THESIS APPROVAL

GRADUATE SCHOOL, KASETSART UNIVERSITY

Master of Engineering (Information and Communication Technology for Embedded Systems)

DEGREE

Information and Communication Technology for Embedded Systems | Electrical Engineering

FIELD | **DEPARTMENT**

TITLE: Improved Adaptive Sliding Mode Controller Design and Parameter Identification using Particle Swarm Optimization for Linear Permanent Magnet Synchronous Motor

NAME: Mr. Tithiwat Therdbankerd

THIS THESIS HAS BEEN ACCEPTED BY

THESIS ADVISOR

(Assistant Professor Peerayot Sanposh, D.Sc.)

THESIS CO-ADVISOR

(Mr. Nattapon Chayopitak, Ph.D.)

THESIS CO-ADVISOR

(Associate Professor Hideaki Fujita, Dr.Eng.)

DEPARTMENT HEAD

(Assistant Professor Srijidtra Charoenlarpnopparut, Ph.D.)

APPROVED BY THE GRADUATE SCHOOL ON _____

DEAN

(Associate Professor Gunjana Theeragool, D.Agr.)

THESIS

IMPROVED ADAPTIVE SLIDING MODE CONTROLLER DESIGN AND PARAMETER IDENTIFICATION USING PARTICLE SWARM OPTIMIZATION FOR LINEAR PERMANENT MAGNET SYNCHRONOUS MOTOR

TITHIWAT THERDBANKERD

A Thesis Submitted in Partial Fulfillment of
the Requirements for the Degree of
Master of Engineering (Information and Communication Technology for Embedded Systems)
Graduate School, Kasetsart University
2010

Tithiwat Therdbankerd 2010: Improved Adaptive Sliding Mode Controller Design and Parameter Identification using Particle Swarm Optimization for Linear Permanent Magnet Synchronous Motor. Master of Engineering (Information and Communication Technology for Embedded Systems), Major Field: Information and Communication Technology for Embedded Systems, Department of Electrical Engineering. Thesis Advisor: Assistant Professor Peerayot Sanposh, D.Sc. 100 pages.

Accurate and effective parameter identification is an important engineering task in high performance control system design. One emerging approach to effectively identify such nonlinear or dynamic unknown parameters is to use Particle Swarm Optimization (PSO) algorithm. Linear Permanent Magnet (LPM) motor is a high performance actuator employed in many applications that require direct linear motion without mechanical transmission for high acceleration and accurate positioning.

However, the knowledge of the LPM motor parameters alone do not guarantee that the nominal values of the LPM motor parameters are sufficient for all controllers. A sliding mode control, which has a simple structure and robustness properties, can overcome the uncertain parameter variations and external disturbances. Although the adaptive sliding mode controller can effectively reduce some of the chattering phenomena, it is still susceptible to large external disturbances and large parameter variations.

Therefore, this thesis proposes: (1) a simple PSO based method with chirp input signals to identify the LPM motor's parameters and (2) an improved adaptive sliding mode controller to reduce the chattering phenomena to control the LPM motor.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

ACKNOWLEDGEMENTS

I would like to gratefully thank Asst. Prof. Peerayot Sanposh who is my thesis advisor for his partial financial support, advice, encouragement and valuable suggestions for writing conference papers and this thesis,.

I would like to sincerely thank Dr. Nattapon Chayopitak who is my thesis co-advisor at NECTEC for his advice and valuable suggestions for writing conference papers and this thesis.

I would like to thank Asst. Prof. Hideaki Fujita who is also my thesis co-advisor in Japan for his valuable comments and suggestions.

I would like to thank my friends in the IMARC laboratory: Jin, Gingo, Day and Kook and ICTES friends, especially, Mr. Chaiwat Suwansaroj who convince me to apply for this program.

This research was financially supported by Thailand Advanced Institute of Science and Technology – Tokyo Institute of Technology (TAIST-Tokyo Tech), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology (Tokyo Tech), and Kasetsart University (KU).

Finally, I deeply appreciate my parents and my sister for their continuous encouragements and support throughout my education.

Tithiwat Therdbankerd

April 2010

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

Figure **Page**

Appendix Figure

LIST OF FIGURES (Continued)

Appendix Figure **Page**

LIST OF ABBREVIATIONS

$\ddot{x}_p$	=	Acceleration of the LPM motor
x_{pi}	=	Actual position
A	=	Amplitude
BMC	=	Baseline Model Control
U_{BMC}	=	Baseline Model Control
c_1, c_2	=	Cognitive coefficients
U_{c}	=	Control variable is used to cancel the nonlinear terms in the model
U_{s}	=	Control variable is used to specify the desired system performance
$U, U(t)$	=	Control variable
U_{b}	=	Curbing Control
i_d	=	Current in d-axis reference frame
i_q	=	Current in q-axis reference frame
$i_q^*(t)$	=	Current reference input in q-axis reference frame
ρ	=	Control gain
DAC	=	Digital to Analog Converter
$\hat{\rho}$	=	Estimated value of ρ
F_L	=	External force
t_f	=	Final time
Fuzzy-PI	=	Fuzzy-Proportional-Integral
GHz	=	gigahertz (unit)
$g_{\mathrm{best_i}}$	=	Global best position of the entire population
Hz	=	Hertz (unit)
$\tilde{U}_{\mathrm{b}}$	=	Improved adaptive sliding mode control
L_d	=	Inductance in d-axis reference frame
L_q	=	Inductance in q-axis reference frame
L	=	Inductance
w	=	Inertia weight factor

LIST OF ABBREVIATIONS (Continued)

$\mathbf{E_0}$	=	Initial state of $\mathbf{E}$
kg	=	kilogram (unit)
$\tilde{M}$	=	Known adding mass of the payload
λ	=	Learning rate of the adaptive algorithm
LPM motor	=	Linear Permanent Magnet synchronous motor
ω_1	=	Lower bound frequency
$W(t)$	=	Lumped uncertainty
MAE	=	Mean Absolute Error
mm	=	millimeter (unit)
p_1', p_2' ,p_3'	=	Motor constant with additional mass
F_e	=	Motor force
K_f	=	Motor force constant
p_1, p_2, p_3	=	Motor parameters without payload
N	=	Newton
N/A	=	Newton per Ampere
Ns/m	=	Newton second per meter
$\bar{M}$	=	Nominal value of the moving mass without payload
$\bar{M}$	=	Nominal value of the moving mass without payload
ϵ	=	Non zero positive constant
n	=	Number of data points
p	=	Number of pole pairs
n	=	Number of the sampling data
$\hat{y}_i$	=	Output response of i^{th} sampling data of estimated model
y_i	=	Output response of i^{th} sampling data of real system
PSO	=	Particle Swarm Optimization
PCI	=	Peripherals Component Interconnect
λ_f	=	Permanent magnet flux
PMSM	=	Permanent Magnet Synchronous Motor
p_{best_i}	=	Personal best position of particle

LIST OF ABBREVIATIONS (Continued)

R_s	=	Phase winding resistance
τ	=	Pole pitch
x_i	=	Position of i^{th} particle
x_p	=	Position of the LPM motor
x_p^*	=	Position reference input
$\hat{x}_p$	=	Position response of the estimated model
x_p	=	Position response of the real system
x_{pi}^*	=	Position reference input
W	=	Positive definite matrix
PID	=	Proportional-Integral-Derivative
PRBS	=	Pseudo Random Binary Signal
γ_1, γ_2	=	Random numbers in $[0, 1]$
RTI	=	Real Time Interface
R&D	=	Research and Development
R	=	Resistance
s, sec	=	second
$sgn(\cdot)$	=	Signum function
$S(t)$	=	Sliding surface
M	=	Total mass of the moving part
ω_2	=	Upper bound frequency
y	=	Vector of output variables
$\hat{y}$	=	Vector of output variables of the estimated model
x	=	Vector of state variables
$\hat{x}$	=	Vector of state variables of the estimated model
u	=	Vector of system inputs
$\hat{p}$	=	Vector of the estimated unknown parameters
p	=	Vector of unknown parameters to be identified
$C(\mathbf{E})$	=	Vector to be designed
$\dot{x}_p$	=	Velocity of the LPM motor

LIST OF ABBREVIATIONS (Continued)

v_i	=	Velocity of i^{th} particle
$\dot{x}_p$	=	Velocity response of the real system
$\hat{\dot{x}}_p$	=	Velocity response of the estimated model
B	=	Viscous friction coefficient
v_d	=	Voltage in d-axis reference frame
v_q	=	Voltage in q-axis reference frame
w_{x_p}	=	Weight factor of the position response
$w_{\dot{x}_p}$	=	Weight factor of the velocity response

IMPROVED ADAPTIVE SLIDING MODE CONTROLLER DESIGN AND PARAMETER IDENTIFICATION USING PARTICLE SWARM OPTIMIZATION FOR LINEAR PERMANENT MAGNET SYNCHRONOUS MOTOR

INTRODUCTION

Linear motors are electric motors that can operate without conventional gears, screws or crack shafts. Linear motors have been widely used in high performance motion control applications demanding faster speed and better accuracy, such as computer-controlled machine tools, semiconductor manufacturing equipments and inspection machines. Linear synchronous motor in particular is a linear motor whose mechanical motion is in synchronism with the magnetic field produced by electromagnets or permanent magnets. In general, Linear Permanent Magnet synchronous (LPM) motors have been widely used in many applications due to their large torque coefficient and high efficiency without conventional gears.

However, the design of high-performance LPM motor control system depends on load variation of the plant, mechanical friction and parasitic effects. In order to achieve high-performance controller design, some knowledge of LPM motor parameters is necessary to implement advanced control algorithms, but the motors' manufacturer usually does not provide all necessary information and some parameters often fluctuate during operation or are difficult to measure in practice. One emerging approach to effectively identify such nonlinear or dynamic unknown parameters is to use Particle Swarm Optimization (PSO) algorithm. The parameter identification method based on PSO has advantages of fast convergence and simplicity when an appropriate test input signal is chosen such as the well-known pseudo random binary signal (PRBS) to extract the characteristics of the system. Several research works have reported successful applications of PRBS for identifying parameters of Permanent Magnet Synchronous Motors (PMSM). Although, PMSM and LPM motors share several magnetic features, and many design and control concepts are transferable, unfortunately this is not the case for parameter identification techniques.

However, PRBS inputs are not suitable for parameter identification of linear motors with finite traveling track length. PRBS is generated in random sequence for extended excitation time resulting in long travel distance and may cause the mover to hit the end of the track resulting in physical damages. Hence, to effectively identify the LPM motor parameters, the chirp signal is proposed in this research.

Nevertheless, the knowledge of the LPM motor parameters alone do not guarantee that the nominal values of the LPM motor parameters are sufficient for all controllers. An adaptive sliding mode controller, which has simple structure and robustness against external disturbances and parameter variations, is considered. Although the adaptive sliding mode controller can effectively reduce some of the chattering phenomena, it is still susceptible to large external disturbances and large parameter variations.

Therefore, this thesis investigates the following two subjects:

1. Parameter identification using PSO to estimate the LPM motor parameters with chirp input signals.
2. Improved adaptive sliding mode controller design to effectively control and to reduce the chattering phenomena of the LPM motor during operation.

The proposed methods have simple structure and easy to implement. The effectiveness of the proposed methods is verified by the experiments.

