Sliding Mode Control의 기초

심 형 보

2011년 8월 17일

본 자료의 내용 중 일부 내용은 엄밀한 의미에서는 좀더 기술적인 조건이 필요한 경우도 있습니다. 그러나 핵심 개념을 짧은 시간 동안 전달하기 위하여 불필요하게 기술적인 부분은 언급하지 않습니다. 따라서 해당 내용을 연구에 응용하려 할 경우 관련 책과 논문 등을 참고하기 바랍니다.

- R.A. DeCarlo, S.H. Zak, & G.P. Matthews (1988) Variable structure control of nonlinear multivariable systems: a tutorial. *Proceedings of the IEEE*
- J.Y. Hung, W. Gao, & J.C. Hung (1993) Variable Structure Control: A Survey. *IEEE Trans. on Industrial Electronics*
- K.D. Young, V.I. Utkin, & U. Ozguner (1998) A control engineer's guide to sliding mode control. *IEEE Trans. Control Systems Technology*
- C. Edwards & S. Spurgeon (1998) *Sliding mode control: Theory and applications*, Taylor & Francis

Non-vanishing perturbation이 있을 때도 원하는 평형점으로 수렴시키는 제어 방법이 있을까요?

$$
\begin{aligned}
\text{Vanishing case:} \quad & \dot{x} = ax + u, \quad |a| \le 1 \\
& u = -2x \quad \Rightarrow \quad \dot{x} = -(2-a)x \\
\text{Non-vanishing case:} \quad & \dot{x} = \delta + u, \quad |\delta| \le 1 \\
& u = -k_1 x(t) - k_2 \int_0^t x(s)ds \\
\text{Non-vanishing case:} \quad & \dot{x} = \delta(t) + u, \quad |\delta(t)| \le 1 \\
& u = \quad ?
\end{aligned}
$$

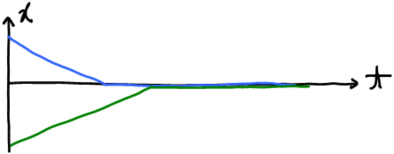
Note: Feedback stabilization의 두 원리

Cancellation vs. Domination

Discontinuous feedback으로 non-vanishing case를 해결할 수 있다

$$\dot{x} = \delta + u, \quad |\delta| \le 1$$

$$u = \alpha(x) = \begin{cases} -2 & \text{if } x > 0 \\ +2 & \text{if } x < 0 \\ \star & \text{if } x = 0 \end{cases}$$



$$\Rightarrow \qquad \dot{x} \begin{cases} < 0 & \text{if } x > 0 \\ > 0 & \text{if } x < 0 \\ = \delta + \star & \text{if } x = 0 \end{cases}$$

그러므로

- 원점으로 잘 수렴한다
- 수렴 후 원점에 머물기 위해서는 $\star = -\delta$여야 한다

- $\dot{x} = \delta + \alpha(x) =: f(x)$는 'discontinuous right-hand side'를 가진 미분 방정식이다. (따라서 해의 존재성과 유일성이 자동으로 보장되지 않는다.)
- 이 경우, $\star = -\delta$이면 모든 초기값에 대하여 해는 잘 정의되고 유일하다.
- 그런데 $\star = -\delta$일 수가 없다 (왜일까?)

(어쩔 수 없이) 새로운 제어식:

$$u = \alpha(x) = \begin{cases} -2 & \text{if } x > 0 \\ +2 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \end{cases}$$

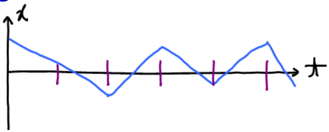
이 경우 고전적인 해의 개념으로는 해가 존재하지 않아 문제가 된다. (이론가의 입장)

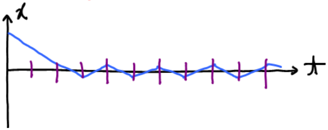
Engineer의 입장에서는 문제가 없다

$$u = \alpha(x) = \begin{cases} -2 & \text{if } x > 0 \\ +2 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \end{cases}$$

구현 시 digital controller에서 $x = 0$의 경우가 발생하지 않는다.
(이론가는 여전히 불만이다.)

Engineer가 신경쓰는 다른 문제: "chattering"





- 실제 digital 구현에서는 sampling period 동안에 제어값을 switching하지 못한다.
- 실제로는 actuator와 sensor에도 dynamics가 있어 delay 효과 발생, 원하는 switching 파형을 제대로 낼 수가 없다.
- 이들 효과가 작아진다면, chattering 효과도 줄어 든다.

