Control of Nonlinear Systems

Nicolas Marchand

CNRS Associate researcher

Nicolas.Marchand@grenoble-inp.fr

Control Systems Department, **gipsa**-lab

Grenoble, France

PSPI

Master PSPI 2009-2010

References

- Nonlinear systems Khalil - Prentice-Hall, 2002
  Probably the best book to start with nonlinear control
- Nonlinear systems S. Sastry - Springer Verlag, 1999
  Good general book, a bit harder than Khalil's
- Mathematical Control Theory - E.D. Sontag - Springer, 1998
  Mathematically oriented, Can be downloaded at:
  `http://www.math.rutgers.edu/~sontag/FTP_DIR/sontag_mathematical_control_theory_springer98.pdf`
- Constructive nonlinear control - Sepulchre et al. - Springer, 1997 *More focused on passivity and recursive approaches*
- Nonlinear control systems - A. Isidori - Springer Verlag, 1995
  A reference for geometric approach
- Applied Nonlinear control - J.J. Slotine and W. Li - Prentice-Hall, 1991
  An interesting reference in particular for sliding mode
- "Research on Gain Scheduling" - W.J. Rugh and J.S. Shamma - Automatica, 36 (10):1401–1425, 2000
  "Survey of gain scheduling analysis and design" - D.J. Leith and W.E. Leithead - Int. Journal of Control, 73:1001–1025, 1999
  Surveys on Gain Scheduling, The second reference can be downloaded at:
  `http://citeseer.ist.psu.edu/leith99survey.html`

Outline

Outline

1. Introduction
   - Linear versus nonlinear
   - The X4 example
2. Linear control methods for nonlinear systems
   - Antiwindup
   - Linearization
   - Gain scheduling
3. Stability
4. Nonlinear control methods
   - Control Lyapunov functions
   - Sliding mode control
   - State and output linearization
   - Backstepping and feedforwarding
   - Stabilization of the X4 at a position
5. Observers

Introduction

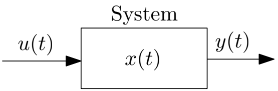
- Some **preliminary vocable and definitions**
- **What is control ?**
- A very **short review of the linear case** (properties, design of control laws, etc.)
- **Why nonlinear control ?**
- **Formulation of a nonlinear control problem** (model, representation, closed-loop stability, etc.)?
- Some **strange** possible **behaviors of nonlinear systems**
- **Example** : The X4 helicopter

Outline

1. Introduction
   - Linear versus nonlinear
   - The X4 example
2. Linear control methods for nonlinear systems
   - Antiwindup
   - Linearization
   - Gain scheduling
3. Stability
4. Nonlinear control methods
   - Control Lyapunov functions
   - Sliding mode control
   - State and output linearization
   - Backstepping and feedforwarding
   - Stabilization of the X4 at a position
5. Observers

Some preliminary vocable

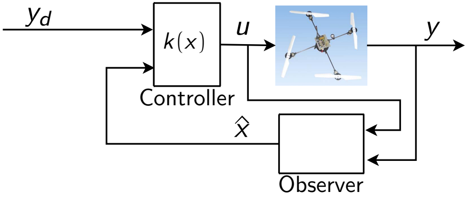
- We consider a **dynamical system**:



where:

- y is the **output**: represents what is "visible" from outside the system
- x is the **state** of the system: characterizes the state of the system
- u is the **control input**: makes the system move

The 4 steps to control a system



1. **Modelization**
   - To get a mathematical representation of the system
   - Different kind of model are useful. Often:
     - a simple model to build the control law
     - a sharp model to check the control law and the observer
2. Design the **state reconstruction**: in order to reconstruct the variables needed for control
3. Design the **control** and test it
4. **Close the loop on the real system**

Linear dynamical systems

Some properties of linear system (1/2)

- **Definition:** Systems such that if y_1 and y_2 are the outputs corresponding to u_1 and u_2, then $\forall\lambda \in \mathbb{R}$:

  $y_1 + \lambda y_2$ is the output corresponding to $u_1 + \lambda u_2$
- **Representation** near the operating point:
  - **Transfer function**:

$$y(s) = h(s)u(s)$$

  - **State space representation**:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx(+Du) \end{cases}$$

- **The model** can be obtained
  - **physical modelization** (eventually coupled with identification)
  - **identification** (black box approach)

both give $h(s)$ or (A, B, C, D) and hence the model.

Linear dynamical systems

Some properties of linear system (2/2)

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx(+Du) \end{cases}$$

- **Nice properties**:
  - **Unique** and **constant equilibrium**
  - **Controllability** (resp. **observability**) directly given by $\operatorname{rank}(B, AB, \ldots, A^{n-1}B)$ (resp. $\operatorname{rank}(C, CA, \ldots, CA^{n-1})$)
  - **Stability** directly given by the **poles of $h(s)$ or the eigenvalues of A** (asymptotically stable $\mathfrak{R} < 0$)
  - **Local properties = global properties** (like stability, stabilizability, etc.)
  - The **time behavior** is **independent of the initial condition**
  - **Frequency analysis** is **easy**
  - **Control is easy**: simply take $u = Kx$ with K such that $\mathfrak{R}(\operatorname{eig}(A + BK)) < 0$, the closed-loop system $\dot{x} = Ax + BKx$ will be asymptotically stable
  - ...
- **Mathematical tool**: linear algebra
- **This is a caricature of the reality** (of course problems due to uncertainties, delays, noise, etc.)

Why nonlinear control ?

Why nonlinear control if linear control is so easy ?

- **All physical systems are nonlinear** because of
  - Actuators saturations
  - Viscosity (proportional to speed^2)
  - Sine or cosine functions in robotics
  - Chemical kinetic in exp(temperature)
  - Friction or hysteresis phenomena
  - ...
- **More and more** the **performance** specification requires nonlinear control (eg. automotive)
- More and more controlled systems are **deeply nonlinear** (eg. µ-nano systems where hysteresis phenomena, friction, discontinuous behavior)
- Nonlinear control is sometimes necessary (oscillators, cyclic systems, . . .)

Nonlinear dynamical systems

How to get a model ?

- **Representation**:
  - **State space representation**:

ODE Ordinary differential equation:

In this course, only:

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned}$$

PDE Partial differential equation (traffic, flow, etc.):

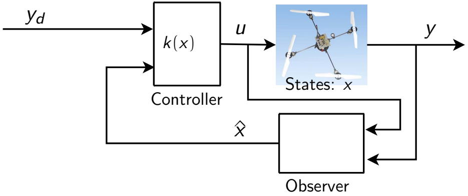
$$\begin{aligned} 0 &= g(x(t), \dot{x}(t), \frac{\partial f(x, \dots)}{\partial x} u(t)) \\ 0 &= h(y(t), x(t), u(t)) \end{aligned}$$

... Algebraic differential equations (implicit), hybrid (with discrete or event based equations), etc.

- **The model** can be obtained
  - **physical modelization** and then nonlinear identification of the parameters (identifiability problems)

Nonlinear dynamical systems

Open-loop control versus closed-loop control ?



Open-loop control

find $u(t)$ such that $\lim_{t\to\infty} \|y(t) - y_d(t)\| = 0$

Widely used for path planning problems (robotics)

Closed-loop control

find $u(x)$ such that $\lim_{t\to\infty} \|y(t) - y_d(t)\| = 0$

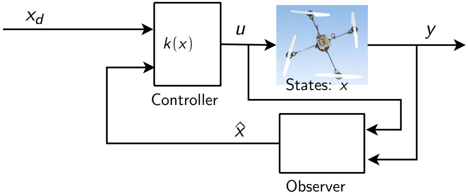
Better because closed loop control can stabilize systems and is robust w.r.t. perturbation.

In this course

Only closed-loop control problems are treated

Nonlinear dynamical systems

Aim of control ?



Tracking problem

find $u(x)$ such that $\lim_{t\to\infty} \|x(t) - x_d(t)\| = 0$

Stabilization problem

find $u(x)$ such that $\lim_{t\to\infty} \|x(t) - x_d(t)\| = 0$ for $x_d(t) = constant$

Null stabilization problem

find $u(x)$ such that $\lim_{t\to\infty} \|x(t) - x_d(t)\| = 0$ for $x_d(t) = 0$

In any cases, **a null stabilization problem of $z(t) = y(t) - y_d(t)$**

In this course

Only the **stabilization problem** will be treated.

Some strange behaviors of nonlinear systems

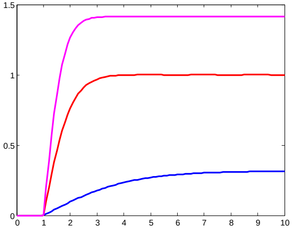
The undersea vehicle

- **Simplified model** of the undersea vehicle (in one direction):

$$\dot{v}(t) = -v\,|v| + u$$

- **Step answer**:

No proportionality between the input and the output



$u(s) = \frac{2}{s}$

$u(s) = \frac{1}{s}$

$u(s) = \frac{1}{100s}$

Some strange behaviors of nonlinear systems

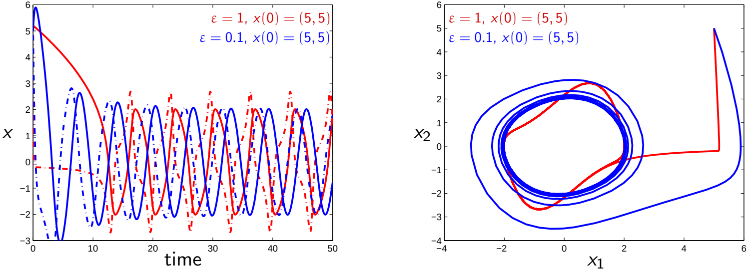
The Van der Pol oscillator (1/2)

- **Van der Pol oscillator**

$$\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - \varepsilon h(x_1)x_2 \end{cases}$$

with $h(x_1) = -1 + x_1^2$

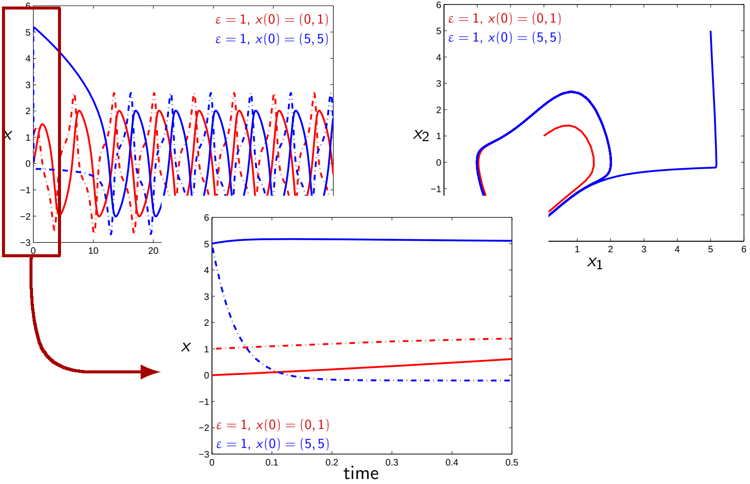
- **Oscillations:** ε tunes the limit cycle



Some strange behaviors of nonlinear systems

The Van der Pol oscillator (2/2)

- **Oscillations:** stable limit cycles, fast dynamics

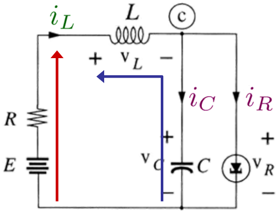


Some strange behaviors of nonlinear systems

The tunnel diode (1/2)

- **The tunnel diode model**

$$\begin{aligned} C\frac{dV_C}{dt} + i_R &= i_L \\ E - Ri_L &= V_C + L\frac{di_L}{dt} \\ i_R &= h(v_R) \end{aligned}$$



- It gives:

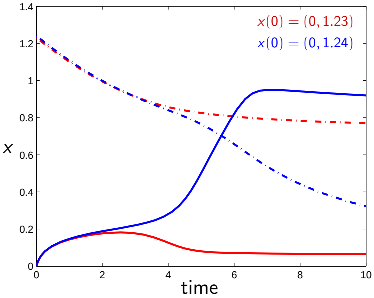
$$\begin{aligned} \dot{x}_1(t) &= \frac{1}{C}(-h(x_1) + x_2) \\ \dot{x}_2(t) &= \frac{1}{L}(-x_1 - Rx_2 + u) \end{aligned}$$

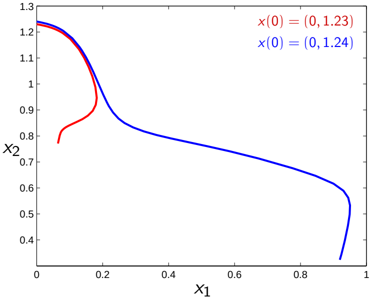
with $x_1 = v_C$, $x_2 = i_L$ and $u = E$

Some strange behaviors of nonlinear systems

The tunnel diode (2/2)

- **The tunnel diode behavior:** bifurcation
  with:
  $$i_R = 17.76v_R - 103.79v_R^2 + 229.62v_R^3 - 226.31v_R^4 + 83.72v_R^5$$





Some strange behaviors of nonlinear systems

The car parking

- **The car parking simplified model**

$$
\begin{aligned}
\dot{x}_1 &= u_1 \\
\dot{x}_2 &= u_2 \\
\dot{x}_3 &= x_2 u_1
\end{aligned}
$$

- System with a **misleading simplicity**

Theorem (Brockett (83))

There is no $(u_1(x), u_2(x))$ continuous w.r.t. x such that x is asymptotically stable

- **In practice:**
  - manoeuvres
  - discontinuous control
  - time-varying control

Marchand and Alamir (2003)

Nonlinear dynamical systems

Everything is possible

$$\begin{cases} \dot{x} = f(x, u) \\ y = h(x, u) \end{cases}$$

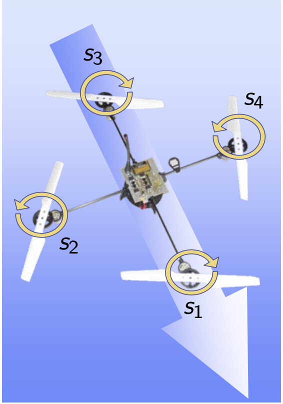
- **Everything is possible**:
  - **Equilibrium** can be unique, multiple, infinite or even not exist
  - **Controllability** (resp. **observability**) are very hard to prove (it is often even not checked)
  - **Stability** may be hard to prove
  - **Local properties** $\neq$ **global properties** (like stability, stabilizability, etc.)
  - The **time behavior** is **depends upon the initial condition**
  - **Frequency analysis** is almost **impossible**
  - **No systematic approach for building a control law**: to each problem corresponds it unique solution
  - ...
- **Mathematical tool**: Lyapunov and differential tools

Outline

The X4 helicopter

How it works ?

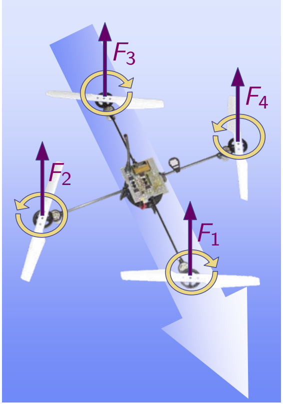
- 4 fixed rotors with controlled rotation speed s_i



The X4 helicopter

How it works ?

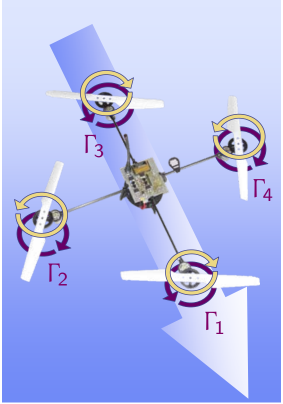
- 4 fixed rotors with controlled rotation speed s_i
- 4 generated forces F_i



The X4 helicopter

How it works ?

- 4 fixed rotors with controlled rotation speed s_i
- 4 generated forces F_i
- 4 counter-rotating torques Γ_i

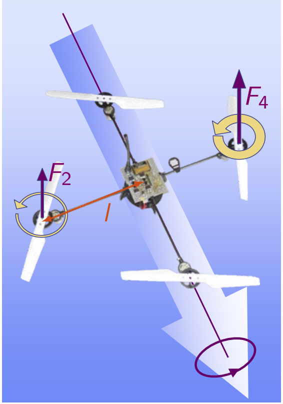


The X4 helicopter

How it works ?

- 4 fixed rotors with controlled rotation speed s_i
- 4 generated forces F_i
- 4 counter-rotating torques Γ_i
- **Roll movement** generated with a dissymmetry between left and right forces:

$$\Gamma_r = l(F_4 - F_2)$$



The X4 helicopter

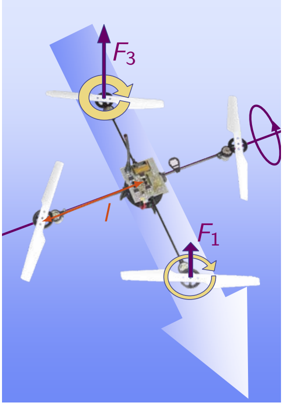
How it works ?

