

# Particle Swarm Optimization in the Exact Linearization Technic for Output Tracking of Non- Minimum Phase Nonlinear Systems

*by*

---

**Submission date:** 09-Mar-2022 01:24PM (UTC+0700)

**Submission ID:** 1780072678

**File name:** naiborhuAMS109-112-2013.pdf (416.67K)

**Word count:** 4149

**Character count:** 18676

# Particle Swarm Optimization in the Exact Linearization Technic for Output Tracking of Non-Minimum Phase Nonlinear Systems

Janson Naiborhu, Firman and Khozin Mu'tamar

Department of Mathematics, Faculty of Mathematics and Natural Science  
Institut Teknologi Bandung, Jl Ganesa 10 Bandung 40132, Indonesia

Copyright © 2013 Janson Naiborhu, Firman and Khozin Mu'tamar. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 7 Abstract

In this paper, we develop a method to design the input control to track the output of a nonminimum-phase nonlinear systems asymptotically. The design of the control input is based on an exact linearization. To perform the exact linearization, the other output should be selected such that its relative degree is equal to the dimension of the system. Furthermore, the desired output of the output which has been selected will be set based on the desired output of the original system. The desired output of the output which has been selected is determined by Particle Swarm Optimization. In applying the input control which is obtained via output-input linearization sometimes led to singularity. To overcome this singularity problem, polynomial control is developed around the point of singularity.

**Keywords:** PSO, exact linearization, internal stability, non-minimum phase, polynomial control

## 1 Introduction

A system is called non-minimum phase if a nonlinear state feedback can hold the system output identically zero while the internal dynamics becomes unstable [6]. Recently, output tracking problems on nonlinear non-minimum phase

systems have been investigated intensively. The stable inversion proposed in [2, 3] is an iterative solution to the tracking problem with unstable zero dynamics. This method requires the system to have well defined relative degree and hyperbolic zero dynamics, i.e. no eigenvalues on imaginary axis. In the absence of imaginary eigenvalues, the zero dynamics manifold can be split into a stable and unstable manifold. This method tries to find a stable solution for the full state space trajectory by steering from the unstable zero dynamics manifold to the stable zero dynamics manifold. In [10], a new approach based on the notion of convergent systems is developed to solve the problem of stable inversion of nonlinear nonminimum phase systems. In [4], a minimum phase approximation to a single-input-single-output (SISO) nonlinear non-minimum phase system is derived. An input-output linearizing controller is designed for this approximation and then applied to the non-minimum phase plant. This leads to a system that internally stable. In [5] a controller is designed based upon an internal equilibrium manifold where this controller pushes the state of a nonlinear non-minimum phase system toward that manifold. This has afforded approximate output tracking for nonlinear non-minimum phase systems while maintaining internal stability. In [11], the asymptotic output tracking which is a class of causal nonminimum phase uncertain nonlinear systems is achieved by using higher order sliding modes (HOSM) without reduction of the input-output dynamics order. In [1], a new nonlinear dynamic controller is described based on the gradient descent control. Performance index is generated by error of output system from output desired value and internal state of the system. Adding of an internal state to generate performance index is mentioned to maintain the stability of internal dynamic of the system.

In this paper we used the "old method", i.e. input-output linearization method [6] to design input control which assures that the nonlinear system is stable asymptotically. Before this method is applied, the other output should be selected such that its relative degree is equal to the dimension of the system. Then, we transform the coordinate to get the normal form, exact linearization. Furthermore, the desired output of the output which has been selected will be set based on the desired output of the original system. In case that the desired output of the output which has been selected can not be determined explicitly, we use PSO to approximate it. In applying the input control which is obtained via output-input linearization sometimes led to singularity. To overcome this singularity problem, polynomial controls are developed around the point of singularity.