OBJECTIVES

1. To study and implement a parameter identification method to effectively identify the LPM model using PSO algorithm.

2. To study and improve the sliding mode control to effectively control the LPM motor.

LITERATURE REVIEW

Many control algorithms for LPM motors are based on the control structures for rotary PMSM. To control the LPM motor, a simple method in industrial practice is based on adjusting Proportional-Integral-Derivative (PID) control gains. In high precision and high performance control system, vector oriented control or field oriented control method based on adjusting both amplitudes and phases of the controlled variables is used to control the force of the LPM motor by decoupling the active and reactive currents in the dq-axis reference frame. In general, accurate parameters are significant for high-performance control system designs. In order to acquire the accurate parameters of the LPM motors, several parameter identification techniques have been investigated for different system. In particular, the use of Genetic Algorithm to identify the parameters of PMSM is reported (Sudhoff *et al.*, 2005). In (Liu *et al.*, 2008) Particle Swarm Optimization (PSO) has been proposed to directly identify the parameters of PMSM. Although, PMSM and LPM motors share several magnetic features, and many design and control concepts are transferable, unfortunately this is not the case for parameter identification techniques. The accuracy of parameter identification techniques rely on the appropriate choice of the test input signals (Ljung, 1987), such as the well known pseudo random binary signal (PRBS), that can excite as many modes for extracting as many features of the system. Several studies, for instance (Hasni *et al.*, 2008), (Michalik, 1998), and (Pacas *et al.*, 2010) have reported successful applications of PRBS inputs for PMSM.

However, PRBS inputs are not suitable for parameter identification of motors with finite traveling track length. PRBS inputs usually require extended excitation time for extended travel distance; this problem is not encountered for PMSM systems since the motor essentially has unlimited travel distance, in other words infinite rotation. Since, linear motor drive systems found in manufacturing applications usually have limited length traveling tracks (e.g. 0.5 2 m), PRBS is essentially not suitable for parameter identification of the LPM motor system as the input command for extended travel distance may cause the mover to hit the end of the track resulting in physical damages.

Literature on parameter identification for linear motors is limited. A parameter auto tuning technique for identifying the parameters of the LPM motor is proposed (Hsu *et al.*, 2000), but the method is complicated to implement. This research proposes a parameter identification method based on PSO using chirp signal and simple procedure of placing two different payloads to the system to effectively identify the mechanical parameters of the LPM motor. The parameter identification method based on PSO has advantages of fast convergence and simplicity when appropriate test input signal is chosen. The proposed method may be used with any systems with any input signals, but the chirp signal is recommended for linear motor systems.

In general, a parameter identification method only determines the nominal values, but actual values of these parameters usually deviate from their nominal values due to external disturbances during operation. If the control system does not account for these situations, low-performance system or instability might occur. To overcome this problem, many control algorithms have been considered such as Fuzzy-PI control (Hsu *et al.*, 2001), adaptive control (Liu *et al.*, 2004) and sliding mode control (Utkin *et al.*, 1999). In particular, the sliding mode control has simple structure and robustness against uncertain perturbations and external disturbances.

Traditional sliding mode control has two major weaknesses: (i) the control system is not robust while the system is still on the reaching phase before entering the sliding surface, and (ii) the system is suffered from the chattering phenomena after reaching the sliding surface. One approach to solve the first problem is the total sliding mode control (Shyu *et al.*, 1999) where the reaching phase is eliminated by initializing the control system on the sliding surface. The adaptive sliding control (Wai, 2000) is proposed to alleviate the second problem by utilizing online an parameter tuning technique to reduce the chattering.

Although the adaptive sliding mode control has simple structure and can effectively reduce some of the chattering phenomena, it is still susceptible to large external disturbances and large parameter variations. The total sliding mode with

Neural Networks (Wai, 2001) and the sliding mode control with real time Genetic algorithm (Chou, 2003) are later proposed to address such problems; however, these methods require high performance processor for complicate computation.

Therefore, another aspect of this thesis is to investigate an improved adaptive sliding mode control using the saturation function to effectively reduce the chattering phenomena. The proposed technique is based on the adaptive sliding mode control to maintain the simple structure and computational efficiency. The simulations and experiments have been conducted to control the LPM motor drive system under large parameter variations to demonstrate the effectiveness of the proposed method.

Description of Linear Permanent Magnet motor

Linear motors are electric motors that can operate without conventional gears, screws or crack shafts to provide direct linear motion. Linear Permanent Magnet (LPM) motors, particularly, have been widely used in many applications demanding faster speed and better accuracy such as computer-controlled machine tools, semiconductor manufacturing equipments and inspection machines. To control the LPM motor, there are several methods; scalar control method based on changing only the amplitudes of the controlled variables are to control force of LPM motor constantly, vector oriented control or field oriented control method based on changing both amplitudes and phases of the controlled variables, as shown in Figure 1, are to also control force of LPM motor by decoupling active and reactive currents in other word dq-axis reference frame. Normally, the vector oriented control methods have fix structure and constant parameters. However the LPM parameters might be deviated while the LPM motors are operating. Although the use of the adaptive control can adjust the parameters of the LPM motor control system, it is still susceptible to large external disturbances and large parameter variations. Therefore, the sliding mode control is one suitable (Gieras and Piech, 1999) of a variable structure control technique that has simple structure and robustness against uncertain perturbations and external disturbances to control in the LPM motor control system. For the sliding

mode control, the mathematical model of the LPM motor, as shown in Figure 1, in the dq-axis reference frame is given by (Boldea and Nasar, 1997)

$$\dot{i}_d = -\frac{R_s}{L_d} i_d + \frac{\pi}{\tau}\frac{L_q}{L_d}\dot{x}_p i_q + \frac{1}{L_d} v_d \tag{1}$$

$$\dot{i}_q = -\frac{L_d}{L_q}\dot{x}_p i_d - \frac{R_s}{L_q} i_q - \frac{\pi}{\tau}\frac{\lambda_f}{L_q}\dot{x}_p + \frac{1}{L_q} v_q \tag{2}$$

$$\begin{aligned} F_e &= \frac{3}{2}\frac{\pi}{\tau} p\lambda_f \\ &\triangleq K_f i_q \end{aligned} \tag{3}$$

where x_p and $\dot{x}_p$ are the position and velocity of the LPM motor;
p is the number of pole pairs;
τ is the pole pitch; F_e is the motor force;
v_d and v_q are the voltages in the dq-axis reference frame;
i_d and i_q are the currents in the dq-axis reference frame;
R_s, L_d, L_q and λ_f are the phase winding resistance, the d and q-axis inductances, and the permanent magnet flux, respectively;
K_f is the motor force constant.

The mechanical system of the LPM motor is composed of the mover and its payload described by

$$F_e = M\ddot{x}_p + B\dot{x}_p + F_L \tag{4}$$

where

F_L : the external force

M : the total mass of the moving part

B : the viscous friction coefficient

$\ddot{x}_p$: the acceleration

Since the force produced by the LPM motor only depends on i_q, it is customary to control $i_d = 0$ (Gieras and Piech, 1999) resulting in i_q chosen as the input of the mechanical system. Hence, (3) and (4) can be rewritten into

$$K_f i_q = M\ddot{x}_p + B\dot{x}_p + F_L \tag{5}$$

Normally, some parameters are provided by motor manufacturers but they are usually not sufficient for high-performance applications, due to parameter variations and various disturbances. Some parameters often fluctuate during operation and are difficult to measure in practice. For example, the motor force constant, K_f, the viscous friction coefficient, B, and the external force, F_L, cannot directly be measured whereas M cannot be easily determined if the LPM motor is already installed in a larger system such as a gantry robot.

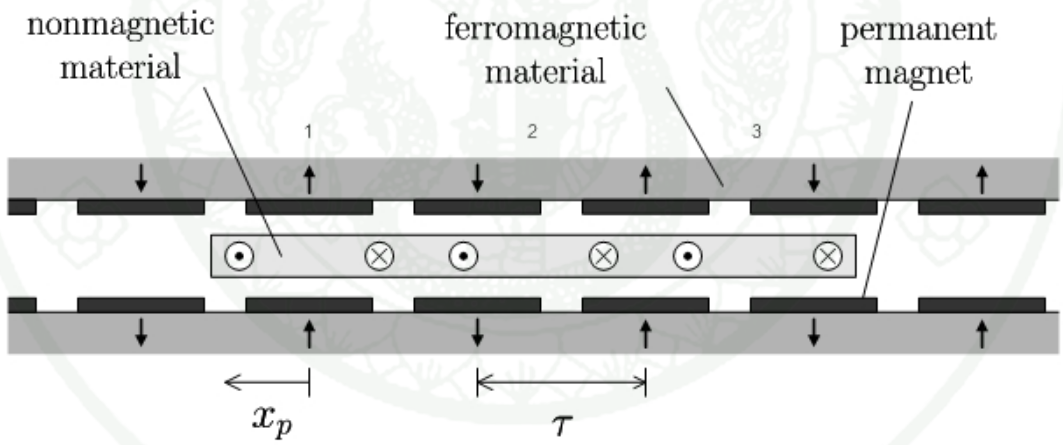


Figure 1 The LPM motor structure

General parameter identification

In order to effectively control the LPM motor, we have to know the parameter of the LPM motor. A method to identify the parameter is called “Parameter Identification”'. In practice, there are several methods to identify the parameter in the system. Especially, in the LPM motor, there are several parameters to identify such as electrical parameters; resistance (R), inductance (L) and/or mechanical parameters; motor force constant (K_f), moving mass part (M) and viscous coefficient (B), and etc. Normally, some parameters are provided by motor manufacturers but they are usually not sufficient for high-performance applications, due to parameter variations and various disturbances. Some parameters often fluctuate during operation and are difficult to measure in practice. For example, the motor force constant, K_f, the viscous friction coefficient, B, and the external force, F_L, cannot directly be measured whereas M cannot be easily determined if the LPM motor is already installed in a larger system such as a gantry robot. Sometimes these parameters have to be measured by particularly instrument, e.g. the motor force constant has to be the force meter and current meter to measure then to estimate its value. One approach to identify the LPM motor's parameters is the heuristic algorithm or stochastic search method. The heuristic algorithm is the algorithm that can be able to find the optimal solution under given constraint, such as time, in practice. However, this algorithm is not formally proven the optimal solution. The heuristic algorithm has two categories for identifying model.

1. Non-parametric model

Non-parametric model uses the relationship between the inputs and the outputs of the systems to predict the mathematical model including model structure and parameters' value. For this method, it is not necessary to specify the mathematical model or model structure or parameters' value in this method.

2. Parametric model

Parametric model has to know the specific mathematical model or model structure to predict the parameters' value from the relationship between the inputs and the outputs. This method uses only to predict the best parameters' value in the specific mathematical model.

In this thesis, the parameter identification based on parametric model is investigated for both simulation and experiment. The method investigated here follows the approach proposed (Liu *et al.*, 2008) in order to identify the LPM motor's parameters. The model structure is constrained to identify the model parameters. Consider a general dynamical system that represents the “real system”' (e.g. the LPM motor).

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

$$y = g(p, x) \tag{7}$$

where x is the vector of state variables; u is the vector of system inputs; p is the vector of unknown parameters to be identified; and y is the vector of output variables. Next consider an estimate model that is used to predict the behavior of the “real system”

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

$$\hat{y} = g(\hat{p}, \hat{x}) \tag{9}$$

where $\hat{x}$ is the vector of state variables of the estimated model; $\hat{y}$ is the vector of output variables of the estimated model; $\hat{p}$ is the vector of the estimated unknown parameters; Note that u is the same for (6) and (8) while $f(\cdot)$ and $g(\cdot)$ represent the same system structure. The block diagram of the parameter identification based on heuristic algorithm can be divided into two steps:

1. Operating & Capturing data

First step in Figure 2, to operate the system by using the system input excites to obtain the system response.

2. Identification

Second step in Figure 3, to identify the captured data from previous step until to obtain the estimated parameters. This step might be repeatedly performed to identify to the estimated parameters in many times.