Differential Inclusion이란 무엇인가요?

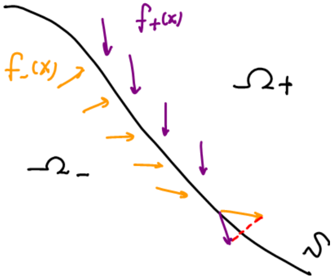
$$\dot{x} \in F(x), \qquad F(x) : \mathbb{R}^n \to \text{subset of } \mathbb{R}^n\text{, set-valued map}$$

위 식을 만족하는 $x(t)$를 differential inclusion의 해라고 한다.

(예) $n = 1$일 때, $\dot{x} \in F(x) = \{z \in \mathbb{R} : z \leq 1\} \quad \Leftrightarrow \quad \dot{x}(t) \leq 1$

Filippov solution이란 무엇인가요?

배경: 상태 공간 $\mathbb{R}^n$이 집합 Ω_+와 Ω_-로 나누어져 있고, 경계선을 집합 S라 하고, 시스템 식이 아래와 같다고 하자.



$$\dot{x} = f(x) = \begin{cases} f_+(x), & \text{at } \Omega_+ \\ f_-(x), & \text{at } \Omega_- \\ \text{don't care}, & \text{at } S \end{cases}$$

단, $f_+(x)$와 $f_-(x)$는 C^1 vector fields.

('don't care'의 값에 따라 고전적인 해의 개념으로는 해가 존재하지 않을 수 있다. 그런데 Engineer의 sense로는 해가 잘 존재하고 별 문제가 없다.)

Definition

$\dot{x} = f(x)$의 'solution in the sense of Filippov'란 아래와 같이 정의된 set-valued map $F(x)$에 대하여 differential inclusion $\dot{x} \in F(x)$의 해를 말한다. 단,

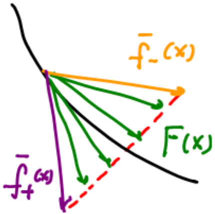
$$F(x) = \begin{cases} \{f_+(x)\} & \text{if } x \in \Omega_+ \\ \{f_-(x)\} & \text{if } x \in \Omega_- \\ \text{co}\{\bar{f}_+(x), \bar{f}_-(x)\} & \text{if } x \in S \end{cases}$$

여기서

$$\bar{f}_+(x) := \lim_{z \to x, z \in \Omega_+} f_+(z),$$

$$\bar{f}_-(x) := \lim_{z \to x, z \in \Omega_-} f_-(z)$$

$$\text{co}\{\bar{f}_+(x), \bar{f}_-(x)\} = \{\alpha \bar{f}_+(x) + (1-\alpha)\bar{f}_-(x) : \alpha \in [0,1]\}$$

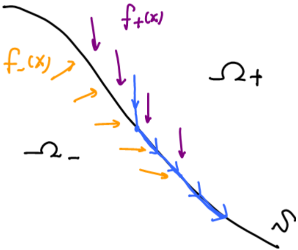


Example

$$\dot{x} = f(x) = \begin{cases} -1, & x > 0 \\ +1, & x < 0 \\ +1, & x = 0 \end{cases} \quad \Rightarrow \quad \dot{x} \in F(x) = \begin{cases} -1, & x > 0 \\ +1, & x < 0 \\ [-1, +1], & x = 0 \end{cases}$$

$\dot{x} = f(x)$는 특정 시간 이후에 해를 갖지 않지만, $\dot{x} \in F(x)$는 해를 갖는다. 따라서, $\dot{x} = f(x)$는 고전적인 의미의 해는 없을 수 있지만, Filippov sense의 해는 항상 존재한다.

Equivalent control이란 무슨 말인가요?