## 2 Problem Statement and Assumptions

Consider the following SISO affine nonlinear control system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u, \tag{1}$$

$$y = h(\mathbf{x}) \tag{2}$$

where  $\mathbf{x} \in \mathcal{R}^n$  is the state vector,  $u \in \mathcal{R}$  is the control input and  $y \in \mathcal{R}$  is the measured output.  $\mathbf{f} : \mathcal{R}^n \rightarrow \mathcal{R}^n$  is a smooth function with  $\mathbf{f}(\mathbf{0}) = \mathbf{0}$ ,  $\mathbf{g} : \mathcal{R}^n \rightarrow \mathcal{R}^n$  and  $h : \mathcal{R}^n \rightarrow \mathcal{R}$  are smooth functions. Assume also that  $h(\mathbf{0}) = 0$ . If the nonlinear system (1)-(2) has relative degree  $r$ , ( $r < n$ ) at  $\mathbf{x}^0$ , the system (1)-(2) can be transformed to

$$S = \begin{cases} \Sigma_{ext} : \begin{cases} \dot{\xi}_k = \xi_{k+1}, & k = 1, \dots, r-1 \\ \dot{\xi}_r = a(\xi, \eta) + b(\xi, \eta)u \end{cases} \\ \Sigma_{int} : \dot{\eta} = q(\xi, \eta) \\ y = \xi_1, \end{cases} \tag{3}$$

with the internal dynamics

$$\Sigma_{int} : \dot{\eta} = q(\xi, \eta). \tag{4}$$

The stability of the internal state  $\eta$  is required to guarantee the output system  $y(t)$  tracks the desired output  $y_d(t)$ . Our objective is to make the output  $y(t)$  tracks the desired output  $y_d(t)$  while keeping the state bounded. To keep the state bounded is difficult for non-minimum phase system. In this paper we design a controller such that external state  $y(t)$  tracks the desired output  $y_d(t)$  while keeping the state bounded via exact linearization.

Thus, in this paper we assume that

**Assumption 1** System (1) is exact linearizable.

The nonlinear system (1) is exact linearizable if it satisfies the theorem below.

**Theorem 1** ([6]) Suppose a nonlinear system (1) is given. The State Space Exact Linearization Problem is solvable near a point  $\mathbf{x}^0$  (i.e. there exists an output function  $\lambda(x)$  for which the system has relative degree  $n$  at  $\mathbf{x}^0$ ) if and only if the following conditions are satisfied

1. the matrix  $([g(x^0)ad_f g(x^0) \cdots ad_f^{n-2} g(x^0)ad_f^{n-1} g(x^0)])$  has rank  $n$ ,
2. the distribution  $D = \text{span}\{g, ad_f g, \dots, ad_f^{n-2} g\}$  is involutive near  $x^0$ .

According to the above Theorem, if the nonlinear systems (1) can be linearized exactly, there is an output function  $\lambda(x)$  such that the nonlinear system

$$\dot{x} = f(x) + g(x)u, \quad x \in \mathcal{R}^n, u \in \mathcal{R} \quad (5)$$

$$\dot{y} = \lambda(x), \quad y \in \mathcal{R} \quad (6)$$

can be transformed to

$$\begin{aligned} \dot{z}_k &= z_{k+1}, \quad k = 1, \dots, n-1 \\ \dot{z}_n &= a(\mathbf{z}) + b(\mathbf{z})u \\ y &= z_1 = \lambda(x). \end{aligned} \quad (7)$$

If  $b(\mathbf{z}(t)) \neq 0, \forall t$ , the tracking output problem can be solved by input-output linearization technique.

The input control which is obtained can be written as a static control law [6]

$$u_r = \frac{1}{b(\mathbf{z})} (-a(\mathbf{z}) + v), \quad (8)$$

where  $v = c_0 z_1 + c_1 \dot{z}_2 + \dots + c_n z_1^{(n)}$  and the value of  $c_i; i = 0, \dots, n$  is chosen such that the real part of the eigen values of polynomial  $p(s)$

$$p(s) = c_n s^n + c_{n-1} s^{n-1} + \dots + c_1 s + c_0$$

are negative.

Furthermore, for handling the case  $b(\mathbf{z}(t_s)) = 0$  for a  $t = t_s$ , in this paper, we design  $u$  in the neighborhood of the  $\mathbf{z}(t_s)$  by polynomial control.

### 3 Polynomial Bridge Singularity

Consider the equation (8). If  $b(\mathbf{z}(t)) \neq 0, \forall t$  it is said that the relative degree of the system is well defined. Otherwise if there is  $t = t_s$  such that  $b(\mathbf{z}(t_s)) = 0$ , it is said that the relative degree of the system is not well defined. The control law (8) is no more valid. In this case  $\mathbf{z}(t_s)$  is called a singular point for asymptotic output tracking [7, 8].

