 [11] C. Bonivento, L. Marconi, and R. Zanasi, "Output regulation of nonlinear systems by sliding mode," *Automatica*, vol. 37, no. 4, pp. 535-542, Apr. 2001.
 [12] I. A. shkolnikov and Y. B. Shtessel, "Tracking in a class of nonminimum-phase systems with nonlinear internal dynamics via sliding mode control using method of system center," *Automatica*, vol. 38, no. 5, pp. 837-842, May 2002.
 [13] S. Gopalswamy and J. K. Hedrick, "Tracking nonlinear nonminimum phase systems using sliding control," *International Journal of Control*, vol. 57, no. 5, pp. 1141-1158, 1993.
 [14] R. H. Middleton, "Trade-offs in linear control system design," *Automatica*, vol. 27, no. 2, pp. 281-292, 1991.
 [15] K. Lau, R. H. Middleton, and J. H. Braslavsky, "Undershoot and settling time tradeoffs for nonminimum phase systems," *IEEE Trans. Automat. Contr.*, vol. 48, no. 8, pp. 1389-1393, Aug. 2003.
 [16] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2002.
 [17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182-197, Apr. 2002.
 [18] K. Deb, *Multi-objective Optimization Using Evolutionary Algorithms*, Wiley, 2001.



이 하 준

2000년 경북대학교 전자전기공학부 학사. 2002년 한국과학기술원 전자전산학과 석사. 2007년 한국과학기술원 전자전산학과 박사. 관심분야는 지능제어, 로봇제어, 신경회로망.



박 철 훈

1984년 서울대학교 전자공학과 학사. 1985년 Caltech 전자공학과 석사. 1990년 Caltech 전자공학과 박사. 1991년~현재 한국과학기술원 전자전산학부 교수. 관심분야는 지능시스템, 신경회로망, 최적화, 지능제어.