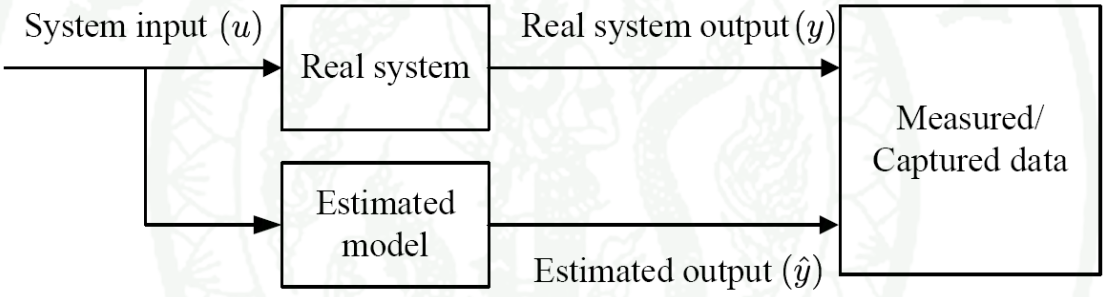


Figure 2 First step of block diagram of parameter identification procedures

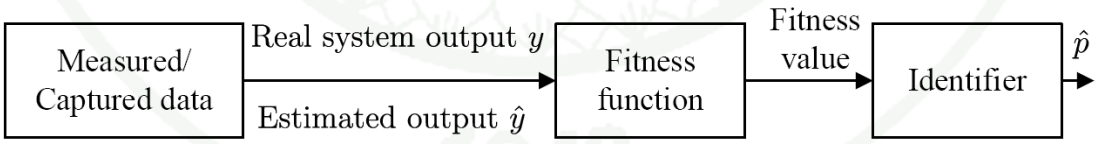


Figure 3 Second step of block diagram of parameter identification procedures

Difficulty factors

To evaluate, however, the difficulty of estimated parameters depends on three factors: system input selection u, fitness function and identifier that should be carefully selected.

1. Input selection

For the PSO identification method, the accuracy of the estimated parameters depends on the appropriate system input that should sufficiently excite the LPM motor model. From trial and error, the chirp signal is more capability to excite the LPM motor model than the step signal. Thus, the chirp signal is chosen for parameter identification in real experiment. The chirp signal is a sinusoidal function with varying frequency ω over a time period $0 \leq t \leq t_f$ that the equation is given by

$$u(t) = A\cos(\frac{\omega_1 t + (\omega_2 - \omega_1)t^2}{2t_f}) \tag{10}$$

where A is the amplitude; ω_1 and ω_2 are the lower and upper bound frequencies; t_f is the final time.

2. Fitness function

Clerc said, “When high precision is required, the probability of failure is very high and to take it directly as a measure of difficulty is not very practical.” It suggests that we will use instead a logarithmic measurement given by the following formula:

$$\text{difficulty} = -\ln(1-\text{failure probability}) = -\ln(\text{success probability}) \tag{11}$$

For this way, in Table 1, we can determine which fitness functions are suitable for optimization problem.

Table 1 Difficulty of three problems compared

Problem	Search space	Value to be reached	Admissible error	Logarithmic difficulty
$\sum_{d=1}^{10} x_d$	$[\ 0 \quad 1\ ]^{10}$	0	0.01	61.2
$\sum_{d=1}^{10} x_d^2$	$[\ 0 \quad 1\ ]^{10}$	0	0.01	29
$\sum_{d=1}^{10} \sqrt{x_d \lvert \sin x_d \rvert}$	$[\ 0 \quad 1\ ]^{10}$	0	0.01	Estimate 63

where the probabilities of success are compared in logarithm. It is easily show that the lower difficulty's value is more the success probability. Thus, the fitness function should be formulated as the sum of squares when high precision result is desired, such as

$$C(\hat{p}) = \int_0^t (y - \hat{y})^T W (y - \hat{y}) dt \tag{12}$$

where W is a positive definite matrix; y, $\hat{y}$ are the real vector output and estimated vector output, respectively.

3. Identifier

Identifier or Intelligent control unit is used to automatically identify the parameters, $\hat{p}$. This unit cooperates with fitness function for identification. For identifier, there are several methods and applied algorithms for parameter identification, i.e. Neuron Network can be modified its algorithm for parameter identification, Genetic Algorithm or Particle Swarm Optimization is a based on search technique to normally find the solution in optimization problems. These algorithms

have the different strong points, i.e. Particle Swarm Optimization has the strong points in term of fast computation, fast convergence, less configuration parameters.

Non-unique structure model

Although the parameter identification based on parametric model can overcome the optimization problem finding solution, the problem occurs when the finding parameter in the structure model is the non unique solution. Considering the LPM motor model as shown in Figure 4, even though the nominal value of the LPM parameters are $K_f = 1$, $M = 1$, $B = 1$, there are several solutions for this model, i.e. $K_f = 2$, $M = 2$, $B = 2$ or $K_f = 3$, $M = 3$, $B = 3$. These parameters are the same solution. To solve a problem, we propose a technique to identify the actual parameters of the LPM motor parameters.

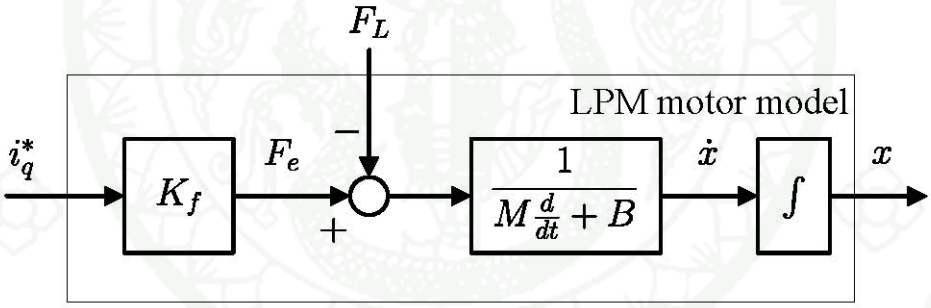


Figure 4 LPM motor control system diagram

Consider (5) when no payload is attached to the system so that $M = \bar{M}$; then

$$\begin{aligned} i_q &= \frac{\bar{M}}{K_f}\ddot{x}_p + \frac{B}{K_f}\dot{x}_p + \frac{F_L}{K_f} \\ &\triangleq p_1\ddot{x}_p + p_2\dot{x}_p + p_3 \end{aligned} \tag{13}$$

where p_1, p_2 and p_3 are the motor parameters without payload; $\bar{M}$ is the nominal value of the moving mass without payload. Next consider when a payload is attached to the original system so that $M \rightarrow M' = \bar{M} + \tilde{M}$

$$\begin{aligned} i_q &= \frac{M'}{K_f}\ddot{x}_p + \frac{B}{K_f}\dot{x}_p + \frac{F_L}{K_f} \\ &\triangleq p_1'\ddot{x}_p + p_2'\dot{x}_p + p_3' \end{aligned} \tag{14}$$

where $p_1\prime$, $p_2\prime$ and $p_3\prime$ are the motor constants with additional mass; $\tilde{M}$ is a known adding mass of the payload. Note that the motor force constant K_f does not depend on the payload hence it remains unchanged. Then each parameter may be calculated as follows:

$$\begin{aligned} p_1' &= \frac{M'}{K_f} \\ &= \frac{\bar{M} + \tilde{M}}{K_f} \\ &= p_1 + \frac{\tilde{M}}{K_f} \\ K_f &= \frac{\tilde{M}}{p_1' - p_1} \end{aligned} \tag{15}$$

Other unknown parameters can be similarly identified by

$$B = p_2 K_f \tag{16}$$

$$F_L = p_3 K_f \tag{17}$$

where p_2 and p_3 are the motor parameters from (13); K_f is known from (15). Thus, we obtain the parameters, B and F_L, by using (16) and (17), respectively.

Particle Swarm Optimization

PSO is a stochastic optimization algorithm first introduced (Kennedy and Eberhart, 1995) that imitates the process of animals' group communication behavior. The information sharing among the group's population, known as a swarm, help the group determine a "better" solution quickly whenever a member of the swarm, known as a particle, "discovers" a better route to the target. The PSO algorithm can be described by the updating rules of each particle (Clerc, 2006) as

$$\begin{aligned} v_i^{k+1} &= wv_i^k + c_1\gamma_1 \times (p_{\text{best}_\text{i}} - x_i^k) \\ &\quad + c_2\gamma_2 \times (g_{\text{best}} - x_i^k) \\ x_i^{k+1} &= x_i^k + v_i^{k+1} \end{aligned} \tag{18}$$

where c_1 and c_2 are positive constants, defined as the cognitive coefficients; w is the inertia weight factor; γ_1 and γ_2 are random numbers in $[0, 1]$; x_i and v_i are the position and velocity of i^{th} particle; p_{best_i} is the personal best position of particle; g_{best} is the global best position of the entire population; $k = 1, 2, \ldots$ indicates the number of iteration.

The flowchart of the PSO algorithm is shown in Figure 5 and the details of each step are as follows:

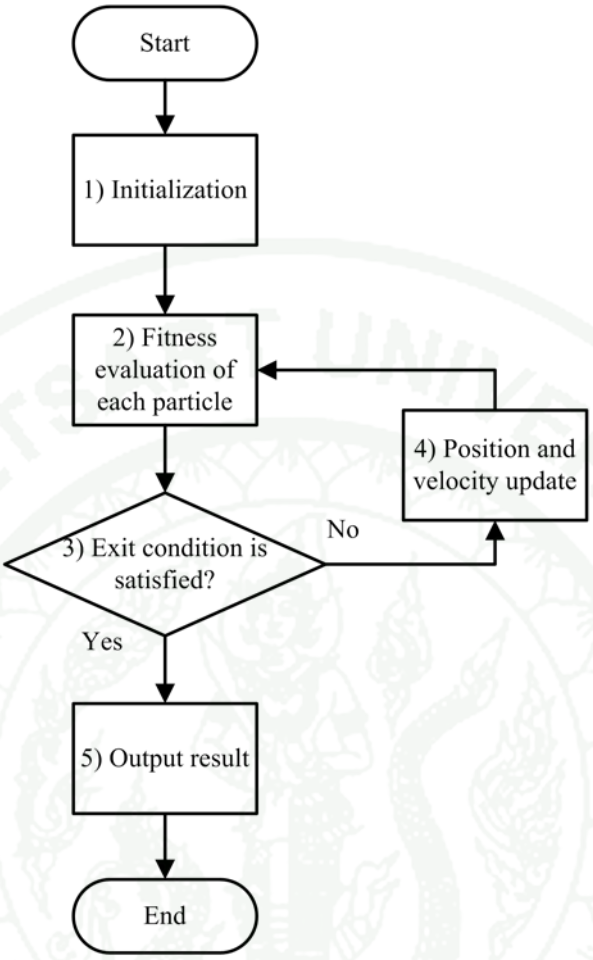


Figure 5 The PSO algorithm flowchart

1. Initialization

In this step, all parameters to be optimized are initialized. The position and velocity of each particle are randomly assigned in the search space; g_{best} and p_{best_i} are set at the maximum.

2. Fitness evaluation of each particle

For each step k, the fitness or performance of each particle is calculated. If the new calculated value is better than the current p_{best_i}, then new p_{best_i} is updated by the new calculated value; if the best of p_{best_i} is better than the current g_{best}, then g_{best} is replaced by the new best of p_{best_i}.

3. Exit condition

If the current g_{best}, satisfies the desired exit condition, i.e. either the algorithm reaches the maximum iteration or the g_{best} is lower than the minimum error criterion, then go to step 5. If not, go to step 4.

4. Position and Velocity update

The position and the velocity particle are updated according to (18). If the updated position and velocity particles are ended up outside the search space, their values are set at the boundaries (M. Clerc, 2006), i.e. at maximum or minimum value.

5. Output result

The best solution of the optimization process, g_{best}, is the output of this step.

PSO approach for LPM motor parameter identification

So far, we describe about the parameter identification, difficulty factors and identifier (PSO). Now thus, we can be able to design the parameter identification based on PSO algorithm. In this section, we present how to use the PSO algorithm for parameter identification of the LPM motor. The overall procedure of parameter identification based on PSO algorithm can be divided into two procedures: (i) Operating & Capturing data and (ii) Identification.

