Lecture – 32

Dynamic Inversion Design – II

Dr. Radhakant Padhi
Asst. Professor
Dept. of Aerospace Engineering
Indian Institute of Science - Bangalore

Topics

- Summary of Dynamic Inversion Design
- Advantages
- Issues and Possible Remedies

Summary of Dynamic Inversion Design

Dr. Radhakant Padhi
Asst. Professor
Dept. of Aerospace Engineering
Indian Institute of Science - Bangalore

Philosophy of Dynamic Inversion

- Carryout a co-ordinate transformation such that the problem “appears to be linear” in the transformed co-ordinates.
- Design the control for the linear-looking system using the linear control design techniques.
- Obtain the controller for the original system using an inverse transformation.
- Intuitively, the control design is carried out by enforcing a stable linear error dynamics.

Problem

- System Dynamics:

$$\dot{X} = f(X, U)$$
$$Y = h(X)$$
$$X \in \mathbb{R}^n, U \in \mathbb{R}^m, Y \in \mathbb{R}^p$$

- Goal (Tracking):

$$Y \to Y^*(t), \quad t \to \infty$$

Assumption: $Y^*(t)$ is smooth

- Special Class:

(control affine & square)

$$\dot{X} = f(X) + [g(X)]U$$
$$p = m, \quad [g_Y(X)] \text{ non-singular } \forall t$$

Dynamic Inversion Design

- Derive the output dynamics:

$$\dot{Y} = \left(\frac{\partial h}{\partial X}\right)\dot{X}$$

$$= \left(\frac{\partial h}{\partial X}\right)\left\{f(X) + [g(X)]U\right\}$$

$$= f_Y(X) + [g_Y(X)]U$$

Known: $\dot{X} = f(X) + [g(X)]U$

$Y = h(X)$

$$\frac{\partial h}{\partial X} \triangleq \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \cdots & \frac{\partial h_p}{\partial x_n} \end{bmatrix}$$

- Define Error of Tracking:

$$E(t) \triangleq \left[Y(t) - Y^*(t)\right]$$

Dynamic inversion

- Select a ***fixed*** gain $K > 0$ such that:

$$\dot{E} + K\,E = 0 \qquad \Rightarrow \qquad \boxed{E = e^{-Kt} E_0 \to 0, \quad t \to \infty}$$

Usually $K = diag\left(1/\tau_i\right),\ \tau_i > 0$

- Carry out the algebra:

$$\left(\dot{Y} - \dot{Y}^*\right) + K\left(Y - Y^*\right) = 0$$

$$f_Y\left(X\right) + \left[g_Y\left(X\right)\right]U = \dot{Y}^* - K\left(Y - Y^*\right)$$

- Solve for the controller:

$$\boxed{U = \left[g_Y\left(X\right)\right]^{-1}\left\{\dot{Y}^* - K\left(Y - Y^*\right) - f_Y\left(X\right)\right\}}$$

Is a first-order error dynamics always enforced?

Answer: **No!**

The order of the error dynamics is dictated by the "relative degree" of the problem, which is defined as the number of times the output needs to be differentiated so that the control variable appears explicitly.

If a second-order error dynamics needs to be enforced, then the corresponding equation is

$$\ddot{E} + K_V \dot{E} + K_P E = 0$$

Usually $K_V = diag\left(2\xi_i \omega_{n_i}\right),\ K_P = diag\left(\omega_{n_i}^2\right),\ \left(\xi_i, \omega_{n_i} > 0\right)$

When Does the Dynamic Inversion Fail?

Fundamental Principle of Dynamic Inversion:

1. Differentiate y repeatedly until the input u appears.
2. Design u to cancel the nonlinearity.

Q : Is it always possible to design u this way?

Ans: Not necessarily !

It is possible only if the relative degree is "well-defined".

Undefined Relative degree

Undefined relative degree : It may so happen that upon sucessive differentiation of y, u appears . However , the coefficient of u may vanish at X_0, whereas it is non-zero at points arbitrarly close to X_0. In such cases, the relative degree is undefined at X_0.