Define the set of singularity as

$$M_s = \{\mathbf{z} \in \mathcal{R}^n \mid b(\mathbf{z}) = 0\}.$$

For simplicity, we consider that  $M_s$  has one point, i.e  $M_s = \{\mathbf{z}_s\}$ . Afterwards, we develop the control law  $u_s(t)$  as a formal power series in the interval  $[t_s - \varepsilon, t_s + \varepsilon] = T_s$ , where  $\varepsilon > 0$  (the neighborhood of singular point). This control law is called formal control.

**Definition 1 ([9])** Given a singular point  $\mathbf{z}_s$  and a sufficiently smooth trajectory  $y_d$ , let  $r(\mathbf{z}_s, y_d) \geq 0$  be the largest integer such that there exists

$$\bar{v}^r = (v_0, v_1, \dots, v_{r-1}) \in \mathcal{R}^r$$

satisfying:

$$y_d^{(k)}(t_s) = y^{(k)}(t_s) = a_k(\mathbf{z}_s, v_0, v_1, \dots, v_{k-\alpha-1})$$

with  $\alpha \leq k \leq \alpha + r$  and  $v_i = u^{(i)}(t_s)$ .  
 $r(\mathbf{z}_s, y_d)$  is called rank of singularity.

**Proposition 1 ([9])** If for a trajectory  $y_d \in C^\infty(\mathcal{R})$  and a singular point  $\mathbf{z}_s$ , the rank of singularity  $r(\mathbf{z}_s, y_d)$  is infinite, then there exists a formal control insuring the tracking of  $y_d$  at the singular point  $\mathbf{z}_s$ .

**Proof.** Suppose that the rank of singularity is infinite. It follows from its definition that :

$$\exists (v_0, v_1, \dots, v_k, \dots) \in \mathcal{R}^\infty$$

solution of the following system of algebraic equations :

$$y_d^{(k)}(t_s) = a_k(\mathbf{z}_s, v_0, v_1, \dots, v_{k-\alpha-1}), \forall k \geq 0$$

where  $a_k(\cdot, \cdot)$  represents the Lie derivatives of order  $k$ , of the output function  $y$  evaluated at the singular point  $\mathbf{z}_s$ . Using  $v_i$ 's we can construct the control law (formal control):

$$u_s(t) = \sum_{i=0}^{\infty} \beta_i (t - t_s)^i, \quad \beta_i = \frac{v_i}{i!}, \quad t \in T_s \tag{9}$$

With this control law, we obtain :

$$\begin{aligned} y_d^{(k)}(t_s) &= a_k(\mathbf{z}_s, u_s(t_s), \dot{u}_s(t_s), \dots, u_s^{(k-\alpha-1)}(t_s)) \\ &= y^{(k)}(t_s), \quad \forall k \geq 0. \end{aligned}$$

**Proposition 2 ([9])** If the control law is analytic, the output of the system (1)-(2) follows the trajectory  $y_d \in C^\infty(\mathcal{R})$  in a neighborhood of the singular point.

Consider (1)-(2) and the derivatives of its output function :

$$\begin{aligned} y^{(j)} &= L_f^j h(\mathbf{z}) = a_j(\mathbf{z}), \\ y^{(r)} &= L_f^r h(\mathbf{z}) + L_g L_f^{r-1} h(\mathbf{z}) u = a_r(\mathbf{z}) + b(\mathbf{z}) u \\ y^{(r+k)} &= a_{r+k}(\mathbf{z}, u, \dot{u}, \dots, u^{(k-1)}) + b(\mathbf{z}) u^{(k)}, \quad k \geq 1. \end{aligned}$$

The value of  $v_k = u^{(k)}(t_s)$ , in equation (9) for  $k = 0, 1, 2, \dots$ , is a solution of (linear/nonlinear) equation systems

$$y_d^{(r+k)}(t_s) = a_{r+k}(\mathbf{z}(t_s), u(t_s), \dot{u}(t_s), \dots, u^{(k-1)}(t_s)), \quad k \geq 1, \quad (10)$$

where  $y_d^{(r+k)}(t_s), k \geq 1$  and  $\mathbf{z}(t_s)$  are known. ( $y_d(t)$  is the desired output).