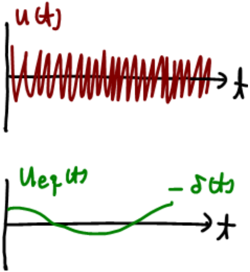


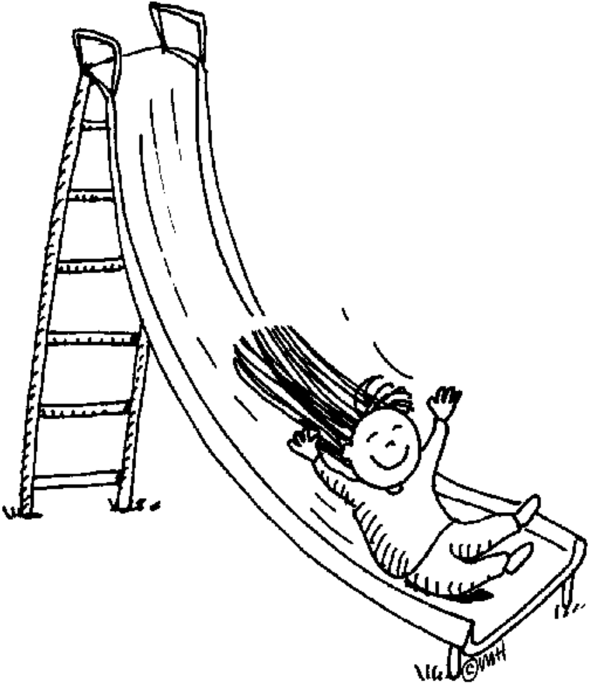
파란색 해를 만드는 control:
"equivalent control" = net-effect of switching control

예:

$$\dot{x} = \delta(t) + u$$
$$u = -K\mathrm{sgn}(x)$$

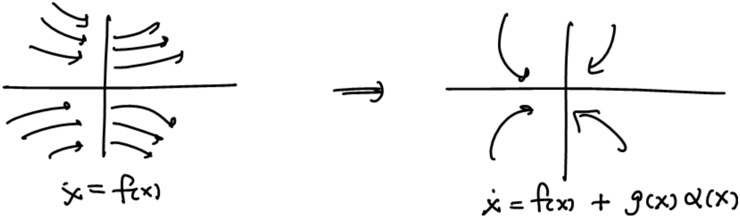
이 경우, $u_{eq} = -\delta(t)$



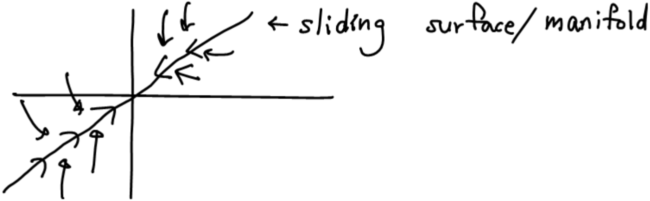

©WMH

Sliding Mode Control은 무엇인가요?

Feedback stabilization (in phase portrait):



Sliding mode control:

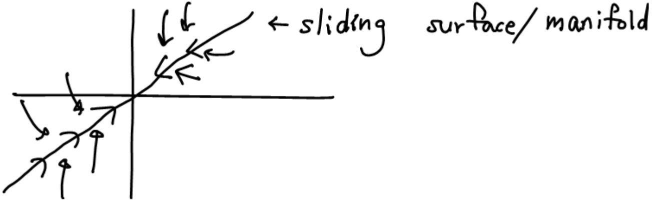


- 유한 시간 내에 정말로 들어간다
- 그 이후 sliding surface를 타고 “흐른다” ($\rightarrow$ sliding)

무엇 때문에 이것이 필요한가요?

Robust Control로 쓰려 하기 때문입니다.

Model uncertainty와 external disturbance에도 불구하고



1. sliding surface로 보내는 법을 알고 있고
2. 일단 sliding surface에 들어오면 uncertainty에 영향을 받지 않고 원점으로 쉽게 보낼 수 있기 때문이다.

Sliding Mode Controller 설계

2차 시스템의 예

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = h(x) + g(x)u$$

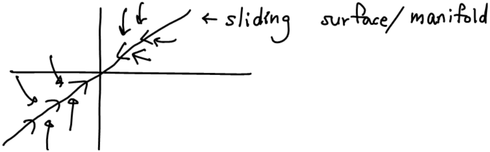
$h(x)$, $g(x)$: unknown, $g(x) \geq g_0 > 0$ and g_0: known

이 때 함수 $s(x) := kx_1 + x_2$라고 정의하고, 제어기가 시스템 상태변수를 $s(x(t)) = 0$이 유지되도록 만들 수 있다면,

$$s(x(t)) = kx_1(t) + x_2(t) = 0 \quad \Rightarrow \quad \begin{cases} \dot{x}_1(t) = -kx_1(t) \\ x_2(t) = -kx_1(t) \end{cases}$$

- $s(x)$는 설계자가 정하므로 uncertainty가 없다. (Design of sliding surface $\{x : s(x) = 0\}$)
- $s(x(t)) = 0$이 유지되면, 시스템은 uncertainty와 무관하게 정해진 방식으로 동작한다.
- 남은 일은 $s(x(t)) = 0$이 된 후, 이를 유지하도록 제어기를 설계하는 것이다. (Design of control u for Reaching phase)

- 해는 보통 Reaching phase 후에 Sliding phase를 겪으면서 원점으로 수렴해 간다.