The procedure can be described into two steps for the first procedure and five steps for the second procedure. The detail of the overall procedure as shown in Figure 6 can be described here:

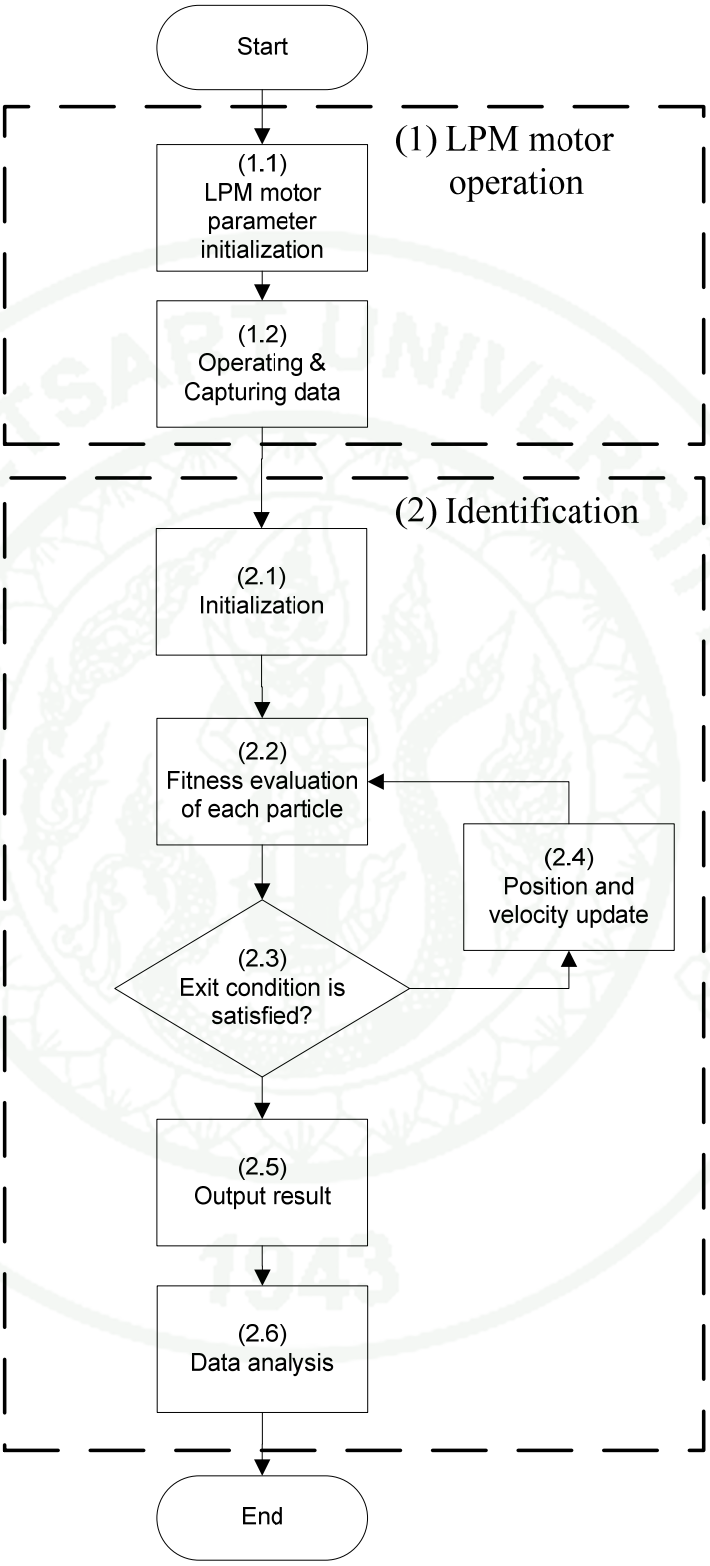


Figure 6 Parameter identification procedures

1. LPM motor operation

1.1 LPM motor parameter initialization

The LPM motor parameters have to initial before simulation or real experiment. For simulation, this step has to initial the LPM motor parameters such as motor force constant, K_f, total mass of moving parts, M, and viscous coefficient, B, by creating LPM motor model as shown in Figure 4. For experiment, this step has just connected the wiring of the LPM motor control system. Then, set the parameter configuration of motor driver in force control mode for operating and capturing data. Next, considering the difficulty factor of the system input selection, we should select the appropriate system input that should sufficiently excite the LPM motor model. In section difficulty factors, we described the system input selection factor. Thus, the appropriate system input should be chirp signal.

1.2 Operating & Capturing data

Normally, in this step, the LPM model simulation or operation is started, then the system outputs are captured, i.e. position and velocity outputs.

2. Identification

2.1 Initialization

After we have finish in Operating & Capturing data step, next we have to identify the LPM motor parameters from the capturing data (position output and velocity output). The initialized parameters are listed as: Population size, Exit condition, Inertia weight factor, Cognitive coefficient 1 and 2, and Random numbers. Normally, the values of these parameters are obtained from the experiment by trial and error. Clerc suggests that these PSO configuration parameters should be in Table 2.

Table 2 Selected parameter suggestion

Parameters	Values
Population size	20
Inertia weight factor, w	0.7
Cognitive coefficient1, c_1	1.43
Cognitive coefficient2, c_2	1.43

2.2 Fitness evaluation of each particle

Design the fitness function by considering the difficulty factor in Table 1. The fitness function should be in the sum of squares and the capturing data are position output and velocity output. Thus, the fitness function is given by:

$$C(\hat{p}) = \sum_{k=1}^{n} \left(w_{x_p}(x_p(k) - \hat{x}_p(k))^2 + w_{\dot{x}_p}(\dot{x}_p(k) - \hat{\dot{x}}_p(k))^2 \right) \qquad (19)$$

where x_p and $\hat{x}_p$ are the position responses of the real system and the estimated model; $\dot{x}_p$ and $\hat{\dot{x}}_p$ are the velocity responses of the real system and the estimated model; w_{x_p} and $w_{\dot{x}_p}$ are the weight factors; n is the number of the sampling data.

2.3 Exit condition is satisfied

Designing the exit condition, in this step, we can design the exit condition in three ways: (i) Using the maximum iteration as the exit condition, (ii) Using the error criterion as the exit condition and (iii) Using both maximum iteration and the error criterion as the exit condition.

2.4 Position and velocity update

Using Eq. (18), this step uses the updating rule equation to adjust the parameters to identify the nominal values of the LPM motor parameters.

2.5 Output result

When the estimated parameters are satisfied the exit condition either reaching the maximum iteration or reaching the minimum error criterion, the PSO identification will have finish the procedure.

2.6 Data analysis

After we obtain output data or estimated parameters that are the motor constants, the output data are not realized the nominal values. To identify the nominal values of the LPM motor parameters, we propose the technique to identify the nominal values.

This concept can be used for offline or online identification. For offline identification, the exit condition is reached when the estimated parameter vector approaches a target criterion ($\hat{p} \approx p$). For online identification, the exit condition could be when $\hat{p} \approx p$ or when the maximum iteration number. In this thesis, only offline identification is implemented to identify the mechanical parameters of the LPM motor.

Total sliding mode control

1. Baseline Model Control

The total sliding mode control, as shown in Figure 7, has a simple structure and can be implemented in a real time control system (Shyu *et al.*, 1999). Following (Wai, 2000) the control law of the total sliding mode control is given as

$$U = U_{\mathrm{BMC}} + U_{\mathrm{b}} \tag{20}$$

where

U : Control Variable

U_{BMC} : Baseline Model Control

U_{b} : Curbing Control.

The first term of the total sliding mode is called the Baseline Model Control (BMC) that is used to specify the desired performance of the nominal model. By rearranging (5), we obtain

$$\begin{aligned} \ddot{x}_p(t) &= -\frac{B}{M}\dot{x}_p(t) + \frac{K_f}{M}i_q^*(t) - \frac{1}{M}F_L \\ &\triangleq C_1\dot{x}_p(t) + C_2U(t) + C_3F_L \end{aligned} \tag{21}$$

where $C_1 = -\dfrac{B}{M}$, $C_2 = \dfrac{K_f}{M} > 0$, and $C_3 = -\dfrac{1}{M}$; $U(t) = i_q^*(t)$ is a control variable. By neglecting the parameter variations in M, K_f, B and the external force disturbances, (21) can be rewritten to obtain the nominal model as

$$\ddot{x}_p(t) = C_{1n}\dot{x}_p(t) + C_{2n}U(t) \tag{22}$$

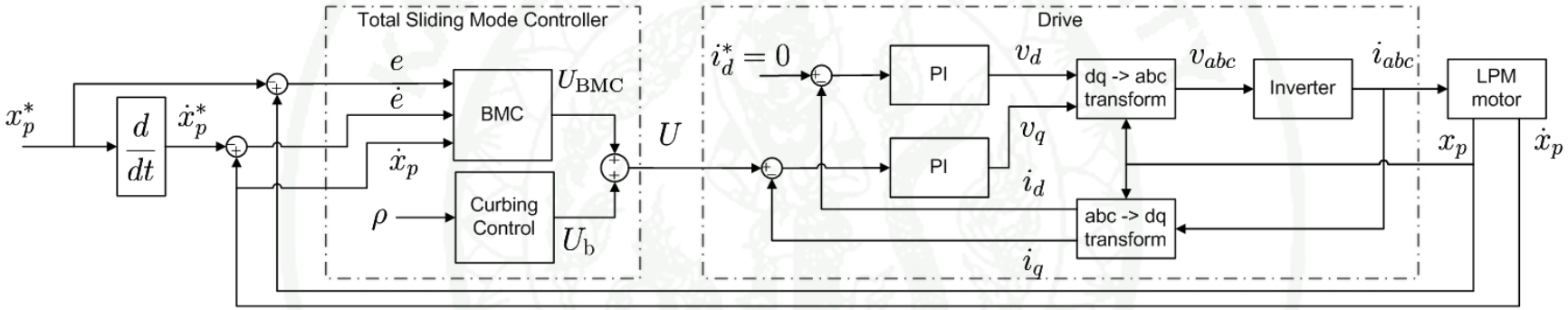


Figure 7 Total Sliding Mode control system

where $C_{1n} = -\frac{\bar{B}}{\bar{M}}$ and $C_{2n} = \frac{\bar{K}_f}{\bar{M}}$ are the nominal values of C_1 and C_2. The nominal parameters are indicated by the overbar symbol. For example, $\bar{K}_f$, $\bar{M}$ and $\bar{B}$ are the nominal values of K_f, M and B, respectively. When the uncertainties or the external force disturbances present, these parameters are deviated from their nominal values. The dynamic position movement (22) can be modified to

$$\begin{aligned} \ddot{x}_p(t) &= (C_{1n} + \triangle C_1)\dot{x}_p(t) \\ &\quad +(C_{2n} + \triangle C_2)U(t) + (C_{3n} + \triangle C_3)F_L \\ &= C_{1n}\dot{x}_p(t) + C_{2n}U(t) + W(t) \end{aligned} \tag{23}$$

where $C_{3n} = -\frac{1}{\bar{M}}$ is the nominal value of C_3. $\triangle C_1$, $\triangle C_2$, $\triangle C_3$, and F_L are the uncertainties. $W(t)$ is the lumped uncertainty, explicitly given by

$$W(t) = \triangle C_1\dot{x}_p(t) + \triangle C_2 U(t) + (C_{3n} + \triangle C_3)F_L \ . \tag{24}$$

Moreover, the lumped uncertainty should be bounded by

$$|W(t)| < \rho \ . \tag{25}$$

The selection of ρ affects the chattering phenomena and the task for adjusting ρ is non trivial. If ρ is too large, the chattering bound will be large. If ρ is too small, the system performance might be unstable. From (22), the BMC control law is given by

$$U_{\mathrm{BMC}} = U_{\mathrm{c}} + U_{\mathrm{s}} \tag{26}$$

where

$$U_{\mathrm{c}} = -C_{2n}^{-1}C_{1n}\dot{x}_p \tag{27}$$

$$U_{\mathrm{s}} = C_{2n}^{-1}[\ddot{x}_p^* - K_p e - K_v\dot{e}] \tag{28}$$

in which U_c is used to cancel the nonlinear terms in the model; U_s is used to specify the desired system performance; K_p and K_v are nonzero positive constants. Define the tracking error as $e \triangleq x_p - x_p^*$, where x_p^* is the position command. By rearranging (22), (26), and (27), we obtain the controlled nominal system

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \tag{29}$$

where K_p and K_v can be chosen to specify the desired second order system performance. However, only BMC cannot guarantee the desired system performance due to the potential lumped uncertainty $W(t)$. To ensure the desired system performance, we need the second term U_b of the control law as described in the next section.