Polynomial control is obtained by truncating the formal power series (formal control), (9). Let  $\mathbf{z}(t_s)$  be a singular point, then substitute  $\mathbf{z}(t_s)$  into the equation (16) to obtain:

$$\begin{aligned} y_d^{(r+1)}(t_s) &= a_{r+1}(\mathbf{z}(t_s), u(t_s)) \\ &\vdots = \vdots \\ y_d^{(r+m)}(t_s) &= a_{r+m}(\mathbf{z}(t_s), u(t_s), \dot{u}(t_s), \dots, u^{(m-1)}(t_s)) \end{aligned} \quad (11)$$

Solve equation (11), to find the value of  $u^{(i)}(t_s), \quad i = 0, 1, 2, \dots, m-1$ . Then we have

$$u_s(t) = \sum_{i=0}^{m-1} \frac{u^{(i)}(t_s)}{i!} (t - t_s)^i, \quad t \in T_s. \quad (12)$$

This polynomial control is used as a bridge which crossing the neighborhood of singular point.

Thus, for nonlinear system (SISO), (1)-(2), to achieve the tracking of the desired output  $y_d(t)$ , we propose the following control law

$$u(t) = \begin{cases} u_r(t) & ; \quad t \in [0, t_s - \varepsilon] \cup [t_s + \varepsilon, \infty) \\ u_s(t) & ; \quad t \in T_s \end{cases} \quad (13)$$

Before apply the control law (13), we have to set up the output desired for  $\lambda(x)$ , i.e.  $\lambda_d(t)$ . In this paper, first we consider the systems which satisfies the following assumptions.

**Assumption 2**  $h(x) = x_l$  for  $l \in \{1, 2, \dots, n\}$ .

**Assumption 3** If  $\lambda(x) = x_k$  then  $\dot{x}_k = f_k(x_l, x_k)$  can be solved by substituting  $x_l = y_d(t)$ .

Thus,  $\lambda_d(t) = x_k(t)$ . The second, we consider the systems which do not satisfy the assumption 3. In other word the output desired  $\lambda_d(t)$  can not be determined explicitly.

## 4 Particle Swarm Optimization

Assume that system (5)-(6) do not satisfy the assumption 3. The  $\lambda_d(t)$  can not be determined explicitly. In this section we approximate the  $\lambda_d(t)$  using PSO. Let the  $\lambda_d(t)$  in the form of Fourier series :

$$\lambda_d(t) = \alpha + \sum_{i=1}^N \beta_i \sin(t) + \gamma_i \cos(t). \tag{14}$$

By PSO we determined the value of  $\alpha$ ,  $\beta_i$  and  $\gamma_i$  for  $i = 1 \dots N$  such that  $\int_0^T (y(t) - y_d(t))dt$  goes to zero.

Let  $nD$  is the number of parameter which is determined by PSO.  $nS$  is the number of swarm which be used.  $nIT$  is the number of iteration maximum which admissible. Then,  $X$  is array of swarm position that it become a candidate solution of parameter which be find, furthermore  $X$  can be state as

$$X = \text{Array}(nD, nS, nIT)$$

$V$  is array velocity swarm where  $V$  can be state as

$$V = \text{Array}(nD, nS, nIT)$$

$Pb$  and  $Gb$  are personal best array dan global best array,

$$Pb = \text{Array}(nD, nS, nIT)$$

$$Gb = \text{Array}(nD, 1, nIT)$$

$F$  is array of the fitness value of swarm. The fitness value is integral error of  $e(t) = y_d(t) - y(t)$ .

$$F = \text{Array}(1, nS, nIT)$$

The fitness value is defined as follows.

$$IAE = \int_{t_0}^{t_f} |y_d(t) - y(t)| dt \tag{15}$$

### Prosedur PSO

1. First step (initialization).

Generate the intial position and velocity of swarm randomly [13].