Ex : $\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \rho(x_1, x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$

$y = x_1^2, \quad \rho :$ Some nonlinear function

Then $\dot{y} = 2x_1\dot{x}_1 = 2x_1x_2$

$$\ddot{y} = 2x_1\dot{x}_2 + 2\dot{x}_1x_2 = 2x_1\left[\rho(x_1, x_2) + u\right] + 2x_2^2$$

Undefined Relative degree

$$\ddot{y} = \underbrace{2x_1\ \rho(x_1, x_2) + 2x_2^2}_{f_y(X)} + \underbrace{2x_1\ u}_{g_y(X)}$$

As $x_1 = 0$, $g_y(x) = 0$.

Hence, at $x_1 = 0$ the relative degree is **not defined**.

Note: If one choses $y = x_1$, then

$$\dot{y} = \dot{x}_1 = x_2$$

$$\ddot{y} = \dot{x}_2 = \rho(x_1, x_2) + u$$

and the coefficient of $u = 1 \neq 0$ globally.

In this case the relative degree is **well defined** globally.

Advantages of DI design

- Simple design: No need of tedious gain scheduling (hence, sometimes dynamic inversion is known as a universal gain scheduling design).
- Easy online implementation: It leads to a 'closed form solution' for the controller.
- Asymptotic (rather exponential) stability is guaranteed for the Error dynamics (subjected to the control availability, this is true 'globally').
- No problem if parameters are updated (the updated values can simply be used in the formula).

Issues and Remedies in Dynamic Inversion Design

Dr. Radhakant Padhi
Asst. Professor
Dept. of Aerospace Engineering
Indian Institute of Science - Bangalore

Issues of DI design

- Does the inverse exist for all time?
- What if the problem is non-square?

  (i.e. the number of controllers are not equal to the number of performance outputs)?
- Is the "*internal dynamics*" stable?
- Is it sensitive to modeling inaccuracies (e.g. parameter inaccuracies)?

Does the inverse exist for all time?

Answer: Not Necessarily!

<u>Strategies:</u>

- Do not update the controller unless $abs\left(\left|g_Y\left(X\right)\right|\right) > \varepsilon$
  (it may lead to performance degradation locally)

- Reformulate the problem

- Some advanced techniques do address this issue partly in the control design process itself

What if the output dynamics is non-square?

- Case - 1: $m < p$

  No. of controllers < No. of objectives

  <u>Theorem:</u> Perfect tracking is not possible in general.

- Case - 2: $m > p$

  No. of controllers > No. of objectives

  Additional objectives can be demanded in the design

  One such approach: *"Optimal Dynamic Inversion"*

Optimal Dynamic Inversion:
Meeting the tracking objective with minimum Control

- Select a **(fixed)** gain $K > 0$ such that:

$$\dot{E} + K\,E = 0 \qquad \Rightarrow \qquad \boxed{E = e^{-Kt} E_0 \to 0, \quad t \to \infty}$$

Usually $K = diag\left(1/\tau_i\right), \;\; \tau_i > 0$

- Carry out the algebra:

$$\left(\dot{Y} - \dot{Y}^*\right) + K\left(Y - Y^*\right) = 0$$

$$f_Y\left(X\right) + \left[g_Y\left(X\right)\right]U = \dot{Y}^* - K\left(Y - Y^*\right)$$

- Hence:

$$\underbrace{\left[g_Y\left(X\right)\right]}_{A}U = \underbrace{\left\{\dot{Y}^* - K\left(Y - Y^*\right) - f_Y\left(X\right)\right\}}_{b}$$

Optimal Dynamic Inversion:
Meeting the tracking objective with minimum Control

- Minimize: $J = \frac{1}{2}\left(U^T R U\right)$
- Subjected to: $AU = b$

$$\begin{bmatrix} A \triangleq \left[g_Y(X) \right] \\ b \triangleq \dot{Y}^* - K\left(Y - Y^*\right) - f_Y(X) \end{bmatrix}$$

- Solution:

Augmented Performance Index: $\bar{J} = \frac{1}{2}\left(U^T R U\right) + \lambda^T \left(AU - b\right)$

Necessary Conditions: $\frac{\partial \bar{J}}{\partial U} = 0, \quad \frac{\partial \bar{J}}{\partial \lambda} = 0$