Reaching phase에서 $s(x(t)) \to 0$을 보장하려면?
$V(x) = \frac{1}{2}s(x)^2$라고 하면,

$$\begin{aligned}\dot{V} = s\dot{s} &= s[kx_2 + h(x) + g(x)u] \\ &\leq |s|g(x)\left|\frac{kx_2 + h(x)}{g(x)}\right| + g(x)su\end{aligned}$$

다음 조건을 만족하는 함수 $\rho(x)$를 정한다:

$$\left|\frac{kx_2 + h(x)}{g(x)}\right| \leq \rho(x), \qquad \forall x$$

for all uncertainties $h(x)$ and $g(x) \geq g_0$.

$$
\begin{aligned}
\dot{V} &\le |s| g(x) \left| \frac{k x_2 + h(x)}{g(x)} \right| + g(x) s u \\
&\le |s| g(x) \rho(x) + g(x) s u
\end{aligned}
$$

이제 제어기를 제안하면,

$$
\begin{aligned}
\text{Option 1:} \quad u &= -\mathrm{sgn}(s)\rho(x) - s \\
\text{Option 2:} \quad u &= -\mathrm{sgn}(s)\rho(x) - \mathrm{sgn}(s) \\
&= -[\rho(x) + 1] \cdot \mathrm{sgn}(s(x))
\end{aligned}
$$

그러면,

$$
\begin{aligned}
\text{Option 1:} \quad & \dot{V} \le -g(x) s^2 \le -2 g_0 V \\
& \Rightarrow \quad V(x(t)) \le e^{-2 g_0 t} V(x(0)) \\
\text{Option 2:} \quad & \dot{V} \le -g_0 |s| \le -g_0 \sqrt{2V} \\
& \Rightarrow \quad \text{유한 시간 내에 sliding surface에 도달}
\end{aligned}
$$

- $\dot{V} \le -g_0\sqrt{2V}$면 왜 유한 시간 내에 $V = 0$이 된다는 건가요?

  $W(x) := \sqrt{2V(x)} = |s(x)|$라 하면,

  $$\dot{W} = \frac{2\dot{V}}{2\sqrt{2V}} \le -g_0$$

  가 되므로, 유한 시간 내에 $W(x(t))$가 0이 됨을 알 수 있다.

- $V = 0$이 된 후에는 어떻게 되나요?

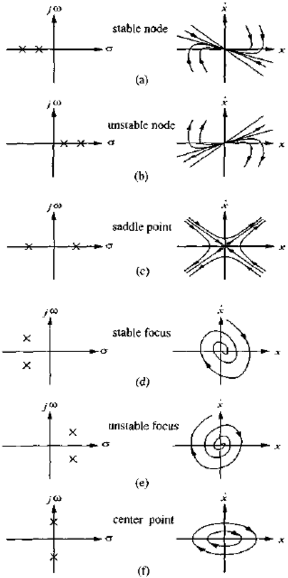
  $\mathrm{sgn}(0)$의 값에 무관하게,

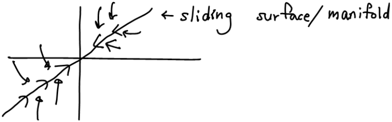
  - Engineer의 관점: chattering이 있겠지만 $s(x(t)) = 0$을 거의 유지
  - 이론가의 관점: solution $x(t)$ in the sense of Filippov는 $s(x(t)) = 0$을 계속 만족

- 유한시간 내에 $s(x(t))$가 0이 되고, 그 이후에는 $x(t)$가 0으로 수렴한다고 하면 안정도 증명이 끝난 것인가요?

  답:

이름이 왜 Sliding Mode Control인가요?





- VSC (Variable Structure Control)
- Optimal control의 Bang-bang control과의 관계는?

정리: Sliding Mode Control의 설계

Sliding surface를 설계하고 입력 u를 설계한다.