$$X_1^i = X_{\min} + \text{rand}(0, 1) \cdot (X_{\max} - X_{\min}) \tag{16}$$

$$V_1^i = V_{\min} + \text{rand}(0, 1) \cdot (V_{\max} - V_{\min}) \tag{17}$$

Generate the value of *inertia weight* [14]

$$W = W_{\max} - \frac{W_{\max} - W_{\min}}{k_{\max}} \times k \tag{18}$$

2. Calculate the fitness value of  $X$ 
  - (a) Solve the equation (7) with control law (8) or (13).
  - (b) Calculate the fitness value :

$$IAE = \int_{t_0}^{t_f} |y_d(t) - y(t)| dt. \quad (19)$$

3. Determine the personal and global best.
4. Update the swarm position

$$X_{k+1}^i = X_k^i + V_{k+1}^i \quad (20)$$

where

$$V_{k+1}^i = V_{k+1}^i + \psi \cdot c_1 (Pb_k^i - X_k^i) + \psi \cdot c_2 (Gb_k - X_k^i). \quad (21)$$

5. Repeat the second step until stopping criteria is satisfied.

## 5 Example

**Example 1.** Consider the following SISO affine nonlinear control 16 system

$$\begin{aligned} \dot{x}_1 &= x_2 + 2x_1^2 \\ \dot{x}_2 &= x_3 + u \end{aligned} \quad (22)$$

$$\begin{aligned} \dot{x}_3 &= x_1 + x_3 \\ y &= x_1; \quad y_d(t) = \sin t. \end{aligned} \quad (23)$$

This system satisfies Theorem 1. Thus, using the output  $\lambda(x) = x_3$ , the nonlinear system (14) can be linearized exactly.

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= a(\mathbf{z}) + u, \end{aligned} \quad (24)$$

where  $a(\mathbf{z}) = z_1 + z_2 + (2(z_2 - z_1) + 1)(z_3 - z_2 - 2(z_2 - z_1)^2 + 2(z_2 - z_1)^2)$ . By input-output linearization technique we get

$$u = -a(\mathbf{z}) + v. \quad (25)$$

Let  $y_d(t) = \dot{z}_1(t) = x_{1d}(t)$ . Next, we choose  $z_{1d}(t)$  such that if  $z_1(t)$  tracks  $z_{1d}(t)$  then  $y(t)$  tracks the desired output  $y_d(t)$ . Consider the equation :

$\dot{x}_3 = x_1 + x_3$ . By replacing  $x_1$  with  $x_{1d}(t) = \sin(t)$ , we have a differential equation  $\dot{x}_3 - x_3 = \sin(t)$ . Then, we solve the differential equation to obtain  $x_3 = 1/2(-\sin(t) - \cos(t))$ . This solution we state as  $x_{3d}(t) = 1/2(-\sin(t) - \cos(t))$ . Thus, for the output tracking problem we have

$$v = \frac{1}{a_3} \dot{z}_{3d} - \sum_{i=1}^3 a_{i-1}(z_i - z_{id}). \tag{26}$$

The simulation is done as follows.

First, we approximate the  $x_{3d}(t)$  as a fourier series

$$= \alpha + \sum_{i=1}^N \beta_i \sin(t) + \gamma_i \cos(t).$$

The value of parameters  $\alpha, \beta_i, \gamma_i, i = 1, 2, 3$ . are determined by PSO. To do PSO, we use the value of parameter as follows.

- Initial value of system,  $x_0 = (1 \ 1 \ 1)^T$
- Interval  $t = [0, 50]$
- The number of swarm,  $nS = 20$
- The cognitive value,  $c_1 = 1.4$  and  $c_2 = 1.4$  [12]
- The number of equation,  $nD = 7$ .
- The maximum iteration,  $nIT = 100$

In Table 1, the value of  $\alpha, \beta_i, \gamma_i, i = 1, 2, 3$  are given which are obtained by PSO.

Table 1: The value of parameters

No	Parameter	Value
1	$\alpha$	0.0009
2	$\beta_1$	0.1194
3	$\beta_2$	-0.4662
4	$\beta_3$	-0.1533
5	$\gamma_1$	-0.5194
6	$\gamma_2$	0.6166
7	$\gamma_3$	-0.5973

The second, we compare  $x_{3,d}$  analytic and  $x_{3,d}$  approximation (the approximation result at the end of iteration) and then we calculate IAE of  $x_{3,d}$  analytic and  $x_{3,d}$  approximation. The results are given in Figure 1 and Figure 2.