Optimal Dynamic Inversion:
Meeting the tracking objective with minimum Control

- Solution:

$$RU + A^T \lambda = 0$$
$$AU - b = 0$$

$$U = -R^{-1} A^T \lambda$$
$$A\left(-R^{-1} A^T \lambda\right) = b$$

$$\lambda = -\left(A R^{-1} A^T\right)^{-1} b$$
$$U = R^{-1} A^T \left(A R^{-1} A^T\right)^{-1} b$$

Useful Features of ODI

- The enforced error dynamics is satisfied exactly in ODI: Asymptotic tracking is guaranteed
- ODI gives a platform for **optimal control allocation** as well (both time-wise as well as location-wise).
- Special case: if $R = I$, then

$$U = R^{-1} A^T \left(A R^{-1} A^T \right)^{-1} b = \overbrace{\left[A^T \left(A A^T \right)^{-1} \right]}^{\textbf{Pseudo inverse}} b = A^{+} b$$

$$\boxed{U = \left[g_Y(X) \right]^{+} \left\{ \dot{Y}^{*} - K\left(Y - Y^{*} \right) - f_Y(X) \right\}}$$

References on ODI

- ***Radhakant Padhi*** *and Mangal Kothari*, "An Optimal Dynamic Inversion Based Neuro-Adaptive Approach for Treatment of Chronic Myelogenous Leukemia", *Computer Methods and Programs in Biomedicine, Vol.87, 2007, pp.208-228.*
- ***Radhakant Padhi*** *and S. N. Balakrishnan*, "Optimal Dynamic Inversion Control Design for a Class of Nonlinear Distributed Parameter Systems with Continuous and Discrete Actuators", *IET Control Theory and Applications, Vol. 1, Issue 6, Nov. 2007, pp.1662-1671.*

Is the "*internal dynamics*" always stable?

Answer: **Not Necessarily!**

- The control solution is meaningless, unless this issue is addressed explicitly.
- Standard Results:
  - If the "relative degree" of the problem is equal to the number of states, then the internal dynamics is stable.
  - Asymptotic stability of "zero dynamics" is sufficient for local input-to-state stability of internal dynamics.
- If analytical justification is not possible, extensive simulation studies must be carried out.
- A nonlinear system is said to be "**minimum phase**" if its zero dynamics is stable.

Is there a procedure for tracking control design for non-minimum phase systems?

Answer: **Fortunately YES!**

An "Output Redefinition Technique" is available, where the internal dynamics is first made locally stable before achieving the tracking objective.

References:

E. M. Wallner and K. H. Well, *"Attitude Control of a Re-entry Vehicle with Internal Dynamic",* Journal of Guidance, Control, and Dynamics, Vol. 26, No. 6, Nov-Dec 2003. (Application oriented: Good for practicing engineers. It also contains information about "zero dynamics" and the relationship of its stability with the stability of internal dynamics)

S. Gopalswamy and J. K. Hedrick, *"Tracking Nonlinear Non-Minimum Phase Systems Using Sliding Control",* International Journal of Control, Vol. 57, No. 5, 1993, pp.1141-1158. (Theoretical foundations for output redefinition technique: Good for academicians and further research)

Philosophy of Output Redefinition Technique

Key Idea: Do not aim for perfect tracking!

Approximate tracking is OK.

Let the original output $y = h(X)$ have unstable zero dynamics.
Then redefine output $y_1 = h_1(X)$ such that the resulting zero-dynamics for y_1 tracking is stable.

Next, design the controller such that $y_1(t) \to y_d(t)$ exactly.
Then this implies good tracking of the original output provided certain conditions are met.

Output Redefinition Technique: An Illustrative Simple Example

Let $y = \left[\frac{(1 - s/b) B_0(s)}{A(s)} \right] u$

where $b > 0$ and zeros of $B_0(s)$ are in the left half plane.

This system has a right-half plane zero at $s = b$.

To avoid the problem of unstable zero dynamics, let us consider the control of a nominal output y_1, which is defined as

$$y_1 = \left[\frac{B_0(s)}{A(s)} \right] u$$

Output Redefinition Technique: An Illustrative Simple Example

Design $u(t)$ such that $y_1(t) \to y_d(t)$ asymptotically.