- 4 fixed rotors with controlled rotation speed s_i
- 4 generated forces F_i
- 4 counter-rotating torques Γ_i
- **Roll movement** generated with a dissymmetry between left and right forces:

$$\Gamma_r = l(F_4 - F_2)$$

- **Pitch movement** generated with a dissymmetry between front and rear forces:

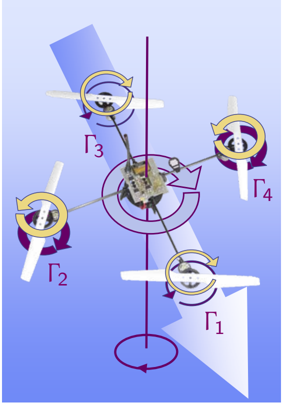
$$\Gamma_p = l(F_1 - F_3)$$



The X4 helicopter

How it works ?

- 4 fixed rotors with controlled rotation speed s_i
- 4 generated forces F_i
- 4 counter-rotating torques Γ_i
- **Roll movement** generated with a dissymmetry between left and right forces:
$$\Gamma_r = l(F_4 - F_2)$$
- **Pitch movement** generated with a dissymmetry between front and rear forces:
$$\Gamma_p = l(F_1 - F_3)$$
- **Yaw movement** generated with a dissymmetry between front/rear and left/right torques:
$$\Gamma_y = \Gamma_1 + \Gamma_3 - \Gamma_2 - \Gamma_4$$



The X4 helicopter

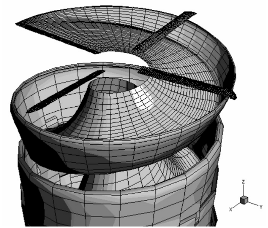
Building a model (1/3)

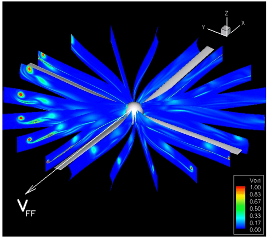
- **Electrical motor**: A 2nd order system with friction and saturation usually *approximated* by a 1rst order system:

$$\dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{1}{J_r}\tau_{\text{load}} + \frac{k_m}{J_r R}\,\mathrm{sat}_{\bar{U}_i}(U_i) \quad i \in \{1,2,3,4\} \quad (1)$$

s_i: rotation speed
U_i: voltage applied to the motor; real control variable
τ_{load}: motor load: $\tau_{\text{load}} = k_{gearbox}\kappa |s_i| s_i$ with κ drag coefficient

- **Aerodynamical forces and torques**: Very complex models exist





The X4 helicopter

Building a model (1/3)

- **Electrical motor**: A 2nd order system with friction and saturation usually *approximated* by a 1rst order system:

$$\dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{1}{J_r}\tau_{\text{load}} + \frac{k_m}{J_r R}\,\text{sat}_{\bar{U}_i}(U_i) \quad i \in \{1,2,3,4\} \quad (1)$$

s_i: rotation speed
U_i: voltage applied to the motor; real control variable
τ_{load}: motor load: $\tau_{\text{load}} = k_{gearbox}\kappa |s_i| s_i$ with κ drag coefficient

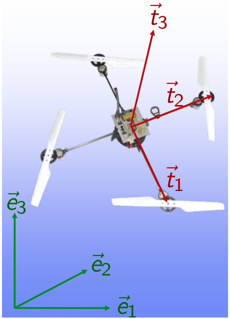
- **Aerodynamical forces and torques**: Very complex models exist but overcomplicated for control, better use the *simplified* model:

$$\begin{aligned} F_i &= b s_i^2 \\ \Gamma_r &= lb(s_4^2 - s_2^2) \\ \Gamma_p &= lb(s_1^2 - s_3^2) \\ \Gamma_y &= \kappa(s_1^2 + s_3^2 - s_2^2 - s_4^2) \end{aligned} \qquad i \in \{1,2,3,4\} \quad (2)$$

b: thrust coefficient, κ: drag coefficient

The X4 helicopter

Building a model (2/3)



- Two frames
  - a fixed frame $E(\vec{e}_1, \vec{e}_2, \vec{e}_3)$
  - a frame attached to the X4 $T(\vec{t}_1, \vec{t}_2, \vec{t}_3)$
- Frame change
  - a rotation matrix R from T to E
- State variables:
  - Cartesian coordinates (in E)
    - position $\vec{p}$
    - velocity $\vec{v}$
  - Attitude coordinates:
    - angular velocity $\vec{\omega}$ in the moving frame T
    - *either:* Euler angles three successive rotations about $\vec{t}_3$, $\vec{t}_1$ and $\vec{t}_3$ of angles angles ϕ, θ and ψ giving R
    - *or:* Quaternion representation $(q_0, \vec{q}) = (\cos\beta/2, \vec{u}\sin\beta/2)$ represent a rotation of angle β about $\vec{u}$

The X4 helicopter

Building a model (3/3)

- **Cartesian coordinates**:

$$\begin{cases} \dot{\vec{p}} = \vec{v} \\ m\dot{\vec{v}} = -mg\vec{e}_3 + R(\sum_i F_i(s_i)\vec{t}_3) \end{cases} \quad (3)$$

- **Attitude**:
  - Euler angles formalism:

$$\begin{cases} \dot{R} = R\vec{\omega}^\times \\ J\dot{\vec{\omega}} = -\vec{\omega}^\times J\vec{\omega} + \Gamma_{tot} \end{cases} \quad \text{with } \vec{\omega}^\times = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \quad (4)$$

  $\vec{\omega}^\times$ is the skew symmetric tensor associated to $\vec{\omega}$

  - Quaternion formalism:

$$\begin{cases} \dot{\vec{q}} = \frac{1}{2}\Omega(\vec{\omega})q \\ \quad = \frac{1}{2}\Xi(q)\vec{\omega} \\ J\dot{\vec{\omega}} = -\vec{\omega}^\times J\vec{\omega} + \Gamma_{tot} \end{cases} \quad \text{with } \begin{cases} \Omega(\vec{\omega}) = \begin{pmatrix} 0 & -\vec{\omega}^T \\ \vec{\omega} & -\vec{\omega}^\times \end{pmatrix} \\ \Xi(q) = \begin{pmatrix} -\vec{q}^T \\ I_{3\times 3}q_0 + \vec{q}^\times \end{pmatrix} \end{cases} \quad (5)$$

where $\Gamma_{tot} = \underbrace{-\sum_i I_r \vec{\omega}^\times \vec{t}_3 s_i}_{\text{gyroscopic torque}} + \Gamma_{pert} + \begin{pmatrix} \Gamma_r(s_2, s_4) \\ \Gamma_p(s_1, s_3) \\ \Gamma_y(s_1, s_2, s_3, s_4) \end{pmatrix}$

The X4 helicopter

Review of the nonlinearities

$$
\begin{cases}
\dot{s}_i &= -\frac{k_m^2}{J_r R} s_i - \frac{k_{gearbox}\,\kappa}{J_r} |s_i|\, s_i + \frac{k_m}{J_r R} \operatorname{sat}_{\bar{U}_i}(U_i) \\
\dot{\vec{p}} &= \vec{v} \\
m\dot{\vec{v}} &= -mg\,\vec{e}_3 + R \begin{pmatrix} 0 \\ 0 \\ \sum_i F_i(s_i) \end{pmatrix} \\
\dot{R} &= R\vec{\omega}^{\times} \\
J\dot{\vec{\omega}} &= -\vec{\omega}^{\times} J\vec{\omega} - \sum_i I_r \vec{\omega}^{\times} \begin{pmatrix} 0 \\ 0 \\ \sum_i s_i \end{pmatrix} + \begin{pmatrix} \Gamma_r(s_2, s_4) \\ \Gamma_p(s_1, s_3) \\ \Gamma_y(s_1, s_2, s_3, s_4) \end{pmatrix}
\end{cases}
$$

In red: the nonlinearities

In blue: where the control variables act

The X4 helicopter

Identification of the parameters

- **Electrical motor**:
  - For small input steps, the system behaves very close to a **linear** first order system
  - Hence, use linear identification tools
  - $\bar{U}_i$ is found on the data-sheet of the motor (damage avoidance)
- **Aerodynamical parameters**: b and κ

b and κ measured with specific test beds,

The X4 helicopter

Identification of the parameters

- **Electrical motor**:
  - For small input steps, the system behaves very close to a **linear** first order system
  - Hence, use linear identification tools
  - $\bar{U}_i$ is found on the data-sheet of the motor (damage avoidance)
- **Aerodynamical parameters**: b and κ

b and κ measured with specific test beds, depends upon temperature, distance from ground, etc.

- **Mechanical parameters**:

l length of an arm of the helicopter, easy to measure

m total mass of the helicopter, easy to measure

J body inertia, hard to have precisely

I_r rotor inertia, hard to have precisely

The X4 helicopter

Values of the parameters

- **Motor parameters**:

parameter	description	value	unit
k_m	motor constant	4.3×10^{-3}	N.m/A
J_r	rotor inertia	3.4×10^{-5}	$J.g.m^2$
R	motor resistance	0.67	Ω
$k_{gearbox}$	gearbox ratio	2.7×10^{-3}	-
$\bar{U}_i$	maximal voltage	12	V

- **Aerodynamical parameters**:

parameter	description	value
b	thrust coefficient	3.8×10^{-6}
κ	drag coefficient	2.9×10^{-5}

- **Body parameters**:

parameter	description	value	unit
J	inertia matrix	$\begin{pmatrix} 14.6 \times 10^{-3} & 0 & 0 \\ 0 & 7.8 \times 10^{-3} & 0 \\ 0 & 0 & 7.8 \times 10^{-3} \end{pmatrix}$	$kg.m^2$
m	mass of the UAV	0.458	kg
l	radius of the UAV	22.5	cm
g	gravity	9.81	m/s^2

The X4 helicopter

Open-loop behavior with roll initial speed

The X4 helicopter

Open-loop behavior with pitch initial speed

The X4 helicopter

Open-loop behavior with yaw initial speed

The X4 helicopter

Open-loop behavior with roll and yaw initial speed

The X4 helicopter

Open-loop behavior with pitch and yaw initial speed

Outline

Outline

Linear systems with saturated inputs

Impact of input saturations on the control of the X4's rotors (1/4)

- We go back to the **X4** example and focus on the **rotors**:

$$\left\{ \dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{1}{J_r}\tau_{\text{load}} + \frac{k_m}{J_r R}\,\mathrm{sat}_{\bar{U}_i}(U_i) \right.$$

- If one wants to **act on the X4 with desired forces** F_i^d, it is necessary to be able to **set the rotors speeds** s_i **to** s_i^d with

$$s_i^d = \sqrt{\frac{1}{b} F_i^d}$$

- A usual way to control the electrical motor consist in
  - taking τ_{load} **as un unknown load**
  - **neglecting the voltage limitations** $\bar{U}_i$

Linear systems with saturated inputs

Impact of input saturations on the control of the X4's rotors (2/4)

- The so obtained **system is linear**

$$\frac{s_i(s)}{U_i(s)} = \frac{\frac{1}{k_m}}{1+\frac{J_r R}{k_m^2}s} = \frac{G}{1+\tau s}$$

- Define a **PI controller** for it:

$$C(s) = K_p + \frac{K_i}{s}$$

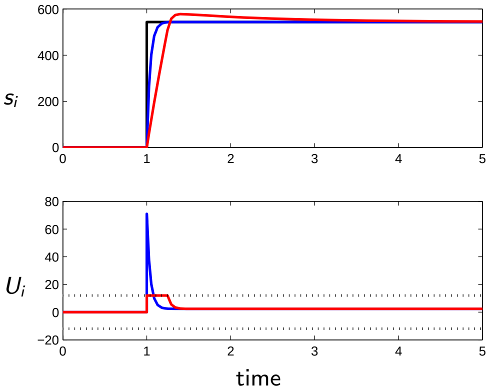
- Taking $K_i = \frac{1}{\tau_{CL} G}$ and $K_p = \tau K_i$, the closed loop system is:

$$\frac{s_i(s)}{U_i(s)} = \frac{1}{1+\tau_{CL} s}$$

Linear systems with saturated inputs

Impact of input saturations on the control of the X4's rotors (3/4)

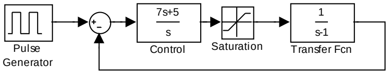
- Make a step that **compensates the weight**, that is such that $s_i^d = \sqrt{\frac{mg}{4b}}$ so that $\sum_i F_i^d = mg$
- Taking $\tau_{CL} = 50$ **ms**, one gets **with saturations**



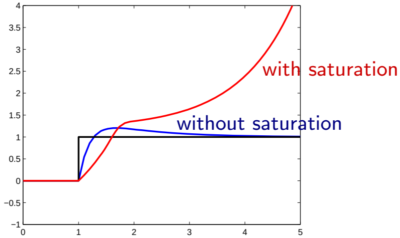
Linear systems with saturated inputs

Saturation may cause instability

- The result could be worse:



- For $u \in [-1.2, 1.2]$, the closed-loop behavior is:

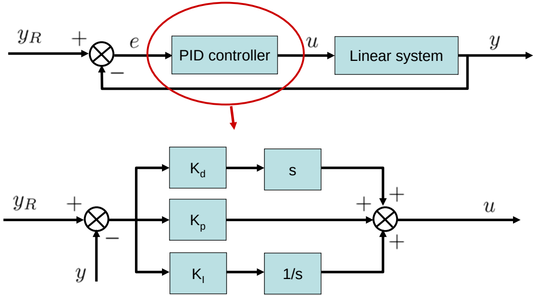


- **Saturations may lead to instability** especially in the presence of integrators in the loop

Linear systems with saturated inputs

Key idea of the anti-windup scheme

- Consider a **linear system with a PID** controller:

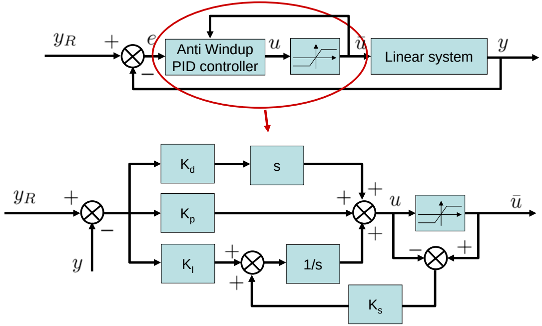


- The instability comes from the **integration** of the error
- **Key idea:** soften the integral effect when the control is saturated

Linear systems with saturated inputs

PID controller with anti-windup

- Structure of the **PID controller with anti-windup**:

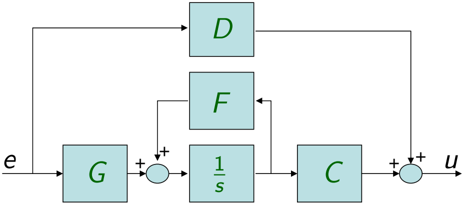


- If $u = \bar{u}$, that is if u is not saturated, **then the PID controller with anti-windup is identical to the classical PID controller**
- **If u is saturated** $(u \neq \bar{u})$, K_s **tunes the reduction of the integral effect** of the PID

Linear systems with saturated inputs

General controller with anti-windup

- Structure of the **general dynamic controller**:



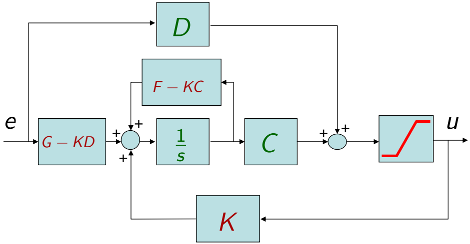
- with dynamics given by:

$$\begin{cases} \dot{x}_c &= Fx_c + Ge \\ u &= Cx_c + De \end{cases}$$

Linear systems with saturated inputs

General controller with anti-windup

- **Anti-windup** structure of the **general dynamic controller**:



- with dynamics given by:

$$\begin{aligned}\dot{x}_c &= Fx_c + Ge + K(u - Cx_c - De)\\ &= (F - KC)x_c + (G - KD)e + Ku\end{aligned}$$

Choice of the antiwindup parameters

Take K such that $(F - KC)$ is stable and $|\lambda_{\max}(F - KC)| > |\lambda_{\max}(F)|$

Linear systems with saturated inputs

A short Lyapunov explanation of the behavior

- Consider a **stable closed loop linear system**: it is globally asymptotically stable. A **saturation on the input** may:
  - **transform** the **global stability** into **local stability**. In this case, the aim of the anti-windup is to **increase the radius of attraction** of the closed loop system
  - **keep** the **global stability property**. In this case, the aim of the anti-windup is to **renders the saturated system closer to its unsaturated equivalent**
- However, there is **no formal proof of stability** of anti-windup strategies