1. [Sliding phase 설계] Sliding surface의 설계는 $s(x)$를 정하는 것으로, $S := \{x : s(x) = 0\}$인 surface 위에서 시스템이 원하는 동작(예: 원점으로 수렴)하도록 정한다.
   - 시스템에 uncertainty가 있더라도 S 위에서는 uncertainty가 나타나지 않는 경우가 대부분이고, 나타난다 하더라도 쉽게 다룰 수 있는 경우가 많다.
   - S 위에서 시스템 차수는 작아진다.
   - S는 invariant set일까?
2. [Reaching phase 설계] $V(x) = \frac{1}{2}s(x)^2$로 두고 $\dot{V}$를 계산한 후 $\dot{V} \leq -c\sqrt{2V}$가 되도록 (즉, $s\dot{s} \leq -c|s|$가 되도록) u를 정한다.
   - 이 과정에서 sgn 함수가 필요하다.
   - Uncertainty를 다루는 것도 이 과정에서 이루어진다.

설계 예제: 복습, 재해석 및 변주

Pendulum system:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -c_1 \sin(x_1 + c_2) - c_3 x_2 + c_4 u$$

Control goal: Stabilize the origin under the variation of c_1, c_2, c_3, and $c_4 \geq c_0 > 0$. (c_1, c_2, c_3, c_4 are unknown, but the bounds of their variations are known.)

Sliding surface design: Let $s(x) = kx_1 + x_2$.
이것을 $(x_1, x_2) \leftrightarrow (x_1, s)$간의 좌표변환이라 생각해도 좋다. 이 경우,

$$\dot{x}_1 = -kx_1 + s$$
$$\dot{s} = -k^2 x_1 + ks - c_1 \sin(x_1 + c_2) - c_3(s - kx_1) + c_4 u$$

지금까지 배운 것을 다시 음미해 보자.

Reaching phase design:

$$
\begin{aligned}
s\dot{s} &= s[-k^2 x_1 + ks - c_1 \sin(x_1 + c_2) - c_3(s - kx_1)] + c_4 s u \\
&\leq |s|\rho(x_1, s) + c_4 s u
\end{aligned}
$$

where

$$
|-k^2 x_1 + ks - c_1 \sin(x_1 + c_2) - c_3(s - kx_1)| \leq \rho(x_1, s)
$$

With

$$
u = -\mathrm{sgn}(s)\frac{1}{c_0}[\rho(x_1, s) + K], \qquad K > 0,
$$

we have

$$
s\dot{s} \leq |s|\rho - |s| \cdot \frac{c_4}{c_0} \cdot [\rho + K] \leq -K|s|.
$$

다시 한번만 생각해 봅시다.

- 시스템에 uncertainty가 있더라도 S 위에서는 uncertainty가 나타나지 않는 경우가 대부분이고, 나타난다 하더라도 쉽게 다룰 수 있는 경우가 많다.

$$\dot{x}_1 = x_2 + d(t)$$

- S 위에서 시스템 차수는 작아진다.

$$\dot{x}_1 = -kx_1, \qquad x_2 = -kx_1$$

- 유한시간 내에 $s(x(t))$가 0이 되고, 그 이후에는 $x(t)$가 0으로 수렴한다고 하면 안정도 증명이 끝난 것인가요?

$$\begin{aligned}\dot{x}_1 &= -x_1 + x_1^2 x_2 \\ \dot{x}_2 &= u\end{aligned} \qquad s(x) = x_2$$

$$\dot{x}_1 = -x_1 + x_1^2 s(t), \qquad s(t) = 2 - t$$

Let $w := 1/x_1$.

$$\dot{w} = -\frac{\dot{x}_1}{x_1^2} = \frac{1}{x_1} - s(t) = w - (2 - t)$$

$$w(t) = e^t w(0) - e^t + 1 - t$$

따라서 매우 작은 $w(0) > 0$ (즉, 매우 큰 $x_1(0) > 0$)에 대하여,

$$w(0) > 0, \qquad w(2) = e^2(w(0) - 1) - 1 < 0$$

즉, $x_1(t) \to \infty$ in finite-time.

Chattering을 없앨 수는 없나요?

방법1 아는 만큼은 cancel 하자.

(예) $\dot{x} = ax + \delta + u$, $1 \le a \le 3$, $1 \le \delta \le 3$

$u = -2x - 2 + v$라고 하면,

$\dot{x} = (a-2)x + (\delta - 2) + v$가 된 후, v를 SMC로 설계

방법2 $\mathrm{sgn}(s)$를 $\mathrm{sat}(\frac{s}{\epsilon})$ 함수로 근사하자.