2. Curbing control

The second term of the total sliding mode control is called Curbing Control that is used to eliminate uncertain perturbation effects from the parameter variation and external force disturbance. The curbing control law is given by

$$U_b = -\rho C_{2n}^{-1} sgn(S(t)) \tag{30}$$

where $\rho > 0$ is the control gain as defined in (25), $sgn(\cdot)$ is the signum function and $S(t)$ is the sliding surface as designed in (Shyu *et al.*, 1999) and (Wai, 2000). From (29), we obtain the state variable form

$$\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} \tag{31}$$

or

$$\dot{\mathbf{E}} = \mathbf{A}\mathbf{E} \tag{32}$$

where

$$\mathbf{E} = \begin{bmatrix} e & \dot{e} \end{bmatrix}^T \tag{33}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix} . \tag{34}$$

Now, considering the sliding surface:

$$S(t) = C(\mathbf{E}) - C(\mathbf{E_0}) - \int_0^t \frac{\partial C}{\partial \mathbf{E}} \mathbf{A}\mathbf{E} dt \tag{35}$$

where $C(\mathbf{E})$ is the vector to be designed such that $\frac{\partial C}{\partial E} = \begin{bmatrix} 0 & C_{2n}^{-1} \end{bmatrix}$, and $\mathbf{E_0}$ is the initial state of $\mathbf{E}$. The objective of Curbing Control is to maintain the controlled system on the sliding surface, $S(t) = 0$, at all time. From (35), it can be seen that this sliding surface has no reaching phase because $S(t) = 0$ when $t = 0$, compared with the traditional sliding mode controller (Slotine and Li, 1991); (Utkin *et al.*, 1999) and (Khalil, 2004).

Adaptive sliding mode control

The adaptive sliding mode control (Wai, 2000), in Figure 8, is a method to adjust ρ automatically in real time for the total sliding mode controller. The control law of the adaptive sliding mode control is given by

$$U = U_{\mathrm{BMC}} + \hat{U}_{\mathrm{b}} \tag{36}$$

where $\hat{U}_{\mathrm{b}}$ is the adaptive curbing control

$$\hat{U}_{\mathrm{b}}(t) = -\hat{\rho}(t) C_{2n}^{-1} sgn(S(t)) \tag{37}$$

and

$$\dot{\hat{\rho}}(t) = \tfrac{1}{\lambda} C_{2n}^{-1} |S(t)| \tag{38}$$

where $\hat{\rho}$ is the estimated value of ρ, and $\lambda > 0$ is the learning rate of the adaptive algorithm. For this control, the control gain parameter, ρ, is selected the upper bound in real time. Thus, the effect from pre selecting too large ρ by the total sliding mode controller is eliminated. If the nominal values of K_f, M and B are deviated from the nominal values, the upper bound of $|W(t)|$ or ρ in (25) will be automatically adjusted to be greater than $|W(t)|$. However, some chattering phenomena still appears using the adaptive sliding mode control.

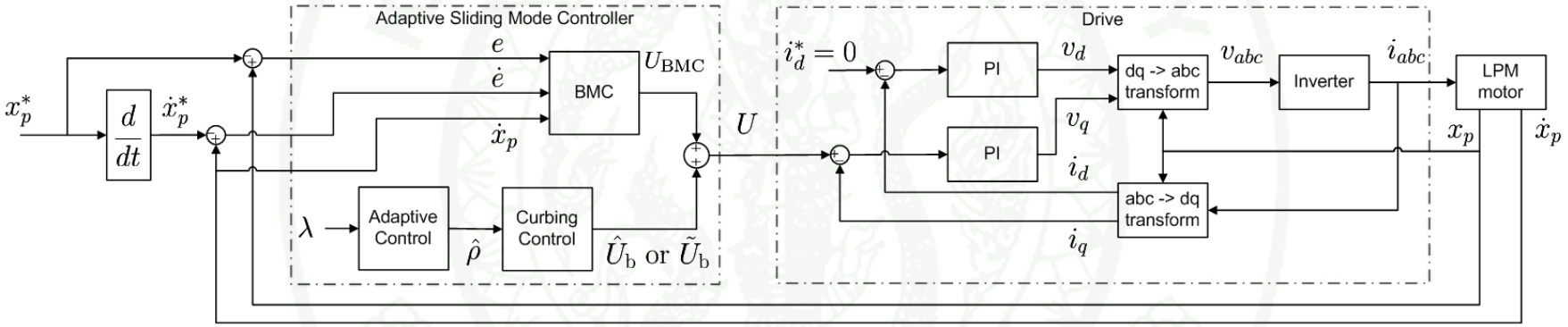


Figure 8 Adaptive Sliding Mode control system

Improved adaptive sliding mode control

To reduce the chattering phenomena, this thesis presents a method to improve the adaptive sliding mode control by using a continuous approximation method in (Slotine *et al.*, 1991), (Utkin *et al.*, 1999) and (Khalil, 2004). The proposed improved adaptive sliding mode control law is given by

$$U = U_{\mathrm{BMC}} + \tilde{U}_{\mathrm{b}} \tag{39}$$

where $\tilde{U}_{\mathrm{b}}$ is the proposed improved adaptive sliding mode control given by

$$\tilde{U}_{\mathrm{b}} = -\hat{\rho}(t) C_{2n}^{-1} sat\left(\frac{S(t)}{\epsilon}\right). \tag{40}$$

The signum function of $\hat{U}_{\mathrm{b}}$ in (37) is replaced by the high slope saturation function. The saturation function with high slope is defined as

$$sat\left(\frac{y}{\epsilon}\right) = \begin{cases} \dfrac{y}{\epsilon} & , \text{ if } |y/\epsilon| \leq 1 \\ sgn(\dfrac{y}{\epsilon}) & , \text{ if } |y/\epsilon| > 1 \end{cases} \tag{41}$$

where ϵ is a non zero positive constant. A good approximation requires ϵ in a small value (Slotine and W. Li, 1991) and (Khalil, 2004). For this control, the chattering phenomena can be reduced when the nominal values of K_f, M and B are deviated from the nominal values.

MATERIALS AND METHODS

Materials

Computer

A computer that supports MATLAB & Simulink Software with one available PCI card slot for DS1104 R&D controller board as shown in Figure 9 and Figure 10. The computer is used for both in the simulation and in the experiment. For simulation, we only use MATLAB & Simulink for simulating the LPM motor control system. For experiment, we use MATLAB & Simulink and DS1104 R&D controller board for real implementation. The computer is used for all computation and data analysis in the implementation.

Figure 9 Computer PC with MATLAB & Simulink software

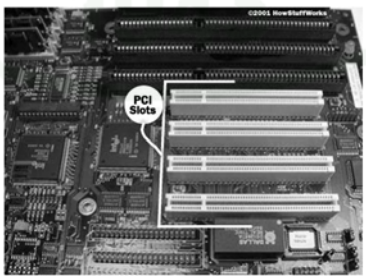


Figure 10 PCI Slot

Motion controller board

The motion controller board is to necessarily control the real LPM motor control system through a computer. The minimum system requirement of the motion controller board must be the features: (i) Digital to Analog Converter (DAC) 1 channel and (ii) Incremental Encoder 1 channel.

DS1104 R&D Controller Board has completed the above features. One DAC is used for the control signal and One Incremental Encoder is used for reading the feedback signal from the Position Linear Encoder that is installed on the LPM motor. DS1104 has the Real Time Interface (RTI) feature that can monitor and capture the signal environment in real time for data analysis. Therefore, DS1104 is suitable for developing in rapid prototype such as parameter identification method or testing control algorithm. For DS1104, the RTI feature can be enabled by using the Control Desk application that is provided by the manufacturer.

Figure 11 DS1104 dSPACE motion controller

Motor driver

Due to the power limitation of the motion controller board, DS1104 cannot directly drive the LPM motor. Therefore, the motor driver or current amplifier is necessary for the LPM motor control system. Elmo servo driver, Harmonica HAR A05/60I, as shown in Figure 12, is one multifunctional motor driver. Although it can be customized to position control mode, velocity control mode and force/current control mode, HAR A05/65I is only used in force/current control mode for power amplification.

Figure 12 Motor driver

Linear Permanent Magnet Motor with position linear encoder

LPM motor, LMS30 C1S1035 as shown in Figure 13, is an electric linear motor that is used for implementation. It is already mounted the position linear encoder as shown in Figure 14. LMS30 C1S1035 is compatible with Elmo servo driver, HAR A05/60I.

Figure 13 Linear Permanent Magnet motor

Figure 14 Linear position encoder

Measurement instruments

Force meter, Weight Scale, Electrical meter and Oscilloscope are used for measuring force, weight and electrical signal in the implementation.

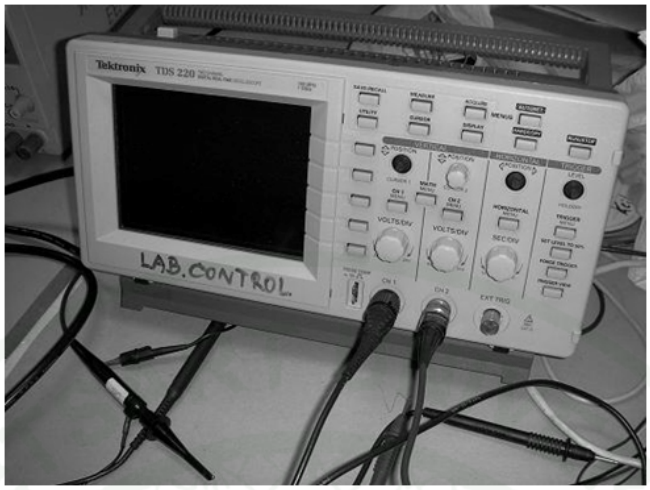


Figure 15 Oscilloscope

External electrical power supply

Electrical Power supply is a device that is used for supplying the electrical energy to the system. For this implementation, we need two external power supplies. One for motor driver using 12 24 volts and another one for driving the LPM motor using 30 60 volts.

Figure 16 External power supply 12-24 volts

Figure 17 External power supply 30-60 volts

Methods

Simulation of parameter identification

a) Simulation

The simulations are implemented using MATLAB & Simulink. All simulations are tested by adjusting two weight factors between [0 , 1]. Each simulation run is performed for three cases:

$$\begin{aligned} \text{Case } 1 &: M = \bar{M} \\ \text{Case } 2 &: M = \bar{M} + 1.72 \text{ kg} \\ \text{Case } 3 &: M = \bar{M} + 2.75 \text{ kg} \end{aligned} \tag{42}$$

where $\bar{M}$ is the nominal value of the moving mass without payload. The PSO identification method is simulated with the set of motor parameters shown in Table 3 and the PSO configuration parameters are listed in Table 4.