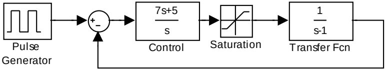
Other approaches for saturated inputs

- Optimal control
- Nested saturation function (under development)

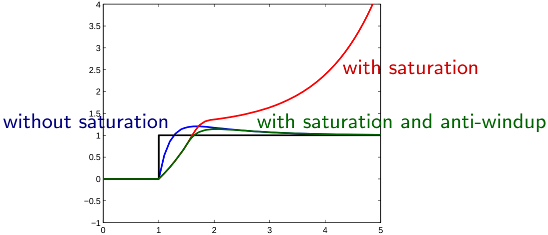
Linear systems with saturated inputs

Back to the unstable case

- The unstable case:



- The closed-loop behavior is:

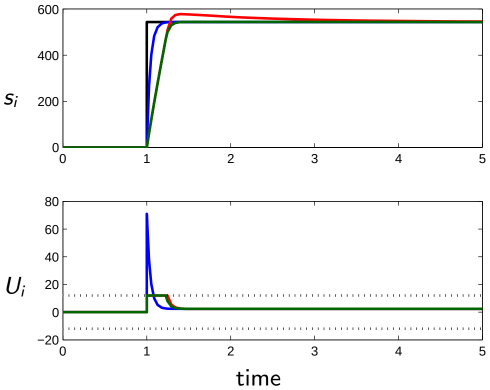


- However, nothing is magic and divergent behavior may occur because of the level of the step input

Linear systems with saturated inputs

Impact of input saturations on the control of the X4's rotors (4/4)

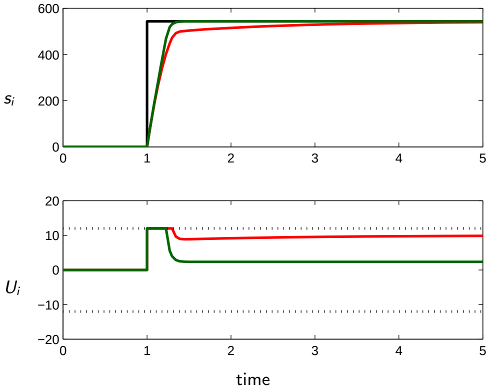
- Make a step that **compensates the weight**, that is such that $s_i^d = \sqrt{\frac{mg}{4b}}$ so that $\sum_i F_i^d = mg$
- Taking $\tau_{CL} = 50$ **ms**, one gets **with anti-windup**



Linearization of nonlinear systems

Impact of the load on the control of the X4's rotors

- Make a step that **compensates the weight**, that is such that $s_i^d = \sqrt{\frac{mg}{4b}}$ so that $\sum_i F_i^d = mg$
- Taking $\tau_{CL} = 50$ **ms**, one gets **with load**



- The PI controller seems badly tuned: for $t > 1.3\,s$, the control is not saturated but the convergence is very slow. **What's wrong ?**

Linearization of nonlinear systems

Impact of the load on the control of the X4's rotors

- Back to the **rotor dynamical equation**:

$$\dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{k_{gearbox} \kappa}{J_r} |s_i| \, s_i + \frac{k_m}{J_r R} \operatorname{sat}_{\bar{U}_i}(U_i)$$

- Taking τ_{load} as unknown implies that the **PI control law was tuned for**:

$$\dot{s}_i = -\frac{k_m^2}{J_r R} s_i + \frac{k_m}{J_r R} \operatorname{sat}_{\bar{U}_i}(U_i)$$

- Implicitly, the **second order terms were neglected**
- Unfortunately, these two systems behave similarly **iff s_i is small**, which is **not the case for $s_i^d = \sqrt{\frac{mg}{4b}} \approx 544$ rad s^{-1}**

Linearization of nonlinear systems

Linearization at the origin

Linearization at the origin

Take a nonlinear system

$$\begin{cases} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{cases}$$

with $f(0,0) = 0$. Then, near the origin, it can be approximated by its linear Taylor expansion at the first order:

$$\begin{cases} \dot{x} &= \underbrace{\left.\frac{\partial f}{\partial x}\right|_{(x=0,u=0)}}_{A} x + \underbrace{\left.\frac{\partial f}{\partial u}\right|_{(x=0,u=0)}}_{B} u \\ y - h(0,0) &= \underbrace{\left.\frac{\partial h}{\partial x}\right|_{(x=0,u=0)}}_{C} x + \underbrace{\left.\frac{\partial h}{\partial u}\right|_{(x=0,u=0)}}_{D} u \end{cases}$$

Linearization at a point

Take a nonlinear system

$$\begin{cases} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{cases}$$

then, near the equilibrium point (x_0, u_0), it can be approximated by its linear Taylor expansion at the first order:

$$\begin{cases} \underbrace{\dot{x}}_{\dot{\tilde{x}}} &= \underbrace{\left.\frac{\partial f}{\partial x}\right|_{(x_0,u_0)}}_{A} \underbrace{(x-x_0)}_{\tilde{x}} + \underbrace{\left.\frac{\partial f}{\partial u}\right|_{(x_0,u_0)}}_{B} \underbrace{(u-u_0)}_{\tilde{u}} \\ y - h(x_0,u_0) &= \underbrace{\left.\frac{\partial h}{\partial x}\right|_{(x_0,u_0)}}_{C} \underbrace{(x-x_0)}_{\tilde{x}} + \underbrace{\left.\frac{\partial h}{\partial u}\right|_{(x_0,u_0)}}_{D} \underbrace{(u-u_0)}_{\tilde{u}} \end{cases}$$

which is linear in the variables $\tilde{x} = x - x_0$ and $\tilde{u} = u - u_0$

Linearization of nonlinear systems

Some properties of the linearization

Controllability properties

Take a nonlinear system

$$\dot{x} = f(x, u) \tag{6}$$

and its linearization at (x_0, u_0):

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u} \tag{7}$$

- If (7) is controllable then (6) is locally controllable
- Nothing can be concluded if (7) is uncontrollable

Example

The car is controllable but not its linearization

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \\ \dot{x}_3 &= x_2 u_1 \end{aligned} \quad \underset{\text{linearization at the origin}}{\Rightarrow} \quad \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} u_1 & u_2 \end{pmatrix}$$

Linearization of nonlinear systems

Some properties of the linearization

Stability properties

Take a nonlinear system

$$\dot{x} = f(x, u) \tag{8}$$

and its linearization at (x_0, u_0):

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u} \tag{9}$$

Assume that one can build a feedback law $\tilde{u} = k(\tilde{x})$ so that:

- (9) is asymptotically stable ($\mathfrak{R} < 0$): then (8) is locally asymptotically stable with $u = u_0 + k(\tilde{x})$
- (9) is unstable ($\mathfrak{R} > 0$): then (8) is locally unstable with $u = u_0 + k(\tilde{x})$
- Nothing can be concluded if (9) is simply stable ($\mathfrak{R} \leq 0$)

Linearization of nonlinear systems

Impact of the load on the control of the X4's rotors

- Back to the **rotor dynamical equation**:

$$\dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{k_{gearbox}\kappa}{J_r} |s_i| \, s_i + \frac{k_m}{J_r R} \mathrm{sat}_{\bar{U}_i}(U_i)$$

- Define U_i^d, the constant control that keeps the steady state speed:

$$U_i^d = k_m s_i^d + \frac{R k_{gearbox}\kappa}{k_m} \left| s_i^d \right| s_i^d$$

- The **linearization of the rotor dynamics** near s_i^d gives:

$$\dot{\tilde{s}}_i = -\left(\frac{k_m^2}{J_r R} + \frac{2 k_{gearbox}\kappa}{J_r} \left| s_i^d \right| \right) \tilde{s}_i + \frac{k_m}{J_r R} \tilde{\mathrm{sat}}_{\bar{U}_i}(\tilde{U}_i)$$

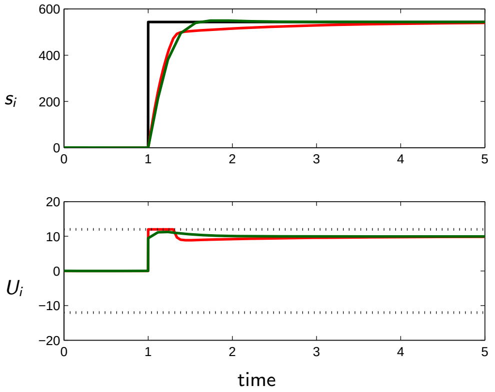
with:

$$\tilde{U}_i = U_i - U_i^d$$
$$\tilde{s}_i = s_i - s_i^d$$
$$\tilde{\mathrm{sat}}_{\bar{U}_i}(\tilde{U}_i) = \begin{cases} \tilde{U}_i & \text{if } U_i \in [-\bar{U}_i, +\bar{U}_i] \\ \bar{U}_i & \text{if } U_i \geq \bar{U}_i \\ -\bar{U}_i & \text{if } U_i \leq -\bar{U}_i \end{cases}$$

Linearization of nonlinear systems

Impact of the load on the control of the X4's rotors

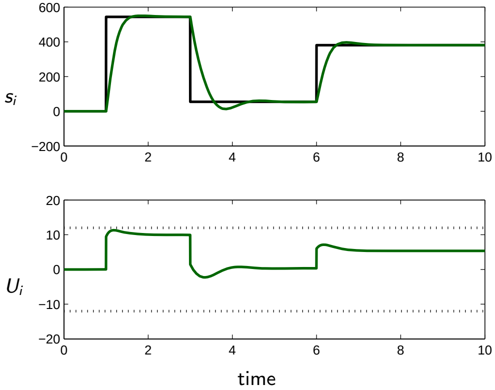
- Make a step that **compensates the weight**, that is such that $s_i^d = \sqrt{\frac{mg}{4b}}$ so that $\sum_i F_i^d = mg$
- Taking $\tau_{CL} = 50$ **ms** and a **PI controller tuned at** s_i^d on the system with load, one has:



Towards gain scheduling

Impact of a change in the steady state speed of the X4's rotors

- Take again $\tau_{CL} = 50$ **ms** and a **PI controller tuned at** s_i^d
- Make speed steps of different level



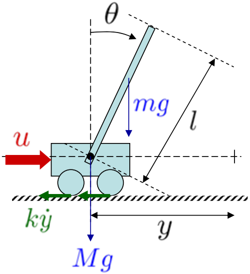
- The controller is well tuned near s_i^d but **not very good a large range of use**

Towards gain scheduling

Local linearization may be inadequate for large excursion

The result could be worse:

- Take the **inverted pendulum**



$$\begin{pmatrix}\ddot\theta\\ \ddot y\end{pmatrix} = \frac{1}{\Delta(\theta)}\begin{pmatrix} m+M & -ml\cos\theta \\ -ml\cos\theta & I+ml^2\end{pmatrix}\begin{pmatrix} mgl\sin\theta \\ u+ml\dot\theta^2\sin\theta - k\dot y\end{pmatrix}$$

where

- I is the inertia of the pendulum
- $\Delta(\theta) = (I+ml^2)(m+M) - m^2l^2\cos^2\theta$

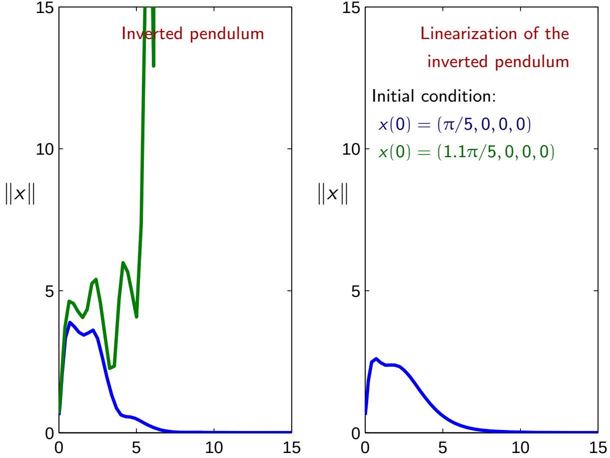
- Linearize it near the upper position ($\theta = 0$) with $m = M = I = g = 1$, $k = 0$ and $x = (\theta, \dot\theta, y, \dot y)$:

$$\begin{pmatrix}\dot x_1\\ \dot x_2\\ \dot x_3\\ \dot x_4\end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0\\ \frac{2}{3} & 0 & 0 & 0\\ 0 & 0 & 0 & 1\\ -\frac{1}{3} & 0 & 0 & 0\end{pmatrix}\begin{pmatrix} x_1\\ x_2\\ x_3\\ x_4\end{pmatrix} + \begin{pmatrix} 0\\ -\frac{1}{3}\\ 0\\ \frac{2}{3}\end{pmatrix} u$$

Towards gain scheduling

Local linearization may be inadequate for large excursion

- Build a linear feedback law that places the poles at -1
- Starting from $\theta(0) = \pi/5$, it converges, from $\theta(0) = 1.1 \times \pi/5$, it diverges



Towards gain scheduling

- For some systems, **the radius of attraction** of a stabilized linearization **may be very small** (less than one degree on the double inverted pendulum for instance)

➜ **Linearization is not suitable for large range of use of the closed-loop system**

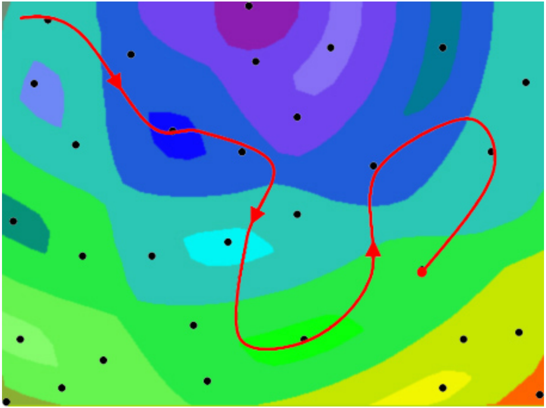
- For other systems, the controller **must be very finely tuned** in order to meet performance requirements (e.g. automotive with more and more restrictive pollution standards)

➜ **Linearization is not suitable for high accuracy closed loop systems**

- Either for **stability reasons** or **performance reasons**, it may be necessary to **tune the controller at more than one operating point**

Towards gain scheduling

- **Gain scheduling** uses the idea of **using a collection of linearization of a nonlinear system at different operating points**



Gain scheduling

- Take a **nonlinear system**

$$\dot{x} = f(x, u)$$

- Define a family of equilibrium points

$$s = (x_{eq}, u_{eq}) \qquad \text{such that } f(x_{eq}, u_{eq}) = 0$$

- **At each point s, linearize the system** assuming s is constant:

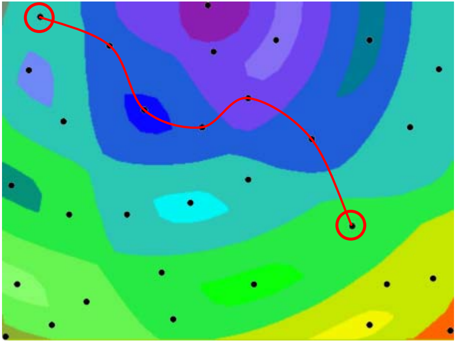
$$\dot{\tilde{x}} = A(s)\tilde{x} + B(s)\tilde{u}$$
$$\text{with } \tilde{x} = x - x_{eq} \text{ and } \tilde{u} = u - u_{eq}$$

- **At each point s, define a feedback law**:

$$u = u_{eq}(s) - K(s)(x - x_{eq}(s))$$
$$\text{with } \Re(\text{Eig}(A(s) - B(s)K(s))) < 0$$

Gain scheduling

- **The controller is then obtained** by changing s in the a priori defined collection of points
- The change of s can be done discontinuously or continuously by interpolation



Gain scheduling

- **Drawbacks** of gain scheduling :
  - ✗ Convergence only if s varies slowly
  - ✗ The performance within a linearization area may be poor
  - ✗ The mapping may be prohibitive if the number of states is large
  - ✗ The stability issues are not clear
- **Recent evolution of gain scheduling**:
  - **Dynamic gain scheduling**: improve the transition between the different controllers, limits the importance of the slow motion of s
  - **LPV**: Linear parameter varying methods considers the nonlinear system as

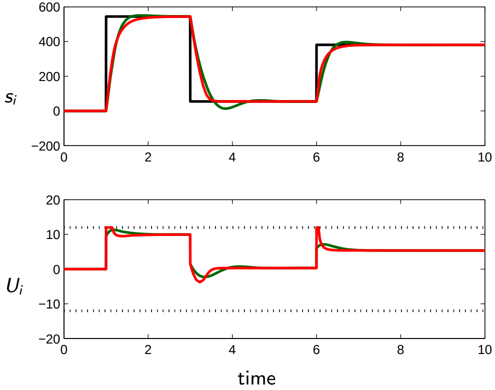
$$\dot{x} = A(\rho(x))x + B(\rho(x))u$$

    and, under some conditions, uses LMI to build a feedback. Improves the performance within a linearization area.