Let us then analyze the case when $y_1(t) = y_d(t)$.

$$y(t) = \left(1 - s/b\right) y_1(t) = \left(1 - s/b\right) y_d(t)$$

Error $e(t) \triangleq \left[y(t) - y_d(t)\right] = -\left(\dot{y}_d(t)/b\right)$

Hence $e(t)$ is bounded as long as $\dot{y}_d(t)$ is bounded.

Further, if b is large (as compared to $\dot{y}_d(t)$)

$e(t)$ remains small.

Output Redefinition Technique: An Illustrative Simple Example

An alternate choice: $y_2 = \left[\dfrac{B_0(s)}{A(s)(1+s/b)} \right] u$

Let us then analyze the case when $y_2(t) = y_d(t)$.

$$y(t) = (1 - s/b)(1 + s/b)\, y_2(t) = \left(1 - s^2/b^2\right) y_d(t)$$

Error $e(t) \triangleq \left[y(t) - y_d(t) \right] = -\left(\ddot{y}_d(t)/b^2 \right)$

Hence $e(t)$ is bounded as long as $\ddot{y}_d(t)$ is bounded.

Furthermore, if b^2 is large (as compared to $\ddot{y}_d(t)$)

$e(t)$ remains small.

Output Redefinition Technique: An Illustrative Simple Example

Out of the two choices,

y_2 choice leads to a better tracking performance, provided $|\ddot{y}_d| < b|\dot{y}_d|$ and vice-versa.

Another Approach to Deal with Non-minimum phase Systems

Philosophy: Neglect the terms containing the input u, while doing successive iterations of the output until the number of successive differentiations becomes n. So there is "approximately" no internal dynamics. Design controller based on this "approximate" input-outpt dynamics.

Note: this approach works as long as the coefficients of u at the intermediate steps are small i.e. the system is "weakly non-minimum phase".

Other Ideas to Deal with Non-minimum phase Systems

(1) Modify the desired trajectories (this may not be feasible and/or it may limit the operating zone.

(2) Modify the plant itself.
This may be possible by relocation/addition of actuators/sensors. May also be possible by physical modification of the plant (e.g. placing the control surfaces at a different location in an aircraft).

Is DI design sensitive to modeling and parameter inaccuracies?

Answer: **Very much YES!**

- Solution: Augmenting the Dynamic Inversion design with other "Robust" and/or "Adaptive" control design techniques.
- One Popular Approach:

"***Neuro-Adaptive Control Design***"

Motivations for Neuro-Adaptive Design

- Perfect system modeling is difficult
- Sources of imperfection can arise from:
  - Unmodelled dynamics (missing algebraic terms in the model)
  - Inaccurate knowledge of system parameters
  - Change of system parameters/dynamics during operation
- The adaptive control design should be able to learn the unknown function through neural network(s) and then compensate for this unknown error.

A Demonstrative Example

$$\dot{x} = 2\sin(x) + 0.1\sin(x)$$

Known part of actual system (nominal system) — $2\sin(x)$

Unknown part of actual system — $0.1\sin(x)$

$$= 2\sin(x) + \Delta c \sin(x)$$

Weight — Δc

Basis Function — $\sin(x)$

Summary: Dynamic Inversion Design

- **DI design offers several advantages:**
  - Its is nonlinear design: No linearization of system dynamics is necessary
  - It is a promising substitute to gain scheduling philosophy. Quite often a constant gain is found to be satisfactory, even though gain scheduling can still be an option for performance improvement
  - It assures “perfect tracking” (i.e. asymptotic stability of error dynamics) under ideal assumption of perfect knowledge of the system dynamics
  - DI offers a “closed form solution” for the control and hence it can easily be implemented online

Summary: Dynamic Inversion Design

- **In DI design, one has to be careful about the following important issues:**
  - Non-existence of the matrix inversion
    - If this is a local problem, do not update the control
  - Stability of internal dynamics
    - If the analysis fails, then either reformulate the problem or opt for output redefinition technique
  - Robustness with respect to modeling inaccuracies
    - Augment the DI design with either robust or adaptive control design ideas). "**Neuro-adaptive technique"** has recently evolved as one such promising idea.

Thanks for the Attention...!