Table 3 LPM motor parameters

LPM motor parameters	Nominal values
Motor force constant, K_f	10.83 N/A
Nominal moving mass, $\bar{M}$	1.4 kg
Viscous friction coefficient, B	5 Ns/m
External frictional force, F_L	0.05 N

Table 4 PSO Configuration parameters

PSO identification parameters	Values
Population size	20
Maximum iteration	150
Inertia weight factor, w	0.7
Cognitive coefficient1, c_1	1.43
Cognitive coefficient2, c_2	1.43
Random numbers, (γ_1, γ_2)	Randomly chosen between [0,1] for each calculation

The simulations are conducted as the diagram in Figure 6.

b) Simulation by using step input

1. LPM motor operation

1.1 LPM motor parameter initialization

Create the motor model for simulation in Simulink, then set LPM motor parameters as following tested Case 1 in (42) and in Table 4. Then, set position

and velocity outputs as capturing data shown in Figure 18 and set input as step signal as shown in Figure 19.

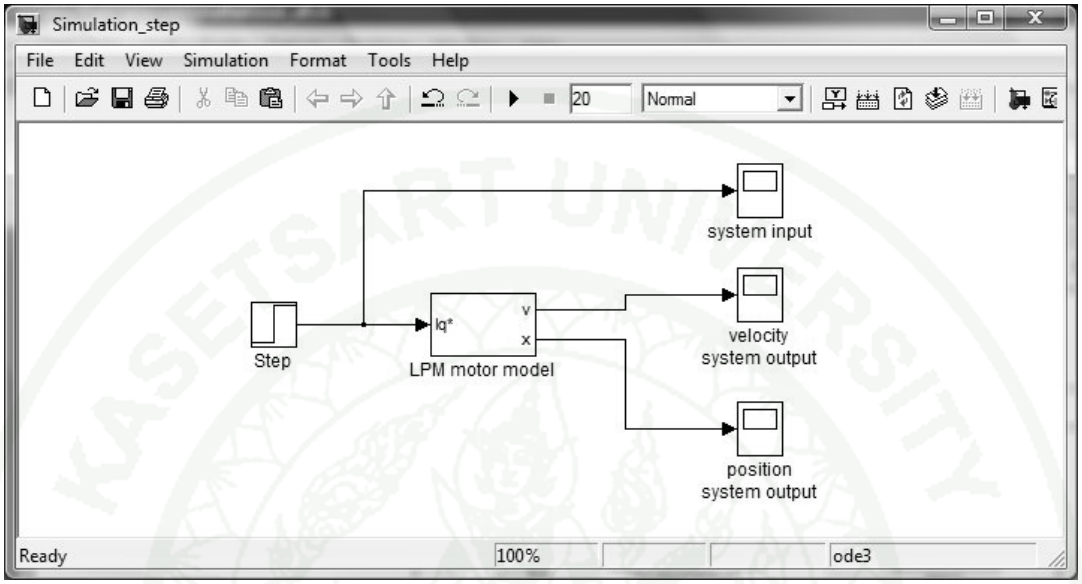


Figure 18 LPM motor model by using step input

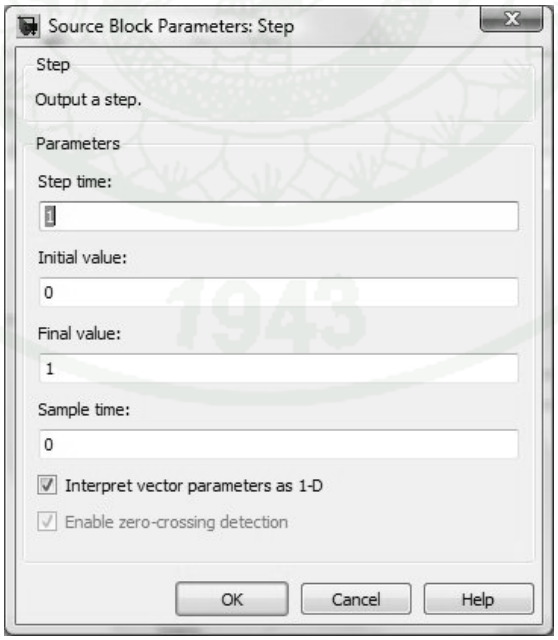


Figure 19 Step input configuration

1.2 Operating & Capturing data

Start the LPM motor control system for simulation to obtain the system outputs, i.e. position output and velocity output. After obtain the system outputs for tested case 1, try to simulate the condition in Case 2 to obtain another the system outputs.

2. Identification

2.1 Initialization

Set the PSO configuration parameters following in Table 4.

2.2 Fitness & Evaluation
2.3 Exit condition is satisfied
2.4 Position and velocity update
2.5 Output result

These four steps (2.2 – 2.5) are implemented in MATLAB M-file for PSO identification.

2.6 Data analysis

Before reach this step, all tested cases have to be simulated to obtain the motor constant parameters. After that, we have to identify the nominal values of the LPM motor parameters by using a technique in (15), (16) and (17).

c) Simulation by using chirp input

The simulation by using chirp input is similar to simulation procedure when using step input. The LPM motor model for simulation in Simulink is changed to chirp input as shown in Figure 20 and Figure 21. The chirp signal that is a

sinusoidal function with varying frequency ω over a time period $0 \leq t \leq t_f$ whose equation is given by

$$u(t) = A\cos(\frac{\omega_1 t + (\omega_2 - \omega_1)t^2}{2t_f}) \tag{43}$$

where A is the amplitude; ω_1 and ω_2 are the lower and upper bound frequencies; t_f is the final time. These parameters are given in Table 5 and Figure 21. Otherwise, the procedures are the same simulation by using step input.

Table 5 Chirp signal input parameters

Chirp signal input parameters	Values
Lower bound frequency, ω_1	0.1 Hz
Upper bound frequency, ω_2	100 Hz
Final time, t_f	20 sec

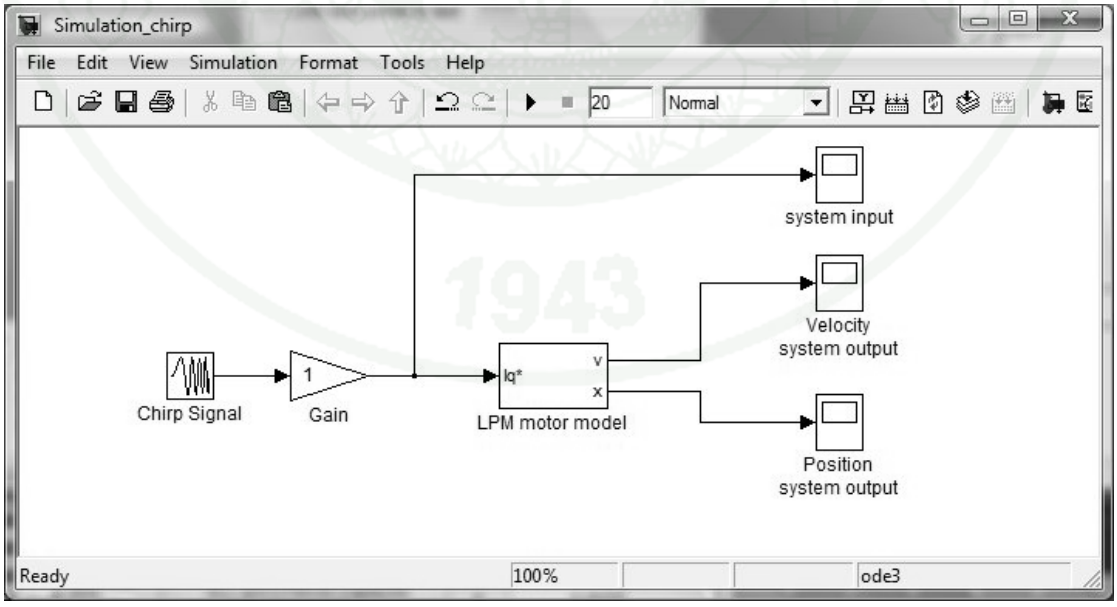


Figure 20 LPM motor model by using chirp signal input

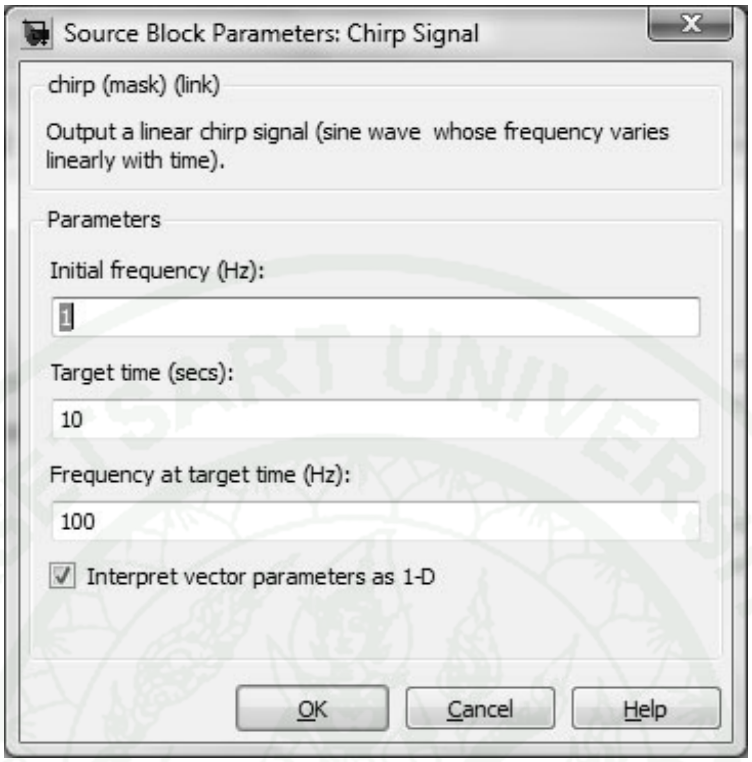


Figure 21 Chirp signal parameters in Simulink

d) Simulation by using PRBS input

The simulations also implement using PRBS input to determine the accuracy among the signal inputs such as step, chirp and PRBS inputs. PRBS is a periodic, deterministic signal with white noise like properties. The simulation using PRBS input is also similar to use step and chirp inputs. The simulation model is shown in Figure 22. The PRBS parameters are shown in Table 6 and Figure 23.