Gain scheduling

Handling changes in the steady state speed of the X4's rotors

- Take again $\tau_{CL} = 50$ **ms** and a **PI controller tuned at** s_i
- Make speed steps of different level



- **The rotors are now well controlled**

Stability

Stability

Consider the autonomous nonlinear system:

$$\dot{x} = f(x, u(x)) = g(x) \quad (10)$$

Stability

System (10) is said to be **stable** at the origin iff:
$\forall R > 0$, $\exists r(R) > 0$ such that $\forall x_0 \in \mathcal{B}(r(R))$, $x(t; x_0)$, solution of (10) with x_0 as initial condition, remains in $\mathcal{B}(R)$ for all $t > 0$.

Attractivity

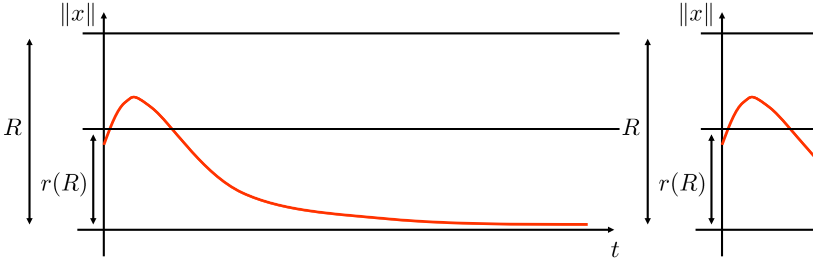
The origin is said to be **attractive** iff:
$\lim_{t\to\infty} x(t; x_0) = 0$.

Asymptotic stability

System (10) is said to be **asymptotically stable** at the origin iff it is stable and attractive

Stability

Graphical interpretation



- For linear systems: Attractivity $\rightarrow$ Stability
- For nonlinear systems: Attractivity $\nrightarrow$ Stability
- Stability and attractivity : properties hard to check ?

Stability

Using the linearization

Asymptotic stability and local linearizaton

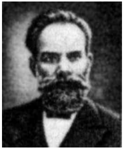
Consider $\dot{x} = g(x)$ and its linearization at the origin $\dot{x} = \left.\frac{\partial g}{\partial x}\right|_{x=0} x$. Then:

- Linearization with eig$< 0 \rightleftarrows$ Nonlinear system is locally asymptotically stable
- Linearization with eig$> 0 \rightleftarrows$ Nonlinear system is locally unstable
- Linearization with eig$= 0$: nothing can be concluded on the nonlinear system (may be stable or unstable)

Only local conclusions

Stability

Lyapunov theory: Lyapunov functions



Aleksandr Mikhailovich **Lyapunov**

Markov's school friend, Chebyshev's student

Master Thesis : On the stability of ellipsoidal forms of equilibrium of a rotating liquid in 1884

Phd Thesis : The general problem of the stability of motion in 1892

6 June 1857 - 3 Nov 1918

Definition: Lyapunov function

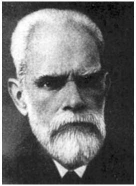
Let $V : \mathbb{R}^n \to \mathbb{R}$ be a continuous function such that:

1. (definite) $V(x) = 0 \Leftrightarrow x = 0$
2. (positive) $\forall x,\ V(x) \geq 0$
3. (radially unbounded) $\lim_{\|x\| \to \infty} V(x) = +\infty$

Lyapunov functions are energies

Stability

Lyapunov theory: Lyapunov theorem



Aleksandr Mikhailovich **Lyapunov**
Markov's school friend, Chebyshev's student
Master Thesis : On the stability of ellipsoidal forms of equilibrium of a rotating liquid in 1884
Phd Thesis : The general problem of the stability of motion in 1892
6 June 1857 - 3 Nov 1918

Theorem: (First) Lyapunov theorem

If $\exists$ a Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}^+$ s.t.

- (strictly decreasing) $V(x(t))$ is **strictly** decreasing for all $x(0) \neq 0$

then the origin is asymptotically stable.

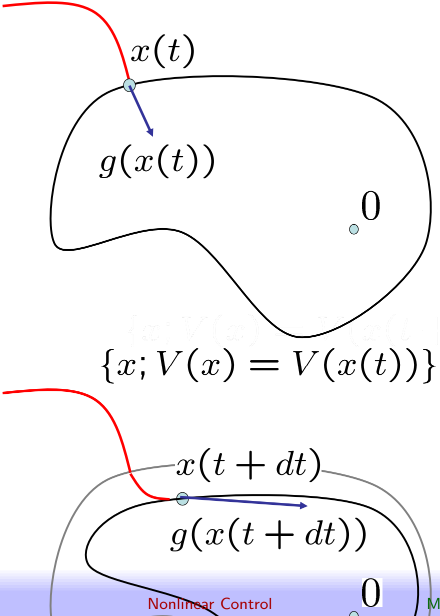
If $\exists$ a Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}^+$ s.t.

- (decreasing) $V(x(t))$ is decreasing

then the origin is stable.

Stability

Lyapunov theorem: a graphical interpretation



Stability

Stability and robustness

- Stability holds robustness:

$$\begin{aligned} \dot{x} = g(x) \text{ stable} \quad &\Rightarrow \quad \dot{V} = \frac{\partial V}{\partial x} g(x) < 0 \\ &\Rightarrow \quad \frac{\partial V}{\partial x} \left(g(x) + \varepsilon(x)\right) < 0 \\ &\Rightarrow \quad \dot{x} = g(x) + \varepsilon(x) \text{ stable} \end{aligned}$$

Outline

The X4 helicopter

The control problems

$$
\begin{cases}
\dot{s}_i = -\frac{k_m^2}{J_r R} s_i - \frac{k_{gearbox}\kappa}{J_r} |s_i| s_i + \frac{k_m}{J_r R} \operatorname{sat}_{\bar{U}_i}(U_i) \\
\dot{\vec{p}} = \vec{v} \\
m\dot{\vec{v}} = -mg\vec{e}_3 + R \begin{pmatrix} 0 \\ 0 \\ \sum_i F_i(s_i) \end{pmatrix} \\
\dot{R} = R\vec{\omega}^\times \\
J\dot{\vec{\omega}} = -\vec{\omega}^\times J\vec{\omega} - \sum_i I_r \vec{\omega}^\times \begin{pmatrix} 0 \\ 0 \\ \sum_i s_i \end{pmatrix} + \begin{pmatrix} \Gamma_r(s_2, s_4) \\ \Gamma_p(s_1, s_3) \\ \Gamma_y(s_1, s_2, s_3, s_4) \end{pmatrix}
\end{cases}
$$

Position control problem (last four equations)

Attitude control problem (last two equations)

Outline

Control Lyapunov functions

Characterizing property (1/3)

- Control Lyapunov functions (CLF) where introduced in the 80's
- CLF are for controllability and control what Lyapunov functions are for stability
- **Lyapunov**: a tool for **stability analysis of autonomous systems**
  - Take

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - x_2 \end{aligned}$$

  - Take the Lyapunov function

$$V(x) = \frac{3}{2}x_1^2 + x_1 x_2 + x_2^2$$

  - Along the trajectories of the system:

$$\dot{V}(x(t)) = \nabla V(x).f(x) = -\|x\|^2$$

hence, $V \searrow 0$ and the system is asymptotically stable

Control Lyapunov functions

Characterizing property (2/3)

- **Control Lyapunov Functions**: a tool for **stabilization of controlled systems**
  - Take now the **controlled system**

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 + u \end{aligned}$$

  - Take the Lyapunov function

$$V(x) = \frac{3}{2}x_1^2 + x_1 x_2 + x_2^2$$

  - Along the trajectories of the system:

$$\begin{aligned} \dot{V}(x(t), u(t)) &= \nabla V(x).f(x, u) \\ &= -x_1^2 + x_1 x_2 + x_2^2 + (x_1 + 2x_2)u \end{aligned}$$

Control Lyapunov functions

Characterizing property (3/3)

- **Control Lyapunov Functions**: a tool for **stabilization of controlled systems**
  - Along the trajectories of the system:

$$\begin{aligned}\dot{V}(x(t), u(t)) &= \nabla V(x).f(x,u) \\ &= -x_1^2 + x_1 x_2 + x_2^2 + (x_1 + 2x_2)u\end{aligned}$$

  - hence
    - if $x_1 + 2x_2 \neq 0$, there exists u such that $V \searrow$
    - otherwise (if $x_1 + 2x_2 = 0$)

$$\dot{V}(x(t), u(t)) = -4x_2^2 - 2x_2^2 + x_2^2 = -5x_2^2$$

which is negative unless $x_2 = 0 = -\frac{1}{2}x_1$: $V \searrow$

In conclusion: characterizing property of CLF

For any $x \neq 0$, there exists u such that $\dot{V}(x(t), u(t)) < 0$

Control Lyapunov functions

Formal definitions

Definition: Lyapunov function

A continuous function $V : \mathbb{R}^n \longrightarrow \mathbb{R}_{\geq 0}$ is called a Lyapunov function if

1. [positive definiteness] $V(x) \geq 0$ for all x and $V(0) = 0$ if and only if $x = 0$
2. [radially unbounded] $\lim_{x\to\infty} V(x) = +\infty$

or:

1. [positive definiteness] $V(x) \geq 0$ for all x and $V(0) = 0$ if and only if $x = 0$
2. [proper] for each $a \geq 0$, the set $\{x|V(x) \leq a\}$ is compact

or, alternatively:

$$(\exists \underline{\alpha}, \overline{\alpha} \in \mathcal{K}_\infty) \quad \underline{\alpha}(\|x\|) \leq V(x) \leq \overline{\alpha}(\|x\|) \quad \forall x \in \mathbb{R}^n$$

Control Lyapunov functions

Formal definitions

Definition: Control Lyapunov Function

A differentiable control Lyapunov function denotes a differentiable Lyapunov function being *infinitesimally decreasing*, meaning that there exists a positive continuous definite function $W : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and such that:

$$\sup_{x\in\mathbb{R}^n} \min_{u\in\mathbb{R}^m} \nabla V(x) f(x,u) + W(x) \leq 0$$

- Roughly speaking a CLF is a Lyapunov function that one can force to decrease
- Extensions to nonsmooth CLF exist and are necessary for the class of system that can not be stabilized by means of smooth static feedback

Control Lyapunov functions

Control affine systems

Definition: Directional Lie derivative

For any f and g with values in appropriate sets, the Lie derivative of g along f is defined by:

$$L_f g(x) := \nabla g(x) \cdot f(x)$$

Iteratively, one defines:

$$L_f^k g(x) := L_f L_f^{k-1} g(x)$$

Theorem

Consider a control affine system $\dot{x} = g_0(x) + \sum_{i=1}^m u_i g_i(x)$ with f smooth and $f(0) = 0$. Assume that the set

$$\mathcal{S} = \left\{ x | L_f V(x) = 0 \text{ and } L_f^k L_{g_i} V(x) = 0 \text{ for all } k \in \mathbb{N},\ i \in \{1, \dots, m\} \right\} = \{0\}$$

then, the feedback

$$k(x) := -\left(\nabla V(x) \cdot G(x)\right)^T = \begin{pmatrix} L_{g_1} V(x) & \cdots & L_{g_m} V(x) \end{pmatrix}^T$$

globally asymptotically stabilizes the system at the origin.

Control Lyapunov functions

Control affine systems

Theorem: Sontag's universal formula (1989)

Consider a control affine system $\dot{x} = g_0(x) + \sum_{i=1}^{m} u_i g_i(x)$ with f smooth and $f(0) = 0$. Assume that there exists a differentiable CLF, then the feedback

$$k_i(x) = -b_i(x)\varphi(a(x), \beta(x)) \quad (= 0 \text{ for } x = 0)$$

- $a(x) := \nabla V(x) f(x)$ and $b_i(x) := \nabla V(x) g_i(x)$ for $i = 1, \dots, m$
- $B(x) = \begin{pmatrix} b_1(x) & \cdots & b_2(x) \end{pmatrix}$ and $\beta(x) = \|B(x)\|^2$
- $q : \mathbb{R} \to \mathbb{R}$ is any real analytic function with $q(0) = 0$ and $bq(b) > 0$ for any $b > 0$
- φ is the real analytic function:

$$\varphi(a, b) = \begin{cases} \frac{a + \sqrt{a^2 + bq(b)}}{b} & \text{if } b \neq 0 \\ 0 & \text{if } b = 0 \end{cases}$$

is such that $k(0) = 0$, k smooth on $\mathbb{R}^n \backslash 0$ and

$$\sup_{x \in \mathbb{R}^n} \nabla V(x) f(x, k(x)) + W(x) \leq 0$$

Control Lyapunov functions

Control affine systems

Definition: Small control property

A CLF V is said to satisfies the *small control property* if for any ε, there is some δ so that:

$$\sup_{x \in \mathcal{B}(\delta)} \min_{u \in \mathcal{B}(\varepsilon)} \nabla V(x) f(x,u) + W(x) \leq 0$$

The small control property implicitly means that if ε is small (hence the control), the system can still be controlled as long as it is sufficiently close to the origin

Theorem: Sontag's universal formula (1989)

Consider an control-affine system $\dot{x} = g_0(x) + \sum i = 1^m u_i g_i(x)$ with f smooth and $f(0) = 0$. Assume that there exists a differentiable CLF, then the previous feedback k with $q(b) = b$ is smooth on $\mathbb{R}^n \backslash 0$ and **continuous at the origin**

Control Lyapunov functions

Robustness issues

Model

$$\begin{cases} \dot{x} &= f(x,u) \\ u &= k(x) \end{cases}$$

Real system

$$\begin{cases} \dot{x} &= f(x,u)+\varepsilon \\ u &= k(x+\delta) \end{cases}$$

Definition: Robustness w.r.t. model errors

A feedback law $k:\mathbb{R}^n \to \mathbb{R}^m$ is said to be *robust with respect to model errors* if $\lim_{t\to\infty} x(t) \in \mathcal{B}(r(\varepsilon))$ with $\lim_{\varepsilon\to 0} r(\varepsilon) = 0$ ($x(t)$ denotes the solution of $\dot{x} = f(x, k(x)) + \varepsilon$)

Definition: Robustness w.r.t. measurement errors

A feedback law $k:\mathbb{R}^n \to \mathbb{R}^m$ is said to be *robust with respect to measurement errors* if $\lim_{t\to\infty} x(t) \in \mathcal{B}(r(\delta))$ with $\lim_{\delta\to 0} r(\delta) = 0$ ($x(t)$ denotes the solution of $\dot{x} = f(x, k(x+\delta))$)

Theorem: Smooth CLF = robust stability (1999)

The existence of a feedback law robust to measurement errors and model errors is equivalent to the existence of a smooth CLF

Control Lyapunov functions

Conclusion

- **A CLF is a theoretical tool**, not a magical tool
- A CLF is **not a constructive tool**
- Finding a stabilizing control law and finding a CLF are at best the same problems. In all other cases, finding a stabilizing control law is easier than finding a CLF
- Even if one can find a CLF for a system, the universal formula often gives a feedback with poor performances
- However, this is an important field of research in the control system theory community that proved important theoretical results, for instance the equivalence between asymptotic controllability and stabilizability of nonlinear systems (1997)

Other structural tools

Passivity

Definition: Passivity

A system

$$\begin{cases} \dot{x} = f(x,u) \\ y = h(x,u) \end{cases}$$

is said to be passive if there exists some function $S(x) \geq 0$ with $S(0) = 0$ such that

$$S(x(T)) - S(x(0)) \leq \int_0^T u^T(\tau) y(\tau) d\tau$$

- Roughly speaking, $S(x(0))$ (called the *storage function*) denotes the largest amount of energy which can be extracted from the system given the initial condition $x(0)$.
- Passivity eases the design of control law and is a powerful tool for interconnected systems
- Important literature

Other structural tools

Input-to-State Stability (ISS)

Class $\mathcal{K}$ A continuous function $\gamma : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is said in $\mathcal{K}$ if it is strictly increasing and $\gamma(0) = 0$

Class $\mathcal{KL}$ A continuous function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is said in $\mathcal{KL}$ if $\beta(\cdot, s) \in \mathcal{K}$ for each fixed s and $\beta(r, \cdot)$ is decreasing for each fixed r and $\lim_{s\rightarrow\infty} \beta(r, s) = 0$

Definition: Input-to-State Stability

A system

$$\dot{x} = f(x, u)$$

is said to be input-to-state stable if there exists functions $\beta \in \mathcal{KL}$ and $\gamma \in \mathcal{K}$ such that for each bounded $u(\cdot)$ and each $x(0)$, the solution $x(t)$ exists and is bounded by:

$$\|x(t)\| \leq \beta(\|x(0)\|, t) + \gamma(\sup_{0\leq\tau\leq t} \|u(\tau)\|)$$

- Roughly speaking, ISS characterizes a relation between the norm of the state and the energy injected in the system: the system can not diverge in finite time with a bounded control
- ISS eases the design of control law
- Important literature