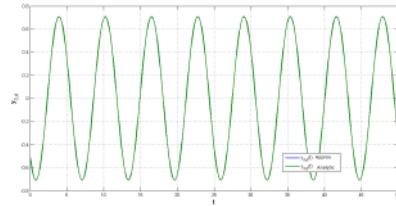


Figure 1: Comparison :  $y_{3,d}$  Analytic and  $y_{3,d}$  approximation

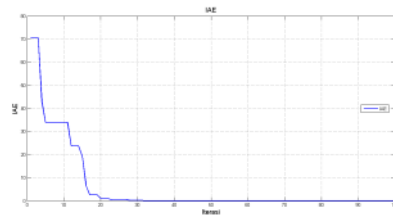


Figure 2: IAE of  $y_{3,d}$  analytic and  $y_{3,d}$  approximation.

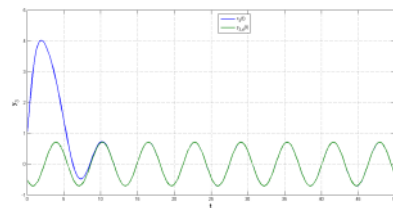


Figure 3: Output Tracking (exact linearization),  $x_3$  to  $x_{3d}$

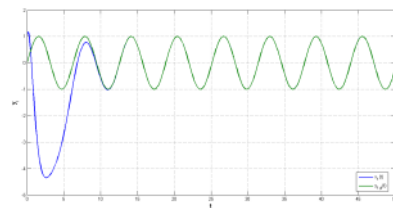


Figure 4: Output Tracking (original system)  $y$  to  $y_d$

Then, third, we solve the equation (22) with control law  $u$  in equation (25). The simulation results are given in Figure 3 and Figure 4.

**Example 2.** Consider the following nonlinear system equation (SISO)

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2x_3 \\ \dot{x}_2 &= x_3 + u \\ \dot{x}_3 &= x_1 + x_3 \\ y &= x_1; \quad y_d(t) = \sin t. \end{aligned} \tag{27}$$

The nonlinear systems (27) has relative degree 2 at any point  $\mathbf{z}^0$  (well defined). In normal form, the nonlinear system (27) becomes

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= \frac{\xi_1}{\eta}(\xi_1 + \xi_2 + \eta) + \eta^2 + \eta u \\ \dot{\eta} &= \xi_1 + \eta \end{aligned}$$

Because the stability of zero dynamics is unstable, the nonlinear system (27) is nonminimum phase. But the system in equation (27) satisfies Theorem 1.

By choosing  $y = \lambda(x) = x_3$ , the nonlinear system (27) can be linearized exactly.

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= a(\mathbf{z}) + b(\mathbf{z})u, \end{aligned} \tag{28}$$

where  $a(\mathbf{z}) = x_3^2 + x_1 + x_3 + x_1x_2 + x_2x_3$ ,  $b(\mathbf{z}) = x_3$ . The input-output linearization technique can not be applied to this systems because the value of  $b(\mathbf{z}) = x_3$  can be zero for some  $\mathbf{z}$ .

Let  $y_d(t) = x_{1d}(t) = \sin(t)$ . Next, we choose  $z_{1d}(t)$  such that if  $z_1(t)$  tracks  $z_{1d}(t)$  then  $y(t)$  tracks the desired output  $y_d(t)$ .

Consider the equation:  $\dot{x}_3 = x_1 + x_3$ . By replacing  $x_1$  with  $x_{1d}(t) = \sin(t)$  we have a differential equation  $\dot{x}_3 = \sin(t) + x_3$ . Then, we solve the differential equation to get  $x_3 = 1/2(-\sin(t) - \cos(t))$ . This solution we state as  $x_{3d}(t) = 1/2(-\sin(t) - \cos(t))$ . Thus, for the output tracking problem we have

$$\begin{aligned} u_r(t) &= (1/x_3(t))[y_d^{(3)}(t) - (a_0(z_1(t)) - y_d(t)) \\ &\quad + a_1(z_2(t) - \dot{y}_d(t)) + a_2(z_3(t) - \ddot{y}_d(t)) \\ &\quad + a_2(z_3(t) - \ddot{y}_d(t)) - (x_3(t)^2 + x_1(t)) \\ &\quad + x_3(t) + x_1(t)x_2(t) + x_2(t)x_3(t)]. \end{aligned} \tag{29}$$