Table 6 PRBS input parameters

PRBS parameters	Values
Amplitude	±1
Frequency	5 Hz

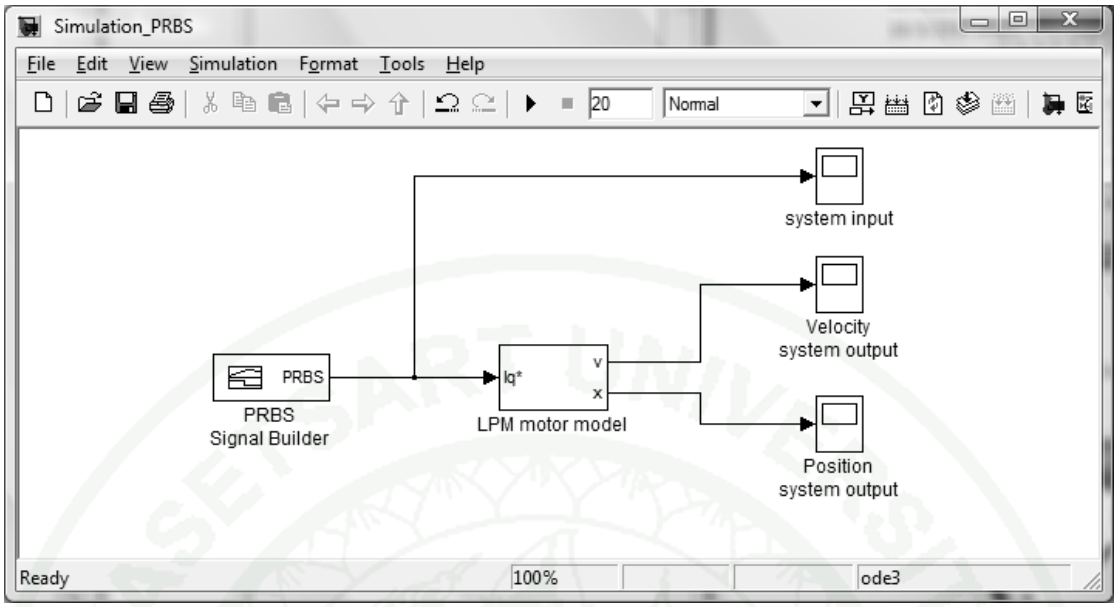


Figure 22 LPM motor model by using PRBS input

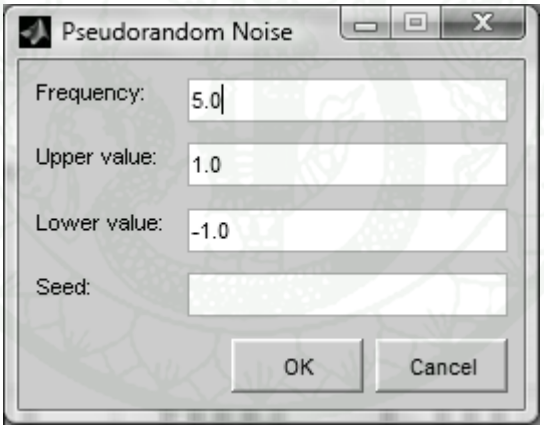


Figure 23 PRBS parameters in Simulink

Experiment of parameter identification

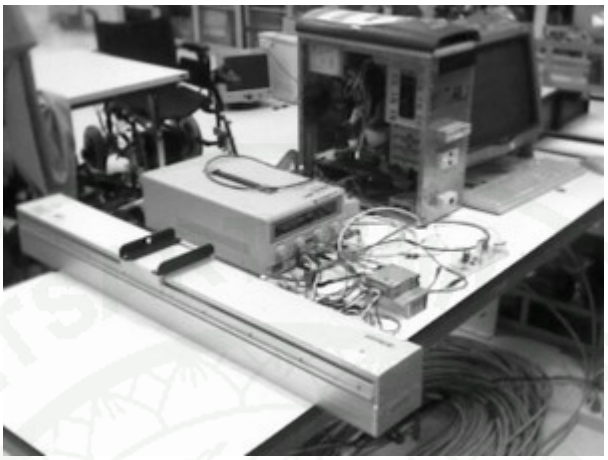


Figure 24 Experiment setup

The PSO parameter identification method is implemented on a PC for the experiment setup, shown in Figure 24, where the LPM motor is controlled in a force/current control mode. The experiment procedures are conducted as the diagram shown in Figure 26. First, setup the hardware for experiment by connecting wire following block diagram of the LPM motor control system diagram.

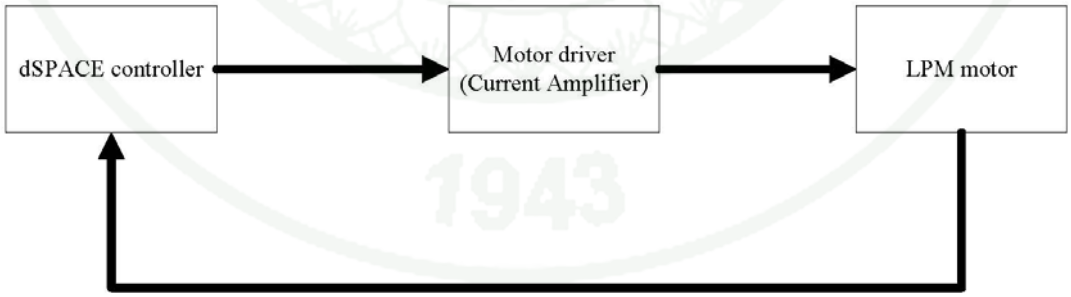


Figure 25 Block diagram of the LPM motor control system

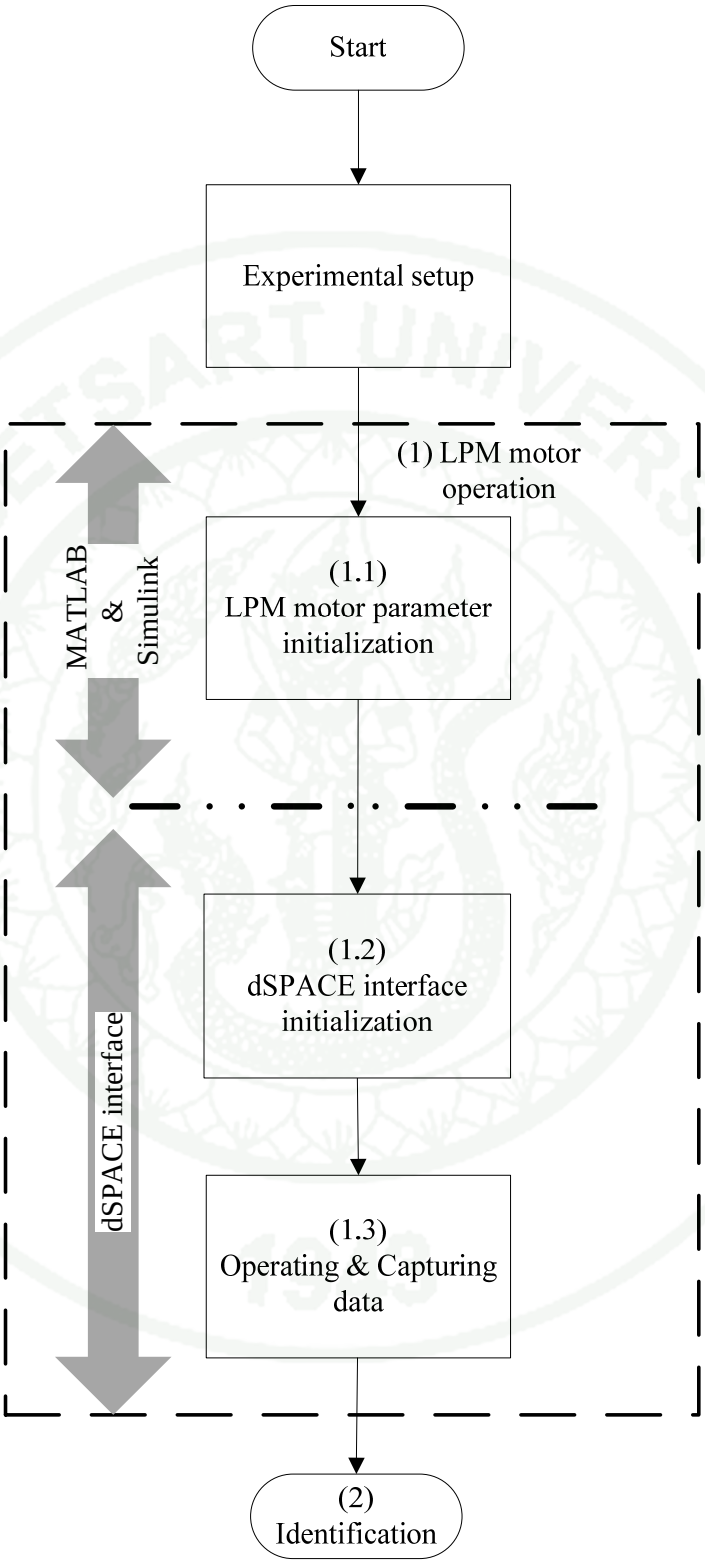


Figure 26 Parameter identification procedures for experiment

1. LPM motor operation

1.1 LPM motor parameter initialization

Create the LPM motor control model for real experiment by using dSPACE block library in Simulink as shown in Figure 27. MATLAB & Simulink is used to directly build source code to dSPACE controller though PCI interface in a computer. dSPACE controller can control the LPM motor through motor driver (current amplifier) using voltage range between 10 ~ 10 volts and getting feedback signal from optical linear encoder in digital signal 10 volt for logic low and 10 volt for logic high. Thus, the LPM motor in Simulink should use the dSPACE block library of analog output and encoder input as shown in Figure 28. After that, build source code for dSPACE controller by using MATLAB & Simulink.

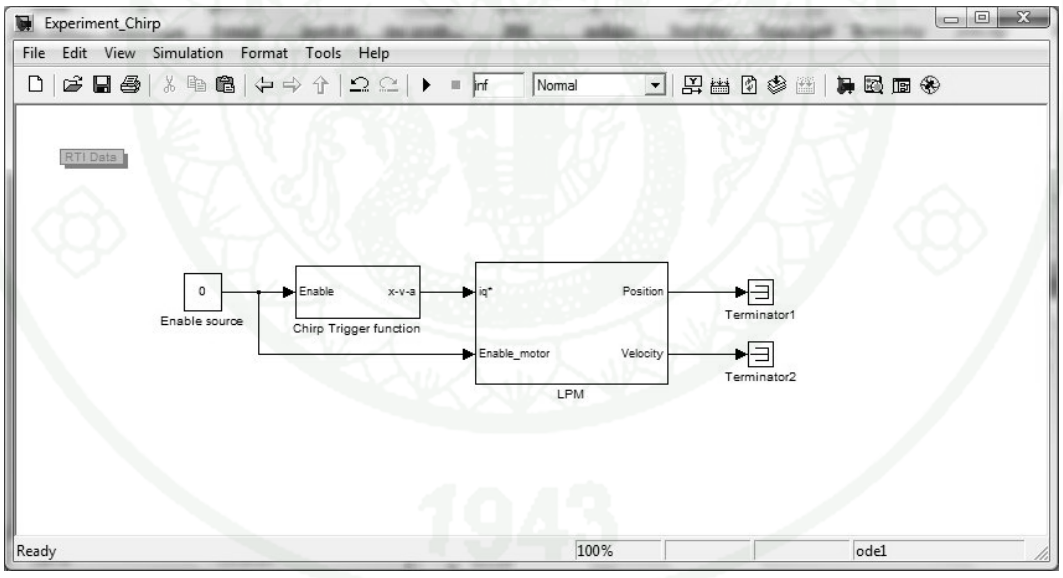


Figure 27 LPM motor control system for experiment

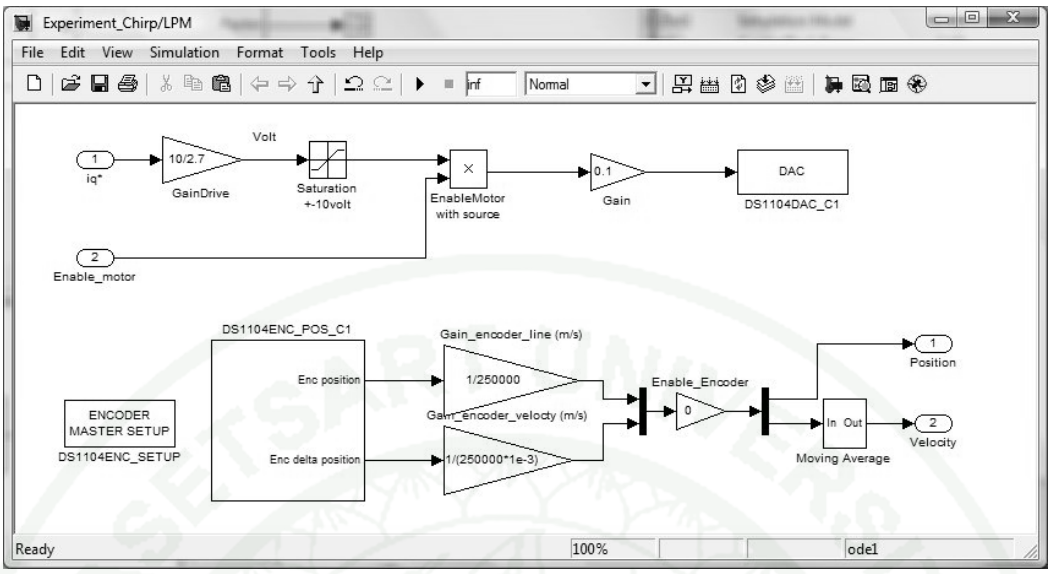


Figure 28 Detail in LPM motor block

1.2 dSPACE interface initialization

After build source code to dSPACE controller, dSPACE controller needs its software from manufacturer to operate and to monitor dSPACE signal data in real time.