Outline

Sliding mode control

An introducing example

- Take again the linear system

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 + u \end{aligned}$$

- Assume u is forced to remain in $[-1, 1]$
  In practical applications, u is always constrained in an interval $[\underline{u}, \overline{u}]$. We will see later on how to handle it. To begin, we take the above simplified constraint.
- Take the CLF

$$V(x) = \frac{3}{2}x_1^2 + x_1 x_2 + x_2^2 = x^T \begin{pmatrix} \frac{3}{2} & \frac{1}{2} \\ \frac{1}{2} & 1 \end{pmatrix} x$$

Goal

Find u in $[-1, 1]$ that makes V decrease as fast as possible

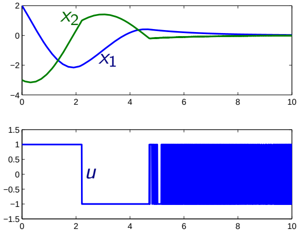
Sliding mode control

An introducing example

- Making V decrease as fast as possible is equivalent to minimizing $\dot{V}$:

$$\begin{aligned} u &= \underset{u\in[-1,1]}{\text{Argmin}}\left\{\dot{V}\right\} \\ &= \underset{u\in[-1,1]}{\text{Argmin}}\left\{\underbrace{-x_1^2+x_1x_2+x_2^2}_{\text{can not be changed}}+\underbrace{(x_1+2x_2)u}_{\text{minimal for } u=-\text{sign}(x_1+2x_2)}\right\} \\ &= -\text{sign}(x_1+2x_2) \end{aligned}$$

- Simulating the closed loop system with initial condition $x(0)=(2,-3)$, it gives:

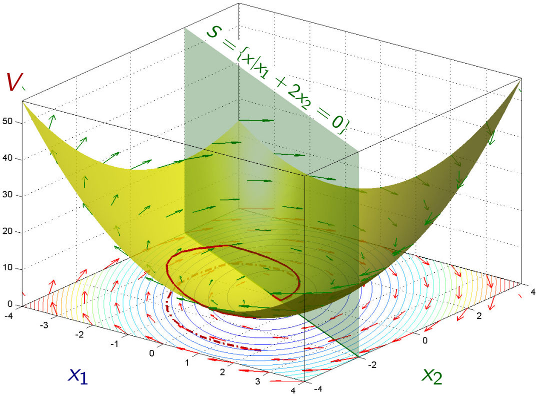


- **Why does it work ?**

Sliding mode control

An introducing example

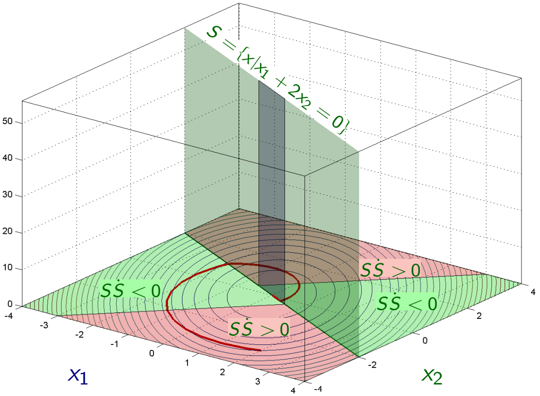
- Define $\sigma(x) = x_1 + 2x_2$. The set $\{S(x) = \{x|\sigma(x) = 0\}\} \cap \{|x_1| \leq 0.6, |x_2| \leq 0.3\}$ is attractive:



Sliding mode control

An introducing example

- Define $\sigma(x) = x_1 + 2x_2$. The set $\{S(x) = \{x|\sigma(x) = 0\}\} \cap \{|x_1| \leq 0.6, |x_2| \leq 0.3\}$ is attractive:



Sliding mode control

An introducing example

- The set $\{S(x) = \{x|\sigma(x) = 0\}\} \cap \{|x_1| \leq 0.6, |x_2| \leq 0.3\}$ is attractive
- Once the set is joined, the control is such that x remains on $S(x)$, that is:

$$\dot{S}(x) = 0 = \dot{x}_1 + 2\dot{x}_2 = x_2 - x_1 + u$$

- Hence, on $S(x)$, $u = -\mathrm{sign}(x_1 + 2x_2)$ has the same influence on the system as the control $u_{\text{eq}} = x_1 - x_2$
- On $S(x)$, the system behaves like:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 + u = -x_2 \end{aligned}$$

- x_2 and hence also x_1 clearly exponentially go to zero without leaving $\{S(x) = \{x|x_1 + 2x_2 = 0\}\} \cap \{|x_1| \leq 0.6, |x_2| \leq 0.3\}$

Sliding mode control

Principle of sliding mode control

Principle:

- Define a **sliding surface** $S(x) = \{x|\sigma(x) = 0\}$
- A stabilizing sliding mode control is a control law
  - **discontinuous in on** $S(x)$
  - that insures the **attractivity of** $S(x)$
  - such that, **on the surface** $S(x)$**, the states "slides" to the origin**

Main characteristic of sliding mode control:

✔ Robust control law

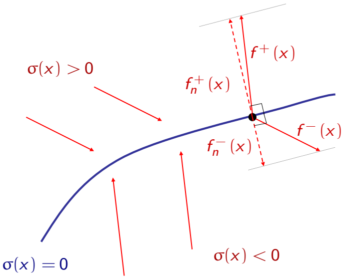
✘ Discontinuities may damage actuators (filtered versions exist)

Sliding mode control

Notion of sliding surface and equivalent control

- With a sliding mode control, the system takes the form:

$$\dot{x} = \begin{cases} f^+(x) & \text{if } \sigma(x) > 0 \\ f^-(x) & \text{if } \sigma(x) < 0 \end{cases}$$



- What happens for $\sigma(x) = 0$?
- The solution on $\sigma(x) = 0$ is the solution of $\dot{x} = \alpha f^+(x) + (1-\alpha) f^-(x)$ where α satisfies $\alpha f_n^+(x) + (1-\alpha) f_n^-(x) = 0$ for the normal projections of f^+ and f^-

Sliding mode control

A sliding mode control for a class of affine systems

Theorem: Sliding mode control for a class of nonlinear systems

- Take the nonlinear system:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{pmatrix} = \begin{pmatrix} f_1(x) + g_1(x)u \\ x_1 \\ \vdots \\ x_{n-1} \end{pmatrix} = f(x) + g(x)u$$

- Choose the control law

$$u = -\frac{p^T f(x)}{p^T g(x)} - \frac{\mu}{p^T g(x)} \operatorname{sign}(\sigma(x))$$

with $\mu > 0$, $\sigma(x) = p^T x$ and $p^T = \begin{pmatrix} p_1 & \cdots & p_n \end{pmatrix}$ is a stable polynomial (all roots have strictly negative real parts)

- Then **the origin** of the closed loop system is **asymptotically stable**

Sliding mode control

Sliding mode control for some affine systems: proof of convergence

Proof:

- Take the Lyapunov function

$$V(x) = \frac{1}{2}x^T pp^T x$$

- Note that $V(x) = \frac{\sigma^2(x)}{2}$
- Along the trajectories of the system, one has:

$$\dot{V}(x) = \sigma^T(x)\dot{\sigma}(x) = x^T p\left(p^T f(x) + p^T g(x)u\right)$$

with the chosen control, it gives:

$$\dot{V}(x) = -\mu\sigma(x)\,\mathrm{sign}(\sigma(x)) = -\mu\,|\sigma(x)|$$

- x **tends to** $S = \{x|\sigma(x) = 0\}$
- μ **tunes how fast the system converges to** $S = \{x|\sigma(x) = 0\}$

Sliding mode control

Sliding mode control for some affine systems: proof of convergence

- On S, the system is such that

$$\sigma(x) = 0 = p_1 x_1 + \cdots + p_{n-1} x_{n-1} + p_n x_n$$

- But:

$$\begin{aligned} x_{n-1} &= \dot{x}_n \\ x_{n-2} &= \dot{x}_{n-1} = x_n^{(2)} \\ &\vdots \\ x_1 &= \dot{x}_2 = \cdots = x_n^{(n-1)} \end{aligned}$$

- That can be written with $z_i = x_{n-i+1}$ in the form:

$$\dot{z} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ -\frac{p_2}{p_1} & -\frac{p_3}{p_1} & \cdots & \cdots & -\frac{p_n}{p_1} \end{pmatrix} z$$

- Hence, x **asymptotically goes to the origin** ■

Sliding mode control

Sliding mode control for some affine systems: time to switch

- Consider an initial point x_0 such that $\sigma(x_0) > 0$.
- Since

$$\sigma(x)\dot{\sigma}(x) = -\mu\sigma(x)\,\mathrm{sign}(\sigma(x))$$

  it follows that as long as $\sigma(x) > 0$

$$\dot{\sigma}(x) = -\mu$$

- Hence, the instant of the first switch is

$$t_s = \frac{\sigma(x_0)}{\mu} < \infty$$

- Moreover, $t_s \to 0$ as $\mu \to \infty$

Sliding mode control

Robustness issues

- Assume that one knows only a model

$$\dot{x} = \hat{f}(x) + \hat{g}(x)u$$

of the true system

$$\dot{x} = f(x) + g(x)u$$

- The time derivative of the Lyapunov function is:

$$\dot{V}(x) = \sigma(x)\left[\frac{p^T(f\hat{g} - \hat{f}g^T)p}{p^T\hat{g}} - \mu\frac{p^Tg}{p^T\hat{g}}\,\mathrm{sign}(\sigma(x))\right]$$

- If $\mathrm{sign}(p^Tg) = \mathrm{sign}(p^T\hat{g})$ and $\mu > 0$ sufficiently large, $\dot{V} < 0$
- **The closed-loop system is robust against model errors**

Outline

State and output linearization

Introducing examples

- Consider nonlinear the system

$$\begin{aligned} \dot{x}_1 &= -2x_1 + ax_2 + \sin(x_1) \\ \dot{x}_2 &= -x_2\cos(x_1) + u\cos(2x_1) \end{aligned}$$

- Take the new set of state variables

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= ax_1 + \sin(x_1) \end{aligned} \qquad \begin{aligned} x_1 &= z_1 \\ x_2 &= \tfrac{z_2 - \sin(z_1)}{a} \end{aligned}$$

- The state equations become:

$$\begin{aligned} \dot{z}_1 &= -2z_1 + z_2 \\ \dot{z}_2 &= -2z_1\cos(z_1) + \cos(z_1)\sin(z_1) + au\cos(2z_1) \end{aligned}$$

- The nonlinearities can then be canceled taking the new control v:

$$u = \frac{1}{a\cos(2z_1)}\left(v - \cos(z_1)\sin(z_1) + 2z_1\cos(z_1)\right)$$

State and output linearization

Introducing examples

- The system then becomes linear:

$$\dot{z} = \begin{pmatrix} -2 & 1 \\ 0 & 0 \end{pmatrix} z + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v$$

- Taking $v = -2z_2$ places the poles of the closed-loop in $\{-2, -2\}$
- Hence, $\lim_{t\to\infty} z = 0$
- Looking back to the transformation:

$$\begin{aligned} x_1 &= z_1 \\ x_2 &= \frac{z_2 - \sin(z_1)}{a} \end{aligned}$$

one also have $\lim_{t\to\infty} x = 0$
- In the original coordinates, the control writes:

$$u = \frac{1}{a\cos(2x_1)}[-2ax_2 - 2\sin(x_2) - \cos(x_1)\sin(x_1) + 2x_1\cos(x_1)]$$

State and output linearization

Introducing examples

- Consider nonlinear the system

$$\begin{aligned}\dot{x}_1 &= \sin(x_2) + (x_2+1)x_3\\ \dot{x}_2 &= x_1^5 + x_3\\ \dot{x}_3 &= x_1^2 + u\end{aligned}$$

- Take x_1 as "output": $y = x_1$

  When not imposed, the choice of the appropriate output is often delicate

- Compute the first time derivative of y:

$$\dot{y} = \dot{x}_1 = \sin(x_2) + (x_2+1)x_3$$

- Compute the second time derivative of y:

$$\ddot{y} = (x_2+1)u + \underbrace{(x_1^5 + x_3)(x_3 + \cos(x_2)) + (x_2+1)x_1^2}_{m(x)}$$

- Taking $u = \frac{v - m(x)}{x_2+1}$ gives the linear system: $\ddot{y} = v$

State and output linearization

Introducing examples

✘ Potential problem if $x_2 = -1$

- Here again, take the linear feedback $v = y + 2\dot{y}$ that places the poles of the closed loop system in $\{-1, -1\}$
- Hence, y and $\dot{y}$ asymptotically converges to the origin
- The state of the system is of dimension 3. Only two variables have been brought to the origin, what about the third one ?
- Write the system with the new coordinates $(z_1, z_2, z_3) = (y, \dot{y}, x_3)$:

Linearized sub-system:
$$\left\{\begin{array}{lcl}\dot{z}_1 & = & z_2\\ \dot{z}_2 & = & v\end{array}\right.$$

Internal dynamics:
$$\left\{\begin{array}{lcl}\dot{z}_3 & = & z_1^2 + u(z)\end{array}\right.$$

State and output linearization

Introducing examples

- One has to check that the internal dynamics is stable. For this, we assume that $y = \dot{y} = 0$ (which happens asymptotically):

$$\begin{aligned}\dot{z}_3 &= \overbrace{z_1^2}^{\to 0} + \frac{\overbrace{v}^{\to 0} - ((\overbrace{z_1^5}^{\to 0} + z_3)(z_3 + \cos x_2) + \overbrace{(x_2+1)z_1^2}^{\to 0})}{x_2 + 1} \\ &= -\frac{z_3(z_3 + \cos x_2)}{x_2 + 1}\end{aligned}$$

- If x_2 and z_3 are small enough, $\dot{z}_3 \approx -z_3(z_3 + 1) \approx -z_3$: the internal dynamics is locally asymptotically stable
- The approach can be applied if and only if the internal dynamics is stable
- If the internal dynamics is unstable, the system is called non-minimum phase
  - Change the input
  - Use additional inputs to stabilize the internal dynamics
  - Other approach in the literature like approximate linearization

State and output linearization

Introduction

- Consider again nonlinear systems affine in the control:

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases}$$

- The linearization approach tries to find
  - a feedback control law

$$u = \alpha(x) + \beta(x)v$$

  - and a change of variable

$$z = T(x)$$

that transforms the nonlinear systems into:

Input-State linearization	Input-Output linearization
$\dot{z} = Az + Bv$	$y^{(r)} = v$
✘ No systematic approach	✔ Systematic approach

State and output linearization

The SISO case

- Take

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = h(x) \end{cases}$$

- The time derivative of y is then given by:

$$\dot{y} = \frac{\partial h}{\partial x}(f(x) + g(x)u) = L_f h(x) + L_g h(x) u$$

- If $L_g h(x) \neq 0$, the control is

$$u = \frac{1}{L_g h(x)}(-L_f h(x) + v)$$

yielding $\dot{y} = v$

- Otherwise (that is $L_g h(x) \equiv 0$), differentiate once more:

$$\ddot{y} = \frac{\partial L_f h}{\partial x}(f(x) + g(x)u) = L_f^2 h(x) + L_g L_f h(x) u$$

State and output linearization

The SISO case

- If $L_g L_f h(x) \neq 0$, the control is

$$u = \frac{1}{L_g L_f h(x)}(-L_f^2 h(x) + v)$$

yielding to $\ddot{y} = v$

- ...