The control law  $u(t)$  (23) is valid only if  $x_3(t) \neq 0, \forall t$ . Let  $\mathbf{x}(t_s)$  be a singular point, then substitute  $\mathbf{x}(t_s)$  into the equation (16) (take  $m = 3$ ), to obtain

$$\begin{aligned} y_d^{(4)}(t_s) &= x_3(t_s) + 3x_1(t_s)x_3(t_s) + 2x_2(t_s)x_3(t_s) \\ &\quad + 3x_3^2(t_s) + x_2^2(t_s)x_3(t_s) + 2x_1(t_s)u(t_s) \\ &\quad + 2x_3(t_s)u(t_s) + x_3(t_s)\dot{u}(t_s), \end{aligned} \quad (30)$$

$$\begin{aligned} y_d^{(5)}(t_s) &= x_1(t_s) + x_3(t_s) + 3x_1^2(t_s) + 9x_1(t_s)x_3(t_s) \\ &\quad + 2x_1(t_s)x_2(t_s) + 2x_2(t_s)x_3(t_s) + 8x_3^2(t_s) \\ &\quad + 5x_2(t_s)x_3^2(t_s) + x_1(t_s)x_2^2(t_s) \\ &\quad + x_3(t_s)x_2^2(t_s) + 4x_3(t_s)u(t_s) + 4x_2(t_s)x_3(t_s)u(t_s) \\ &\quad + 3x_1(t_s)\dot{u}(t_s) + 3x_3(t_s)\dot{u}(t_s) + x_3(t_s)\ddot{u}(t_s), \end{aligned} \quad (31)$$

$$\begin{aligned} y_d^{(6)}(t_s) &= x_3(t_s) + 11x_1x_3(t_s) + 3x_2(t_s)x_3(t_s) \\ &\quad + 3x_2^2(t_s)x_3(t_s) + x_2^3(t_s)x_3(t_s) \\ &\quad + 8x_1(t_s)x_2(t_s)x_3(t_s) + 9x_1^2(t_s)x_3(t_s) \\ &\quad + 2\dot{x}_2(t_s)x_3(t_s) + 16\dot{x}_3(t_s)x_3(t_s) \\ &\quad + 5\dot{x}_2(t_s)x_3^2(t_s) + 10x_2(t_s)x_3(t_s)\dot{x}_3(t_s) \\ &\quad + 2x_2(t_s)x_3(t_s)\dot{x}_2(t_s) + 3x_1^2(t_s) \\ &\quad + 6x_1(t_s)u(t_s) + 6x_1(t_s)x_2(t_s)u(t_s) + 4x_3(t_s)(\dot{u}(t_s) \\ &\quad + u(t_s)) + 7x_2(t_s)x_3(t_s)\dot{u}(t_s) + 4x_3(t_s)u(t_s)\dot{x}_2(t_s) \\ &\quad + 4x_2(t_s)x_3(t_s)u(t_s) + 4x_1(t_s)\ddot{u}(t_s) \\ &\quad + 3x_3(t_s)\dot{u}(t_s) + 4x_3(t_s)\ddot{u}(t_s) + x_3(t_s)\ddot{\ddot{u}}(t_s) \end{aligned} \quad (32)$$

Solve the equation (22)-(24), to find the value of  $u^{(i)}(t_s)$ ,  $i = 0, 1, 2$ . Finally, we have

$$u_{ps}(t) = \sum_{i=0}^2 \frac{u^{(i)}(t_s)}{i!} (t - t_s)^i, \quad t \in T_s. \quad (33)$$

The simulation is done as follows.

First, we approximate the  $x_{3d}(t)$  as a fourier series

$$= \alpha + \sum_{i=1}^N \beta_i \sin(t) + \gamma_i \cos(t).$$

The value of parameters  $\alpha, \beta_i, \gamma_i, i = 1, 2, 3$  are determined by PSO. To do PSO, we use the value of parameter as in example 1. In Table 2, the value of  $\alpha, \beta_i, \gamma_i, i = 1, 2, 3$  are given which are obtained by PSO.