Definition: Relative degree

There exists an integer $r \leq n$ such that $L_g L_f^i h(x) \equiv 0$ for all $i \in \{1, \ldots, r-2\}$ and $L_g L_f^{r-1} h(x) \neq 0$ that is called the relative degree of the system

- The control law

$$u = \frac{1}{L_g L_f^{r-1} h(x)}(-L_f^r h(x) + v)$$

yields the linear system $y^{(r)} = v$

State and output linearization

The MIMO case

- Take

$$\begin{cases} \dot{x} = f(x) + g_1(x)u_1 + g_2(x)u_2 \\ y_1 = h_1(x) \\ y_2 = h_2(x) \end{cases}$$

- The relative degree $\{r_1, r_2\}$ is then given by:

$$\begin{array}{ll} L_{g_1} L_f^{i-1} h_1(x) = L_{g_2} L_f^{i-1} h_1(x) = 0 & \forall i < r_1 \\ L_{g_1} L_f^{i-1} h_2(x) = L_{g_2} L_f^{i-1} h_2(x) = 0 & \forall i < r_2 \end{array}$$

- Assume that the Input-Output decoupling condition is satisfied, that is:

$$\mathrm{rank} \begin{pmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & L_{g_2} L_f^{r_1-1} h_1(x) \\ L_{g_1} L_f^{r_2-1} h_2(x) & L_{g_2} L_f^{r_2-1} h_2(x) \end{pmatrix} = 2$$

State and output linearization

The MIMO case

- One has:

$$\begin{aligned}
\dot{y}_1 &= L_f h_1(x) + \underbrace{L_{g_1} h_1(x)}_{=0} u_1 + \underbrace{L_{g_2} h_1(x)}_{=0} u_2 \\
\ddot{y}_1 &= L_f^2 h_1(x) + \underbrace{L_{g_1} L_f h_1(x)}_{=0} u_1 + \underbrace{L_{g_2} L_f h_1(x)}_{=0} u_2 \\
&\vdots \\
y_1^{(r_1)} &= L_f^{r_1} h_1(x) + \underbrace{L_{g_1} L_f^{r_1-1} h_1(x)}_{\text{either } \neq 0} u_1 + \underbrace{L_{g_2} L_f^{r_1-1} h_1(x)}_{\text{or } \neq 0} u_2
\end{aligned}$$

- Make the same for y_2:

$$\begin{aligned}
\dot{y}_2 &= L_f h_2(x) + \underbrace{L_{g_1} h_2(x)}_{=0} u_1 + \underbrace{L_{g_2} h_2(x)}_{=0} u_2 \\
\ddot{y}_2 &= L_f^2 h_2(x) + \underbrace{L_{g_1} L_f h_2(x)}_{=0} u_1 + \underbrace{L_{g_2} L_f h_2(x)}_{=0} u_2 \\
&\vdots \\
y_2^{(r_2)} &= L_f^{r_2} h_2(x) + \underbrace{L_{g_1} L_f^{r_2-1} h_2(x)}_{\text{either } \neq 0} u_1 + \underbrace{L_{g_2} L_f^{r_2-1} h_2(x)}_{\text{or } \neq 0} u_2
\end{aligned}$$

State and output linearization

The MIMO case

- Hence, one has:

$$\begin{pmatrix} y_1^{(r_1)} \\ y_2^{(r_2)} \end{pmatrix} = \underbrace{\begin{pmatrix} L_f^{r_1} h_1(x) \\ L_f^{r_2} h_2(x) \end{pmatrix}}_{b(x)} + \underbrace{\begin{pmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & L_{g_2} L_f^{r_1-1} h_1(x) \\ L_{g_1} L_f^{r_2-1} h_2(x) & L_{g_2} L_f^{r_2-1} h_2(x) \end{pmatrix}}_{A(x)} \underbrace{\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}}_{u}$$

- Thanks to the decoupling condition, one can take:

$$u = -A(x)^{-1} b(x) + A(x)^{-1} v$$

which gives two chains of integrators:

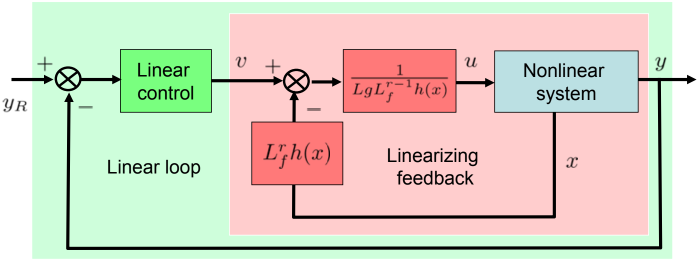
$$y_1^{(r_1)} = v_1 \qquad y_2^{(r_2)} = v_2$$

- This approach can be extended to any output and input sizes

State and output linearization

A block diagram representation

Linearization techniques can be interpreted as:



State and output linearization

The general case

- The internal dynamics or zero-dynamics is then of dimension $n - r$; its stability has to be checked
- For linear systems $r =$number of poles-number of zeros

- This approach took a rapid development in the 80's

✔ Maybe the only *"general"* approach for nonlinear system with predictive control

✔ Can be extended to MIMO systems with a decoupling condition

✔ Many extensions in particular with the notion of **flatness**

✔ Power tool for path generation

✘ Non robust approach since it is based on coordinate changes that can be stiff

✘ The control may take too large values

✘ "Kills" nonlinearities even if they are good

Outline

Backstepping

Main results

Theorem: backstepping design

Assume that the system $\dot{\chi} = f(\chi, v)$ with $f(0,0) = 0$ can asymptotically be stabilized with $v = k(\chi)$. Let V denote a $\mathcal{C}^1$ Lyapunov function (definite, positive and radially unbounded) such that $\frac{\partial V}{\partial \chi} f(\chi, k(\chi)) < 0$ for all $\chi \neq 0$. Then, the system

$$\begin{aligned} \dot{\chi} &= f(\chi, \xi) \\ \dot{\xi} &= h(\chi, \xi) + u \end{aligned}$$

with $h(0,0) = 0$ is also asymptotically stabilizable with the control:

$$u(\chi, \xi) = -h(\chi, \xi) + \frac{\partial k}{\partial \chi} f(\chi, \xi) - \xi + k(\chi) - \left[\frac{\partial V}{\partial \chi} G(\chi, \xi - k(\chi)) \right]^T$$

with

$$G(\chi, \xi) = \int_0^1 \frac{\partial f}{\partial \xi}(\chi, k(\chi) + \lambda \xi) d\lambda$$

The Lyapunov function

$$W(\chi, \xi) = V(\chi) + \frac{1}{2} \| \xi - k(\chi) \|^2$$

is then strictly decreasing for any $(\chi, \xi) \neq (0,0)$

Backstepping

Main results

The theorem can **recursively** be applied to

$$
\begin{aligned}
\dot{\chi} &= f(\chi, \xi_1) \\
\dot{\xi}_1 &= a_1(\chi, \xi_1) + b_1(\chi, \xi_1)\xi_2 \\
\dot{\xi}_2 &= a_2(\chi, \xi_1, \xi_2) + b_2(\chi, \xi_1, \xi_2)\xi_3 \\
&\vdots \\
\dot{\xi}_{n-1} &= a_{n-1}(\chi, \xi_1, \xi_2, \dots, \xi_{n-2}) + b_{n-1}(\chi, \xi_1, \xi_2, \dots, \xi_{n-2})\xi_{n-1} \\
\dot{\xi}_n &= a_n(\chi, \xi_1, \xi_2, \dots, \xi_{n-1}) + b_n(\chi, \xi_1, \xi_2, \dots, \xi_{n-1})u
\end{aligned}
$$

Forwarding

Strict-feedforward systems

- Consider the class of *strict-feedforward systems*:

$$\begin{aligned}
\dot{x}_1 &= x_2 + f_1(x_2, x_3, \ldots, x_n, u) \\
\dot{x}_2 &= x_3 + f_2(x_3, \ldots, x_n, u) \\
&\vdots \\
\dot{x}_{n-1} &= x_n + f_{n-1}(x_n, u) \\
\dot{x}_n &= u
\end{aligned}$$

- Strict-feedforward systems are in general no feedback linearizable
- Backstepping is not applicable

Forwarding

Forwarding procedure

- At step 0: Begin with stabilizing the system $\dot{x}_n = u_n$.
  Take e.g. $u_n = -x_n$ and the corresponding Lyapunov function $V_n = \frac{1}{2}x_n^2$
- At step 1: Augment the control law

$$u_{n-1}(x_{n-1}, x_n) = u_n(x_n) + v_{n-1}(x_{n-1}, x_n)$$

such that u_{n-1} stabilizes the cascade

$$\begin{aligned} \dot{x}_{n-1} &= x_n + f_{n-1}(x_n, u_{n-1}) \\ \dot{x}_n &= u_{n-1} \end{aligned}$$

- At step k: Augment the control law

$$u_{n-k}(x_{n-k}, \dots, x_n) = u_{n-k+1}(x_{n-k+1}, \dots, x_n) + v_{n-k}(x_{n-k}, \dots, x_n)$$

such that u_{n-k} stabilizes the cascade

$$\begin{aligned} \dot{\xi}_{n-k} &= x_{n-k+1} + f_{n-k}(x_{n-k+1}, \dots, x_n, u_{n-k}) \\ &\vdots \\ \dot{x}_{n-1} &= x_n + f_{n-1}(x_n, u_{n-k}) \\ \dot{x}_n &= u_{n-k} \end{aligned}$$

Forwarding

Forwarding procedure

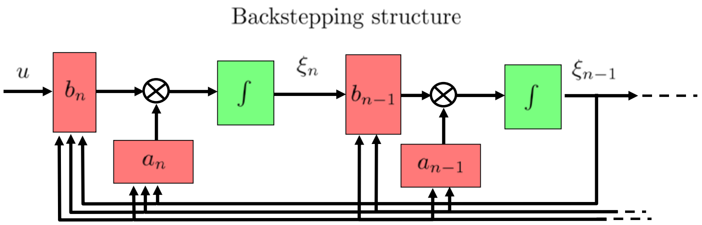
- The control law u_{n-k} always exists for all $k \in \{1, \ldots, n-1\}$
- The Lyapunov function at step k can be given by:

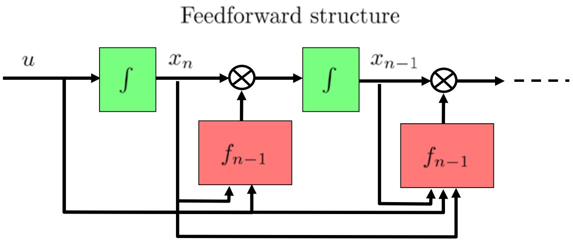
$$V_k = V_{k+1} + \frac{1}{2}x_k^2 + \int_0^\infty x_k(\tau) f_k(x_{k+1}(\tau), \ldots, x_n(\tau)) d\tau$$

- To avoid computations of the integrals, low-gain control can be used. It gave rise to an important litterature on bounded feedback control (Teel (91), Sussmann et al. (94))
- Various extensions for more general system exist but are still based on the same recursive construction
- Forwarding can be interpreted with passivity, ISS or optimal control

Backstepping/Forwarding

An interpretation of the names





Outline

The X4 helicopter

X4 position stabilization

- The aim is now to apply nonlinear control techniques to the stabilization problem of an X4 helicopter at a point
- Input-Ouput Linearization techniques can not be applied since the system is non minimum phase
- The backstepping approach is inspired from

  `S. Bouabdallah and R. Siegwart, ''Backstepping and sliding-mode techniques applied to an indoor micro quadrotor'', at the EPFL (Lausanne, Switzerland) presented at the International Conference on Robotics and Automation 2005 (ICRA'05, Barcelonna, Spain)`
- The PVTOL control strategy comes from

  `A. Hably, F. Kendoul, N. Marchand and P. Castillo, Positive Systems, chapter: "Further results on global stabilization of the PVTOL aircraft", pp 303-310, Springer Verlag , 2006`
- The saturated control law is taken in:

  `N. Marchand and A. Hably, "Nonlinear stabilization of multiple integrators with bounded controls", Automatica, vol. 41, no. 12, pp 2147-2152, 2005.`
- Revealing example of what often happens in nonlinear: a **solution comes from the combination of different methods**

The X4 helicopter

A two steps approach

$$\left\{\begin{array}{rcl} \dot{s}_i & = & -\frac{k_m^2}{J_r R} s_i - \frac{k_{gearbox}\kappa}{J_r} |s_i| s_i + \frac{k_m}{J_r R} \mathrm{sat}_{\bar{U}_i}(U_i) \\ \dot{\vec{p}} & = & \vec{v} \\ m\dot{\vec{v}} & = & -mg\vec{e}_3 + R \begin{pmatrix} 0 \\ 0 \\ \sum_i F_i(s_i) \end{pmatrix} \\ \dot{R} & = & R\vec{\omega}^\times \\ J\dot{\vec{\omega}} & = & -\vec{\omega}^\times J\vec{\omega} - \sum_i I_r \vec{\omega}^\times \begin{pmatrix} 0 \\ 0 \\ \sum_i s_i \end{pmatrix} + \begin{pmatrix} \Gamma_r(s_2, s_4) \\ \Gamma_p(s_1, s_3) \\ \Gamma_y(s_1, s_2, s_3, s_4) \end{pmatrix} \end{array}\right.$$

Position control problem

Attitude control problem

The X4 helicopter: attitude stabilization

A backstepping approach

- **The aim** of this first step **is to be able to bring** (ϕ, θ, ψ) **to any desired configuration** $(\phi_d, \theta_d, \psi_d)$
- Attitude equations (with an appropriate choice of Euler angles):

$$\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= a_1 x_4 x_6 + b_1 \Gamma_r \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= a_2 x_2 x_6 + b_2 \Gamma_p \\ \dot{x}_5 &= x_6 \\ \dot{x}_6 &= a_3 x_2 x_4 + b_3 \Gamma_y \end{cases}$$

with $(x_1, \dots, x_6) = (\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi})$, $a_1 = \frac{J_2 - J_3}{J_1}$, $a_2 = \frac{J_3 - J_1}{J_2}$, $a_3 = \frac{J_1 - J_2}{J_3}$, $b_1 = \frac{1}{J_1}$, $b_2 = \frac{1}{J_2}$ and $b_3 = \frac{1}{J_3}$

- The system would be trivial to control if (x_2, x_4, x_6) was the control instead of $(\Gamma_r, \Gamma_p, \Gamma_y)$
- This is precisely the philosophy of backstepping

The X4 helicopter: attitude stabilization

Recall of backstepping main result

Theorem: backstepping design

Assume that the system $\dot{\chi} = f(\chi, v)$ with $f(0,0) = 0$ can asymptotically be stabilized with $v = k(\chi)$. Let V denote a $\mathcal{C}^1$ Lyapunov function (definite, positive and radially unbounded) such that $\frac{\partial V}{\partial \chi} f(\chi, k(\chi)) < 0$ for all $\chi \neq 0$. Then, the system

$$\begin{aligned} \dot{\chi} &= f(\chi, \xi) \\ \dot{\xi} &= h(\chi, \xi) + u \end{aligned}$$

with $h(0,0) = 0$ is also asymptotically stabilizable with the control:

$$u(\chi, \xi) = -h(\chi, \xi) + \frac{\partial k}{\partial \chi} f(\chi, \xi) - \xi + k(\chi) - \left[\frac{\partial V}{\partial \chi} G(\chi, \xi - k(\chi))\right]^T$$

with

$$G(\chi, \xi) = \int_0^1 \frac{\partial f}{\partial \xi}(\chi, k(\chi) + \lambda \xi) d\lambda$$

The Lyapunov function

$$W(\chi, \xi) = V(\chi) + \frac{1}{2} \|\xi - k(\chi)\|^2$$

is then strictly decreasing for any $(\chi, \xi) \neq (0,0)$

The X4 helicopter: attitude stabilization

A backstepping approach

- Define first the tracking error:

$$\begin{cases} z_1 = x_1 - x_{1_d} = \phi - \phi_d \\ z_2 = x_3 - x_{3_d} = \theta - \theta_d \\ z_3 = x_5 - x_{5_d} = \psi - \psi_d \end{cases}$$

- It gives the subsystem

$$\begin{cases} \dot{z}_1 = x_2 - \dot{x}_{1_d} \\ \dot{z}_2 = x_4 - \dot{x}_{3_d} \\ \dot{z}_3 = x_6 - \dot{x}_{5_d} \end{cases}$$

where, at this step, (x_2, x_4, x_6) is the control vector

- Take first $V_1 = \frac{1}{2} z_1^2$.

The X4 helicopter: attitude stabilization

A backstepping approach

- Along the trajectories of the system, the Lyapunov function gives:

$$\dot{V}_1 = z_1(x_2 - \dot{x}_{1_d})$$

- Hence taking as fictive control $x_2 = \dot{x}_{1_d} - \alpha_1 z_1$ with $\alpha_1 > 0$ insures the decrease of V_1:

$$\dot{V}_1 = -\alpha_1 z_1^2$$

- x_1 will asymptotically converge to x_{1_d} ... unfortunately x_2 is not the control and we have to build thanks to backstepping approach "the Γ_r that will force x_2 to make x_1 converge to x_{1_d}"
- For this, define the tracking error for x_2:

$$z_2 = x_2 - \underbrace{(\dot{x}_{1_d} - \alpha_1 z_1)}_{\substack{\text{input of subsystem in } z_1 \\ \text{we would like to put}}}$$

The X4 helicopter: attitude stabilization

A backstepping approach

- One has

$$\dot{z}_2 = \underbrace{a_1 x_4 x_6 + b_1 \Gamma_r}_{\dot{x}_2} - \ddot{x}_{1_d} + \underbrace{\alpha_1 x_2 - \alpha_1 \dot{x}_{1_d}}_{\alpha_1 \dot{z}_1}$$

- Recall we have to build "the Γ_r that will force x_2 to make x_1 converge to x_{1_d}". For this, we use the formula of the backstepping theorem:

Theorem: backstepping design

$$\vdots$$

$$u(\chi, \xi) = -h(\chi, \xi) + \frac{\partial k}{\partial \chi} f(\chi, \xi) - \xi + k(\chi) - \left[\frac{\partial V}{\partial \chi} G(\chi, \xi - k(\chi))\right]^T$$

$$\vdots$$

with in our case:

$$\begin{aligned} \chi &= z_1 & \quad k &= \dot{x}_{1_d} - \alpha_1 z_1 \\ \xi &= z_2 & \quad f &= x_2 - \dot{x}_{1_d} \\ h &= a_1 x_4 x_6 - \ddot{x}_{1_d} + \alpha_1 x_2 - \alpha_1 \dot{x}_{1_d} & \quad V &= \tfrac{1}{2} z_1^2 \end{aligned}$$

The X4 helicopter: attitude stabilization

A backstepping approach

- Or more simply, we construct it:

$$W(z_1, z_2) = V(z_1) + \frac{1}{2}\left\|x_2 - (\dot{x}_{1_d} - \alpha_1 z_1)\right\|^2 = \frac{1}{2}(z_1^2 + z_2^2)$$

- Along the trajectories of the system:

$$\dot{W} = z_1(x_2 - \dot{x}_{1_d}) + z_2\left(a_1 x_4 x_6 + \overbrace{b_1 \Gamma_r}^{\text{control}} - \ddot{x}_{1_d} + \alpha_1 x_2 - \alpha_1 \dot{x}_{1_d}\right)$$

- Taking v as new control variable:

$$\begin{aligned} v &= a_1 x_4 x_6 + b_1 \Gamma_r - \ddot{x}_{1_d} + \alpha_1 x_2 - \alpha_1 \dot{x}_{1_d} \\ b_1 \Gamma_r &= v - a_1 x_4 x_6 + \ddot{x}_{1_d} - \alpha_1 x_2 + \alpha_1 \dot{x}_{1_d} \end{aligned}$$

gives:

$$\dot{W} = z_1(x_2 - \dot{x}_{1_d}) + z_2 v$$

The X4 helicopter: attitude stabilization

A backstepping approach

- Trying to write

$$\begin{aligned} x_2 - \dot{x}_{1_d} &= \underbrace{-\alpha_1 z_1}_{\text{ideal case}} +? \underbrace{(x_2 - \dot{x}_{1_d} + \alpha_1 z_1)}_{\text{difference between real and ideal cases}} \\ &= -\alpha_1 z_1 + \underbrace{x_2 - \dot{x}_{1_d} + \alpha_1 z_1}_{z_2} \end{aligned}$$

- Hence, it gives:

$$\dot{W} = -\alpha_1 z_1^2 + \underbrace{z_1 z_2}_{\text{can be compensated with } v} + z_2 v$$

- Taking:

$$v = -z_1 - \alpha_2 z_2$$

where $\alpha_2 > 0$ in order to insure

$$\dot{W} = -\alpha_1 z_1^2 - \alpha_2 z_2^2 < 0$$

- **Repeating this for θ and ψ gives the wanted control law**

The X4 helicopter: attitude stabilization

A backstepping approach

Attitude stabilization

The control law given by

$$\begin{aligned}
b_1\Gamma_r &= -z_1 - \alpha_2 z_2 - a_1 x_4 x_6 + \ddot{x}_{1_d} - \alpha_1 x_2 + \alpha_1 \dot{x}_{1_d} \\
b_2\Gamma_p &= -z_3 - \alpha_4 z_4 - a_2 x_2 x_6 + \ddot{x}_{3_d} - \alpha_3 x_4 + \alpha_3 \dot{x}_{3_d} \\
b_3\Gamma_y &= -z_5 - \alpha_6 z_6 - a_3 x_2 x_4 + \ddot{x}_{5_d} - \alpha_5 x_6 + \alpha_5 \dot{x}_{5_d}
\end{aligned}$$

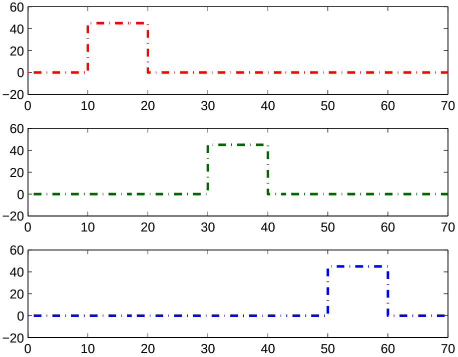
with $z_1 = x_1 - x_{1_d}$, $z_2 = x_2 - \dot{x}_{1_d} + \alpha_1 z_1$, $z_3 = x_3 - x_{3_d}$, $z_4 = x_4 - \dot{x}_{3_d} + \alpha_3 z_3$, $z_5 = x_5 - x_{5_d}$ and $z_6 = x_6 - \dot{x}_{5_d} + \alpha_5 z_5$ **asymptotically stabilizes** (x_1, x_3, x_5) **to their desired position** $(x_{1_d}, x_{3_d}, x_{5_d})$

- This kind of approach appeared in the literature in the last 90's
- Now better approach based on forwarding are appearing taking into account practical saturation of the control torques

The X4 helicopter: attitude stabilization

A backstepping approach

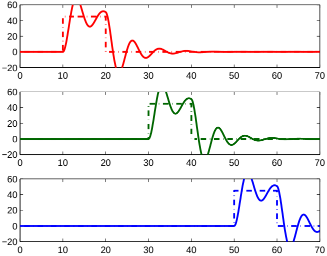
- Simulations: apply successive steps of 45 in roll, pitch and yaw



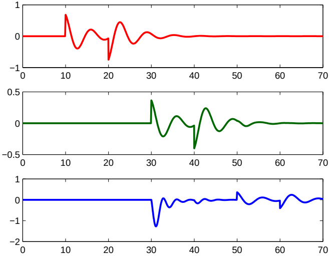
The X4 helicopter: attitude stabilization

A backstepping approach

- Adjusting the α's: with $\alpha_i = 0.2$



Roll, pitch and yaw answers



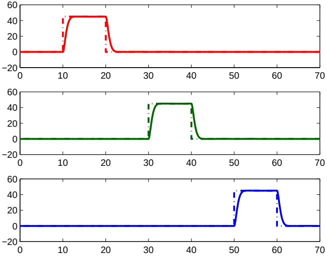
Γ_r, Γ_p and Γ_y controls

✘ **Too many oscillations**

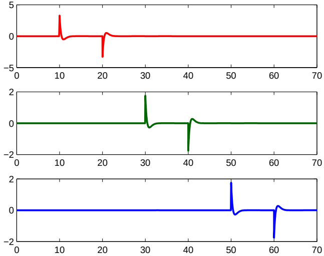
The X4 helicopter: attitude stabilization

A backstepping approach

- Adjusting the α's: with $\alpha_i = 2$



Roll, pitch and yaw answers



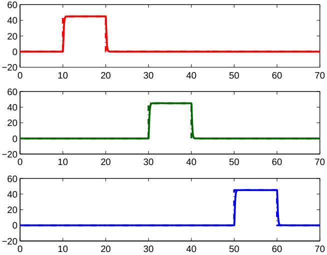
Γ_r, Γ_p and Γ_y controls

✔ **Seems to be a good choice**

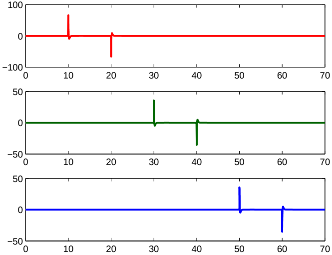
The X4 helicopter: attitude stabilization

A backstepping approach

- Adjusting the α's: with $\alpha_i = 10$



Roll, pitch and yaw answers



Γ_r, Γ_p and Γ_y controls

✘ The controls are **too large**, may be **two fast** to consider the rotors as instantaneous

The X4 helicopter: attitude stabilization

A backstepping approach

Successive steps of 45 in roll, pitch and yaw with $\alpha_i = 2$

The X4 helicopter: attitude stabilization

A backstepping approach

- The saturation of the possible control torques Γ_r, Γ_p, Γ_y may be problematic
- On the X4 system, the maximum speed of the rotors is given by the solution of:

$$k_m^2 s_{\max} + k_{gearbox} \kappa R s_{\max^2} - k_m \overline{U} = 0$$

which gives: $s_{\max} = 604 \text{ rad.s}^{-1}$

- The maximum roll and pitch torques are when one rotor is at rest, the other one at is maximum speed (we assume that the rotation is in only one direction):

$$\Gamma_{r,p}^{\max} = lbs_{\max}^2 = 0.31 \text{ N.m}$$

- The maximum yaw torque is obtained when two opposite rotors turn at the $s_{\max}$ the two others are at rest:

$$\Gamma_y^{\max} = 2\kappa s_{\max}^2 = 21.2 \text{ N.m}$$

The X4 helicopter: attitude stabilization

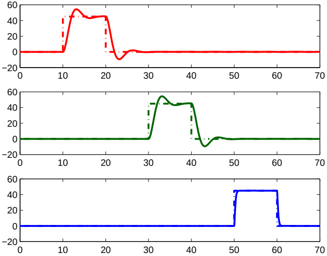
A backstepping approach

- Adapt the parameters to the saturation:
  - Take larger $\alpha_{5,6}$ than the $\alpha_{1,\ldots,4}$
  - Take small enough α_i to fulfill the saturation constraint

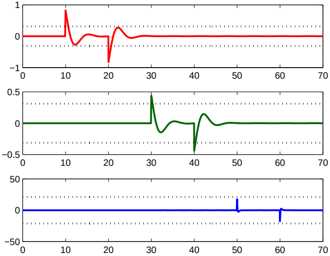
The X4 helicopter: attitude stabilization

A backstepping approach

- Adjusting the α's: with $\alpha_{1,\ldots,4} = 0.4$ and $\alpha_{5,6} = 8$



Roll, pitch and yaw answers



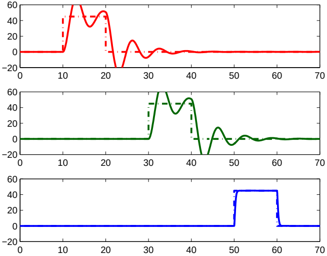
Γ_r, Γ_p and Γ_y controls

✘ The controls are still too large

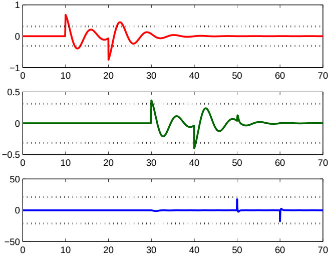
The X4 helicopter: attitude stabilization

A backstepping approach

- Adjusting the α's: with $\alpha_{1,\dots,4} = 0.2$ and $\alpha_{5,6} = 8$



Roll, pitch and yaw answers



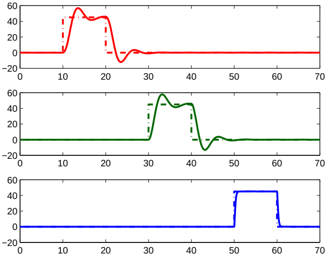
Γ_r, Γ_p and Γ_y controls

✘ Oscillations are now present

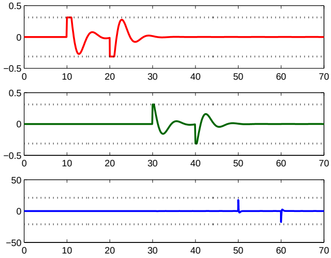
The X4 helicopter: attitude stabilization

A backstepping approach

- Taking $\alpha_{1,\dots,4} = 0.2$ and $\alpha_{5,6} = 8$ and applying it on the system **with saturations**, it gives:



Roll, pitch and yaw answers



Γ_r, Γ_p and Γ_y controls

✔ Seems to work

The X4 helicopter: attitude stabilization

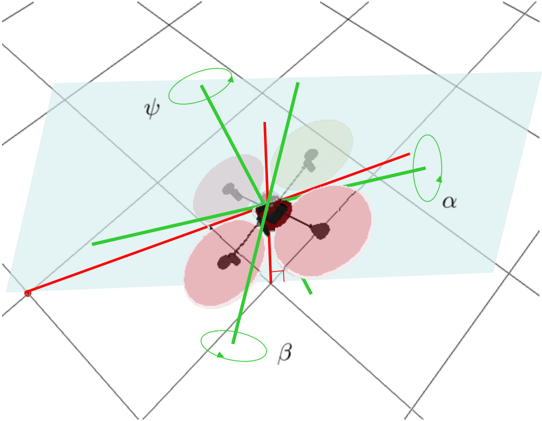
A backstepping approach

The control law with the saturated system

The X4 helicopter: position control

Simplification of the problem

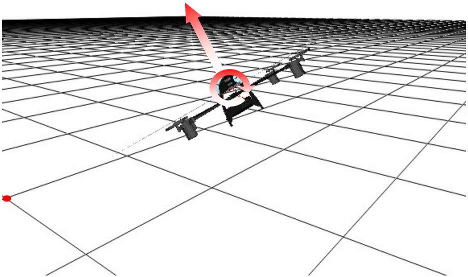
- First use the attitude control to adjust the X4 in the right direction:



The X4 helicopter: position control

Simplification of the problem

- There exists a relation between (α, β, ψ) and (ϕ, θ, ψ) (a rotation about the yaw axis)
- **Use the attitude control to drive and keep α and ψ to the origin**
- Take now the system in the plane:

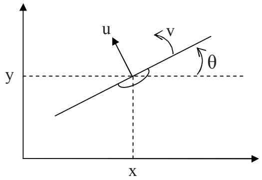


The X4 helicopter: position control

Simplification of the problem

- This problem is referred in the literature as the PVTOL aircraft stabilization problem (Planar Vertical Take Off and Landing aircraft)
- The "generic" equations are:

$$\begin{aligned} \ddot{x} &= -\sin(\theta)u \\ \ddot{y} &= \cos(\theta)u - 1 \\ \ddot{\theta} &= v \end{aligned}$$



where '-1' represent the normalized gravity, v represents the equivalent control torque.

The X4 helicopter: position control

A saturation based control

- Define $z \triangleq (z_1, z_2, z_3, z_4, z_5, z_6)^T \triangleq (x, \dot{x}, y, \dot{y}, \theta, \dot{\theta})^T$
- The system becomes:

Translational part:

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= -u\sin(z_5) \\ \dot{z}_3 &= z_4 \\ \dot{z}_4 &= u\cos(z_5) - 1 \end{aligned}$$

Rotational part:

$$\begin{aligned} \dot{z}_5 &= z_6 \\ \dot{z}_6 &= v \end{aligned}$$

- The idea is to use v to drive z_5 to z_5^d such that:

$$z_{5_d} \triangleq \arctan\left(\frac{-r_1}{r_2 + 1}\right)$$

with $\varepsilon < \frac{1}{2}$ (tuning parameter) and $\sigma(\cdot) = \max(\min(\cdot, 1), -1)$:

$$\begin{aligned} r_1 &= -\varepsilon\sigma(z_2) - \varepsilon^2\sigma(\varepsilon z_1 + z_2) \\ r_2 &= -\varepsilon\sigma(z_4) - \varepsilon^2\sigma(\varepsilon z_3 + z_4) \end{aligned}$$

The X4 helicopter: position control

A saturation based control

- When z_5 reaches z_{5_d} and applying the thrust control input u

$$u = \sqrt{r_1^2 + (r_2 + 1)^2}$$

the translational subsystem takes the form of two independent second order chain of integrators

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = r_1 \end{cases} \qquad \begin{cases} \dot{z}_3 = z_4 \\ \dot{z}_4 = r_2 \end{cases}$$

- If we could prove that

$$\begin{aligned} r_1 &= -\varepsilon\sigma(z_2) - \varepsilon^2\sigma(\varepsilon z_1 + z_2) \\ r_2 &= -\varepsilon\sigma(z_4) - \varepsilon^2\sigma(\varepsilon z_3 + z_4) \end{aligned}$$

insures $(z_1, z_2, z_3, z_4) \to 0$

- Then, it will follow that once $(z_1, z_2, z_3, z_4) = 0$, $z_{5_d} = 0$ and $\dot{z}_{5_d} = z_{6_d} = 0$ hence also $(z_5, z_6) \to 0$

The X4 helicopter: position control

A saturation based control

Remain two problems:

- Prove that

$$\begin{aligned} r_1 &= -\varepsilon\sigma(z_2) - \varepsilon^2\sigma(\varepsilon z_1 + z_2) \\ r_2 &= -\varepsilon\sigma(z_4) - \varepsilon^2\sigma(\varepsilon z_3 + z_4) \end{aligned}$$

  brings (z_1, z_2, z_3, z_4) to zero
- Build a control so that (z_5, z_6) tends to $(z_{5_d}, \dot{z}_{5_d} = z_{6_d})$

The X4 helicopter: position control

A saturated control law for linear systems

- An integrator chain is defined by:

$$\dot{x} = \underbrace{\begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ \vdots & \ddots & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix}}_{=:A} x + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{=:B} u \qquad (11)$$

- **Problem**: stabilizing (11) with

$$-\bar{u} \leq u \leq \bar{u}$$

and $\bar{u}$ is a positive constant

The X4 helicopter: position control

A saturated control law for linear systems

- First compute

$$\prod_{i=1}^{n}(\lambda+\varepsilon^i) = p_0 + p_1\lambda + \cdots + p_{n-1}\lambda^{n-1} + \lambda^n$$

- Apply the coordinate change

$$y := \frac{\sum_{i=1}^n \varepsilon^i}{\bar{u}} Tx \quad \text{with} \quad \begin{cases} T_n &= B \\ T_{n-1} &= (A+p_{n-1}I)B \\ T_{n-2} &= (A^2+p_{n-1}A+p_{n-2}I)B \\ & \vdots \\ T_1 &= (A^{n-1}+p_{n-1}A^{n-2}+\cdots+p_1 I)B \end{cases}$$

- The system becomes normalized in

$$\dot{y} = \begin{pmatrix} 0 & \varepsilon^{n-1} & \varepsilon^{n-2} & \dots & \varepsilon \\ 0 & 0 & \varepsilon^{n-2} & \dots & \varepsilon \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & \varepsilon \\ 0 & \dots & \dots & 0 & 0 \end{pmatrix} y + \begin{pmatrix} 1 \\ \vdots \\ \vdots \\ \vdots \\ 1 \end{pmatrix} v$$

with $-\sum_{i=1}^n \varepsilon^i \leq v \leq \sum_{i=1}^n \varepsilon^i$

The X4 helicopter: position control

A saturated control law for linear systems

Theorem: An efficient saturated control

- Let $\bar{\varepsilon}(n)$ denote the only root of $\varepsilon^n - 2\varepsilon + 1 = 0$ in $]0, 1[$.
- Then for all ε with $0 < \varepsilon < \bar{\varepsilon}(n)$,

$$u = -\frac{\bar{u}}{\sum_{i=1}^{n} \varepsilon^i} \sum_{i=1}^{n} \varepsilon^{n-i+1} \operatorname{sat}_1(y_i)$$

globally asymptotically stabilizes the integrator chain.

The X4 helicopter: position control

A saturated control law for linear systems

- **Sketch of the proof:**
  - Assume that $|y_n| > 1$ and take $V_n := \frac{1}{2}y_n^2$
  - Then

$$\dot{V}_n = -y_n \underbrace{\varepsilon \operatorname{sat}_1(y_n)}_{=\varepsilon} - y_n \Big[\underbrace{\varepsilon^2 \operatorname{sat}_1(y_{n-1}) + \cdots + \varepsilon^n \operatorname{sat}_1(y_1)}_{|\cdot| < \varepsilon^2 + \cdots + \varepsilon^n} \Big]$$

  - So V is decreasing if

$$\varepsilon > \varepsilon^2 + \cdots + \varepsilon^n \Leftrightarrow 1 - 2\varepsilon + \varepsilon^n > 0 \Leftrightarrow \varepsilon < \bar{\varepsilon}(n)$$

  - y_n joins $[-1, 1]$ in finite time and remains there
  - During that time, y_{n-1} to y_1 can not blow up
  - Repeating this scheme for y_{n-1} to y_1 gives $y \in \mathcal{B}(1)$ after some time
  - In $\mathcal{B}(1)$, the system is linear and stable $\Rightarrow$ GAS
- The construction of this feedback law is based on feedforward

The X4 helicopter: position control

A saturation based control

Remain two problems:

- Prove that

$$\begin{aligned} r_1 &= -\varepsilon\sigma(z_2) - \varepsilon^2\sigma(\varepsilon z_1 + z_2) \\ r_2 &= -\varepsilon\sigma(z_4) - \varepsilon^2\sigma(\varepsilon z_3 + z_4) \end{aligned}$$

  brings (z_1, z_2, z_3, z_4) to zero

  ➸ direct application of the saturated control law

- Build a control so that (z_5, z_6) tends to $(z_{5_d}, \dot{z}_{5_d} = z_{6_d})$

  ➸ take:

$$v = \sigma_\beta(\ddot{z}_{5_d}) - \varepsilon\sigma(z_6 - \dot{z}_{5_d}) - \varepsilon^2\sigma(\varepsilon(z_5 - z_{5_d}) + (z_6 - \dot{z}_{5_d}))$$

  that work applying the saturated control law on the variables $z_5 - z_{5_d}$ and $z_6 - \dot{z}_{5_d}$

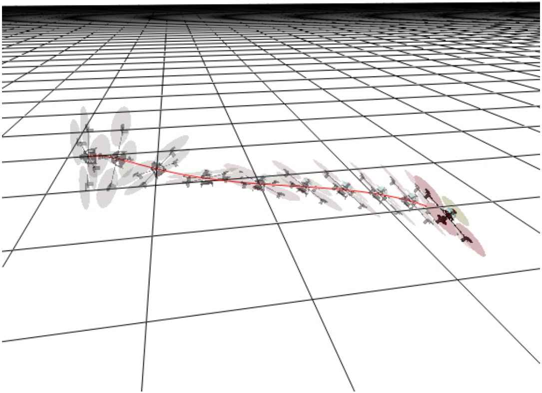
The X4 helicopter

Second step: position control

- $(\phi, \theta, \psi, x, y, z) = (\frac{pi}{2}, \frac{pi}{2}, \frac{pi}{4}, 5, 5, 5)$
- $(\dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}, \dot{z}) = (0, 0, 0, 1, 2, 0)$

The X4 helicopter

Second step: position control



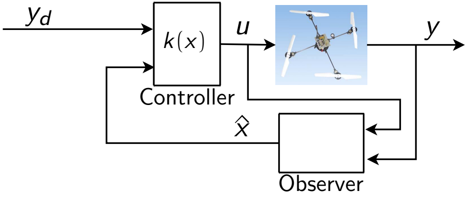
Zoom on the 10 first seconds with a snapshot every second

Initial conditions:

- $(\phi, \theta, \psi, x, y, z) = (\frac{pi}{2}, \frac{pi}{2}, \frac{pi}{4}, 5, 5, 5)$
- $(\dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{x}, \dot{y}, \dot{z}) = (0, 0, 0, 1, 2, 0)$

Outline

Nonlinear observers



1. **Modelization**
   - To get a mathematical representation of the system
   - Different kind of model are useful. Often:
     - a simple model to build the control law
     - a sharp model to check the control law and the observer
2. Design the **state reconstruction**: in order to reconstruct the variables needed for control
3. Design the **control** and test it

- **Close the loop**

- **Linear systems**: Simple observer

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

  - **Observability given by** $\mathrm{rank}(C, CA, \ldots, CA^{n-1})$
  - Once the observability has been checked, **define the observer**:

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x} + Bu(t) + L(\hat{y}(t) - y(t)) \\ \hat{y}(t) = C\hat{x}(t) \end{cases}$$

    with L such that $\Re(\mathrm{eig}(A + LC)) < 0$.
  - **Then $\hat{x}$ tends to x asymptotically with a speed related to eig$(A + LC)$**

- **Linear systems (cntd.)**:
  - convergence of the observer
    Define the error $e(t) := \hat{x}(t) - x(t)$. Then:

$$\dot{e}(t) = Ae(t) + L(\hat{y}(t) - y(t)) = (A + KC)e(t)$$

    Hence, if $\Re(\text{eig}(A + LC)) < 0$, $\lim_{t\to\infty} \hat{x}(t) = x(t)$
  - **Separation principle**: the controller and the observer can be designed separately, if each are stable, their association will be stable

- **Linear systems (cntd.)**: Kalman filter

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L(\hat{y}(t) - y(t)) \\ \hat{y}(t) = C\hat{x}(t) \end{cases}$$

- The way to chose L proposed by Kalman is:

$$AP + PA^T - PC^TW^{-1}CP + V + \delta P = 0$$
$$W = W^T > 0$$
$$L = -PC^TW^{-1}$$

with $\delta > 2\|A\|$ or $V = V^T > 0$.

- **δ enables to tunes the speed of convergence of the observer**
- The observer can also be computed as $L = -S^{-1}C^TW^{-1}$ where

$$A^TS + SA - C^TW^{-1}C + SVS + \delta S = 0$$

- **Time-varying linear systems**:

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) \end{cases}$$

- **Kalman filter**:

$$\begin{cases} \dot{\hat{x}} = A(t)\hat{x}(t) + B(t)u(t) + L(t)(\hat{y}(t) - y(t)) \\ \hat{y}(t) = C(t)\hat{x}(t) \end{cases}$$

with:

$$\dot{P} = AP + PA^T - PC^T W^{-1} CP + V + \delta P$$

$$P(0) = P_0 = P_0^T > 0$$

$$W = W^T > 0$$

$$L = -PC^T W^{-1}$$

$$\delta > 2\|A\| \text{ or } V = V^T > 0$$

- **δ enables to tunes the speed of convergence of the observer**

- The observer can also be computed as $L = -S^{-1}C^TW^{-1}$ where

$$\begin{aligned} -\dot{S} &= A^TS + SA - C^TW^{-1}C + SVS + \delta S \\ S(0) &= S_0 = S_0^T > 0 \end{aligned}$$

- The observer is optimal in the sense that it minimizes

$$\int_0^t [C(\tau)z(\tau) - y(\tau)]^T W^{-1} [C(\tau)z(\tau) - y(\tau)] + [B(\tau)u(\tau)]^T V^{-1} [B(\tau)u(\tau)] \, d\tau + (x_0 - \hat{x}_0)^T P_0^{-1} (x_0 - \hat{x}_0)$$

- The observer is also optimal for
  - x is affected by a white noise w_x of variance V
  - y is affected by a white noise w_y of variance W
  - w_x and w_y are uncorrelated

- **Nonlinear systems**:

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ y(t) = h(x(t)) \end{cases}$$

- **Observability**
  - Linked with the notion that for two trajectories of the system x_1 and x_2 defined on $[0, t]$, we must have

$$\int_0^t \|h(x_1(\tau)) - h(x_2(\tau))\| \, d\tau > 0 \text{ if } x_1 \neq x_2$$

  - **Linked with the input**:

$$\dot{x} = \begin{pmatrix} 0 & u \\ 0 & 0 \end{pmatrix} x \quad y = \begin{pmatrix} 1 & 0 \end{pmatrix} x$$

is observable for any $u(t) \neq 0$ but not for $u(t) = 0$

- **Extended Kalman filter (EKF)**:

$$\begin{cases} \dot{\hat{x}} = f(x(t), u(t)) - L(t)(\hat{y}(t) - y(t)) \\ \hat{y}(t) = h(\hat{x}(t)) \end{cases}$$

with

$$\dot{P} = AP + PA^T - PC^T W^{-1} CP + V + \delta P$$
$$P(0) = P_0 = P_0^T > 0$$
$$W = W^T > 0$$
$$L = PC^T W^{-1}$$
$$\delta > 2\|A\| \text{ or } V = V^T > 0$$
$$A = \frac{\partial f}{\partial x}(\hat{x}(t), u(t)) \quad C = \frac{\partial h}{\partial x}(\hat{x}(t))$$

- **No guarantee of convergence** (except for specific structure conditions)

Luenberger-Like observers

- For a system of the form:

$$\begin{cases} \dot{x}(t) = Ax(t) + \phi(Cx(t), u) \\ y(t) = Cx(t) \end{cases}$$

with (A, C) observable, if $A - KC$ is stable, an **observer** is:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + \phi(y(t), u(t)) - K(C\hat{x}(t) - y(t))$$

- For a system of the form:

$$\begin{cases} \dot{x}(t) = A_0 x(t) + \phi(x(t), u(t)) \\ y(t) = C_0 x(t) \end{cases}$$

with (A_0, C_0) are in canonical form, if $A_0 - K_0 C_0$ is stable, λ sufficiently large, ϕ global lipschitz and

$$\frac{\partial \phi_i}{\partial x_j}(x, u) = 0 \text{ for } j \geq i + 1$$

an **observer** is:

$$\dot{\hat{x}}(t) = A_0\hat{x}(t) + \phi(\hat{x}(t), u(t)) - \mathrm{diag}(\lambda, \lambda^2, \ldots, \lambda^n) K_0 (C_0\hat{x}(t) - y(t))$$

- **Kalman-Like observers**
  - For a system of the form:

$$\begin{cases} \dot{x}(t) = A(u(t))x(t) + B(u(t)) \\ y(t) = Cx(t) \end{cases}$$

an **observer** is:

$$\dot{\hat{x}}(t) = A(u(t))\hat{x}(t) + B(u(t)) - K(t)(C\hat{x}(t) - y(t))$$

with

$$\dot{P} = A(u(t))P + PA^T(u(t)) - PC^TW^{-1}CP + V + \delta P$$
$$P(0) = P_0 = P_0^T > 0, \; W = W^T > 0$$
$$\delta > 2\left\|A(u(t))\right\| \text{ or } V = V^T > 0$$
$$L = PC^TW^{-1}$$

Kalman-Like observers (cntd.)

- For a system of the form:

$$\begin{cases} \dot{x}(t) = A_0(u(t), y(t))x(t) + \phi(x(t), u(t)) \\ y(t) = C_0 x(t) \end{cases}$$

with:

$$\begin{aligned} A_0(u,y) &= \begin{pmatrix} 0 & a_{12}(u,y) & & 0 \\ & & \ddots & \\ & & & a_{n-1n}(u,y) \\ 0 & & & 0 \end{pmatrix} \text{ bounded} \\ C_0 &= \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix} \\ \frac{\partial \phi_i}{\partial x_j}(x,u) &= 0 \text{ for } j \geq i+1 \end{aligned}$$

an **observer** is:

$$\dot{\hat{x}} = A(u,y)\hat{x} + \phi(\hat{x}, u) - \mathrm{diag}(\lambda, \lambda^2, \ldots, \lambda^n) K_0(t)(C_0\hat{x} - y)$$

with $\dot{P} = \lambda \left(A(u(t))P + PA^T(u(t)) - PC^T W^{-1} CP + \delta P \right)$

$P(0) = P_0 = P_0^T > 0, \ W = W^T > 0$

$\delta > 2 \left\| A(u(t)) \right\|$ and λ large enough

$L = PC^T W^{-1}$

Not mentionned:

- **Optimal observers** (robust, very efficient and easy to tune but costly)
  Based on something like:

$$\hat{x} = \operatorname{Arg}\min_{x(\cdot)} \int_{t_0}^{t_1} \| h(x(\tau)) - y(\tau) \| \, d\tau$$

- **Sliding mode observers**

Observers

Application to the attitude estimation

- **Application: attitude estimation**
  - **9 sensors**:
    - 3 triax accelerometers
    - 3 triax gyrometers
    - 3 triax magnetometers
  - The accelerometers give:

$$\vec{b}_{acc} = C(q)(\underbrace{\vec{a}}_{\text{acceleration}} + \underbrace{\vec{g}}_{\text{gravity}}) + \underbrace{\eta_{acc}}_{\text{noise}}$$

where $C(q) = (q_0^2 - \vec{q}^T\vec{q})I_3 + 2(\vec{q}\vec{q}^T - q_0\vec{q}^\times)$ is called the Rodrigues matrix, that is the rotation from the fixed frame to the mobile one.

  - The magnetometers give:

$$\vec{b}_{mag} = C(q)\vec{h}_{mag} + \underbrace{\eta_{mag}}_{\text{noise}}$$

where $\vec{h}_{mag}$ are the coordinates of the magnetic field in the fixed frame.

Observers

Application to the attitude estimation

- The gyrometers give:

$$\vec{b}_{gyr} = \omega + \underbrace{\eta_{gyr_1}}_{\text{noise}} + \underbrace{\nu_{gyr_1}}_{\text{bias}}$$

where $\vec{h}_{mag}$ are the coordinates of the magnetic field in the fixed frame.

- The bias drift is the main error and it deteriorates the accuracy of the rate gyros on the low frequency band.
- We take:

$$\left\{ \begin{array}{l} \vec{b}_{gyr} = \omega + \eta_{gyr_1} + \nu_{gyr_1} \\ \dot{\nu}_{gyr_1} = -\frac{1}{\tau}\nu_{gyr_1} + \underbrace{\eta_{gyr_2}}_{\text{noise}} \end{array} \right.$$

Observers

Application to the attitude estimation

Nonlinear observer for attitude estimation

An observer of the attitude can be:

$$\begin{cases} \dot{\hat{q}} = \frac{1}{2}\Omega(\vec{b}_{gyr} - \hat{\nu}_{gyr_1} + K_1\varepsilon)\hat{q} \\ \dot{\hat{\nu}} = -T^1\hat{\nu} - K_2\varepsilon \end{cases}$$

where T is a diagonal matrix of time constant, K_i are positive definite matrices, ε is given by:

$$\varepsilon = \vec{q}_e \,\mathrm{sign}(q_{e_0})$$

where $q_e = (q_{e_0}, \vec{q}_e)$ is the quaternion error between the estimate $\hat{q}$ and a direct projection obtained with $\vec{b}_{mag}$ and $\vec{b}_{acc}$

Observers

Application to the attitude estimation

- Block diagram of the observer