1.3 Operating & Capturing data

Start operation and wait until the dSPACE controller finishes capturing the system outputs.

2. Identification

This step is the same simulation procedures.

Simulation of sliding mode control

The simulations and experiments, as shown in Figure 24, are implemented for three control systems: (i) the total sliding mode control, (ii) the adaptive sliding mode control, and (iii) the proposed improved adaptive sliding mode control. These control systems have been applied to control the three phase LPM motor. The simulation are implemented on a Pentium 4, 3.0 GHz PC with MATLAB & Simulink and the dSPACE (DS1104) controller card. All simulations and experiments are tested for two cases: (i) without payload and (ii) with an additional 3.5 kg payload, i.e.

$$\begin{aligned} &\text{Case 1}: \quad M = \bar{M} \\ &\text{Case 2}: \quad M = \bar{M} + 3.5 \text{ kg} . \end{aligned} \tag{44}$$

The sinusoidal function is used as the position reference input x_p^*. The performance criterion for the three control systems is the Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |x_{pi} - x_{pi}^*| = \frac{1}{n}\sum_{i=1}^{n} |e_i| \tag{45}$$

where n is the number of data points, x_{pi} and x_{pi}^* are the actual position and the position reference input, respectively. The nominal values of the LPM motor and control parameters are listed in Table 7 and Table 8.

Table 7 LPM motor parameters

LPM motor parameters	Values
$\bar{K}_f$	10.86 N/A
$\bar{M}$	1.4 kg
$\bar{B}$	2 Ns/m

Table 8 Control parameters

Control parameters	Values
K_p	2500
K_v	100
ρ	3
λ	0.01
ϵ	0.002

RESULTS AND DISCUSSION

Results

Simulation results of parameter identification

Table 9 Simulation results using step signal input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	$(0.0, 1.0)$	13.620	1.746	5.894	0.178
2	$(0.1, 0.9)$	13.397	1.737	5.872	0.022
3	$(0.2, 0.8)$	21.274	2.761	8.520	0.014
4	$(0.3, 0.7)$	12.750	1.247	4.602	3.154
5	$(0.4, 0.6)$	26.298	3.361	10.073	0.417
6	$(0.5, 0.5)$	12.162	1.253	4.618	2.519
7	$(0.6, 0.4)$	12.163	1.346	4.859	1.802
8	$(0.7, 0.3)$	13.643	1.748	5.900	0.183
9	$(0.8, 0.2)$	11.000	1.053	4.101	2.895
10	$(0.9, 0.1)$	21.384	2.771	8.546	0.049
11	$(1.0, 0.0)$	12.775	1.366	4.910	2.262

Table 10 Simulation results using chirp signal input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	$(0.0, 1.0)$	10.830	1.400	5.000	0.050
2	$(0.1, 0.9)$	10.830	1.400	5.000	0.050
3	$(0.2, 0.8)$	10.234	1.332	3.486	0.104
4	$(0.3, 0.7)$	10.606	1.383	4.717	0.000
5	$(0.4, 0.6)$	10.897	1.390	4.921	0.339
6	$(0.5, 0.5)$	11.014	1.423	5.261	0.045
7	$(0.6, 0.4)$	10.786	1.403	4.981	0.000
8	$(0.7, 0.3)$	10.119	1.340	3.931	0.047
9	$(0.8, 0.2)$	10.682	1.380	4.865	0.000
10	$(0.9, 0.1)$	11.007	1.427	5.206	0.000
11	$(1.0, 0.0)$	10.957	1.413	5.130	0.000

Table 11 Simulation results using PRBS input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	$(0.0, 1.0)$	10.830	1.400	5.000	0.050
2	$(0.1, 0.9)$	10.828	1.399	4.993	0.066
3	$(0.2, 0.8)$	10.819	1.398	5.013	0.050
4	$(0.3, 0.7)$	10.828	1.399	4.985	0.089
5	$(0.4, 0.6)$	10.724	1.363	4.945	0.126
6	$(0.5, 0.5)$	10.827	1.399	4.974	0.161
7	$(0.6, 0.4)$	10.813	1.394	4.983	0.076
8	$(0.7, 0.3)$	10.821	1.398	5.002	0.050
9	$(0.8, 0.2)$	10.816	1.399	5.033	0.050
10	$(0.9, 0.1)$	10.814	1.394	4.929	0.221
11	$(1.0, 0.0)$	10.827	1.399	4.996	0.058

Table 12 Percent absolute error in case using step signal input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (%)	M (%)	B (%)	F_L (%)
1	$(0.0, 1.0)$	25.76	24.69	17.88	256.46
2	$(0.1, 0.9)$	23.70	24.08	17.44	56.70
3	$(0.2, 0.8)$	96.44	97.21	70.41	71.63
4	$(0.3, 0.7)$	17.73	10.96	7.95	6,207.00
5	$(0.4, 0.6)$	142.83	140.09	101.46	733.08
6	$(0.5, 0.5)$	12.30	10.53	7.64	4,938.20
7	$(0.6, 0.4)$	12.31	3.88	2.82	3,503.80
8	$(0.7, 0.3)$	25.97	24.86	18.01	266.80
9	$(0.8, 0.2)$	1.57	24.79	17.97	5,690.40
10	$(0.9, 0.1)$	97.45	97.91	70.91	1.67
11	$(1.0, 0.0)$	17.96	2.46	1.79	4,423.40

Table 13 Percent absolute error in case using chirp signal input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (%)	M (%)	B (%)	F_L (%)
1	$(0.0, 1.0)$	0.00	0.00	0.00	90.00
2	$(0.1, 0.9)$	0.00	0.00	0.00	90.00
3	$(0.2, 0.8)$	5.50	4.86	30.28	108.00
4	$(0.3, 0.7)$	2.07	1.21	5.66	99.86
5	$(0.4, 0.6)$	0.62	0.71	1.58	578.00
6	$(0.5, 0.5)$	1.70	1.64	5.22	10.00
7	$(0.6, 0.4)$	0.41	0.21	0.38	99.86
8	$(0.7, 0.3)$	6.57	4.29	21.38	6.00
9	$(0.8, 0.2)$	1.37	1.43	2.70	99.86
10	$(0.9, 0.1)$	1.63	1.93	4.12	99.85
11	$(1.0, 0.0)$	1.17	0.93	2.60	99.85

Table 14 Percent absolute error in case using PRBS input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (%)	M (%)	B (%)	F_L (%)
1	$(0.0, 1.0)$	0.00	0.00	0.00	0.00
2	$(0.1, 0.9)$	0.02	0.07	0.14	32.00
3	$(0.2, 0.8)$	0.10	0.14	0.26	0.00
4	$(0.3, 0.7)$	0.01	0.04	0.31	77.98
5	$(0.4, 0.6)$	0.98	2.68	1.09	152.10
6	$(0.5, 0.5)$	0.03	0.07	0.52	222.00
7	$(0.6, 0.4)$	0.16	0.46	0.34	51.84
8	$(0.7, 0.3)$	0.08	0.12	0.03	0.18
9	$(0.8, 0.2)$	0.13	0.11	0.66	0.13
10	$(0.9, 0.1)$	0.15	0.43	1.42	342.00
11	$(1.0, 0.0)$	0.03	0.07	0.08	16.00

Table 15 Average of percent absolute error for comparison among three test signal inputs

Input	K_f (%)	M (%)	B (%)	F_L (%)
Step	43.09	41.95	30.39	2,377.19
Chirp	1.91	1.56	6.72	125.57
PRBS	0.15	0.38	0.44	81.29

Experimental results of parameter identification

Table 16 Experimental results using chirp signal input

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)	MAE (Position)	MAE (Velocity)
1	$(0.0, 1.0)$	10.812	1.538	6.271	0.057	0.780	15.170
2	$(0.1, 0.9)$	10.892	1.551	6.328	0.072	0.894	15.367
3	$(0.2, 0.8)$	10.845	1.545	6.362	0.061	1.040	16.098
4	$(0.3, 0.7)$	10.806	1.540	5.760	0.143	0.675	12.022
5	$(0.4, 0.6)$	10.799	1.538	6.520	0.026	1.481	18.490
6	$(0.5, 0.5)$	10.820	1.541	6.083	0.116	0.571	13.308
7	$(0.6, 0.4)$	10.958	1.560	5.257	0.247	2.227	21.201
8	$(0.7, 0.3)$	10.702	1.523	6.299	0.057	1.052	16.374
9	$(0.8, 0.2)$	11.261	1.487	2.967	1.432	8.486	77.957
10	$(0.9, 0.1)$	10.571	1.499	6.213	0.058	0.915	16.262
11	$(1.0, 0.0)$	10.534	1.472	6.503	0.000	1.170	20.231

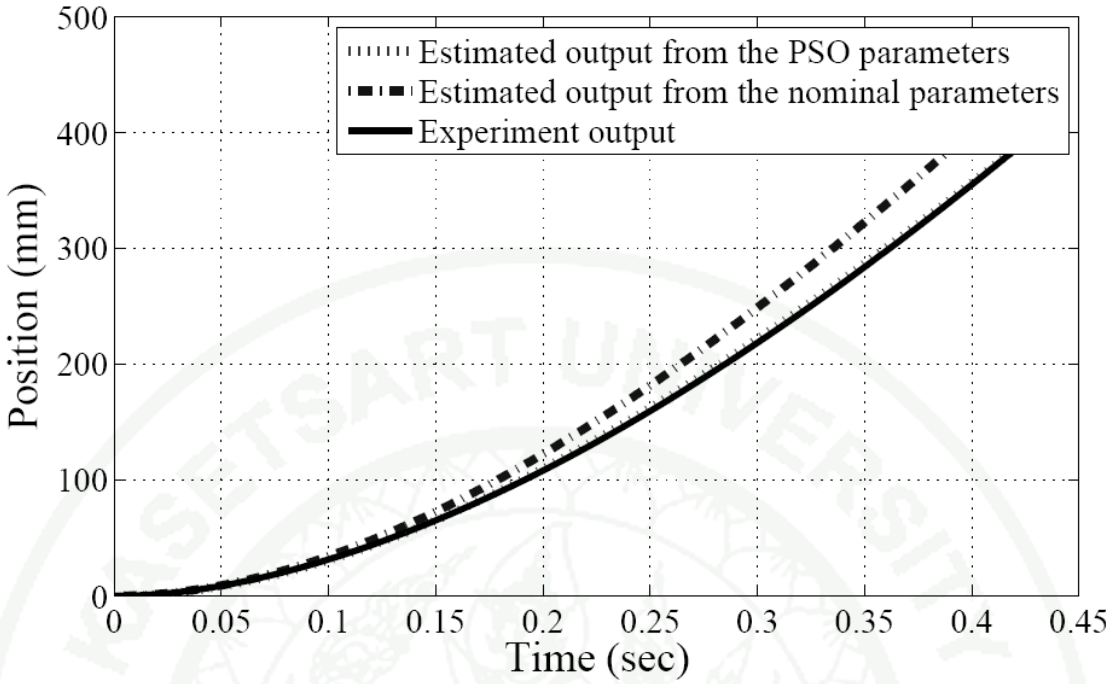


Figure 29 Position of the system outputs

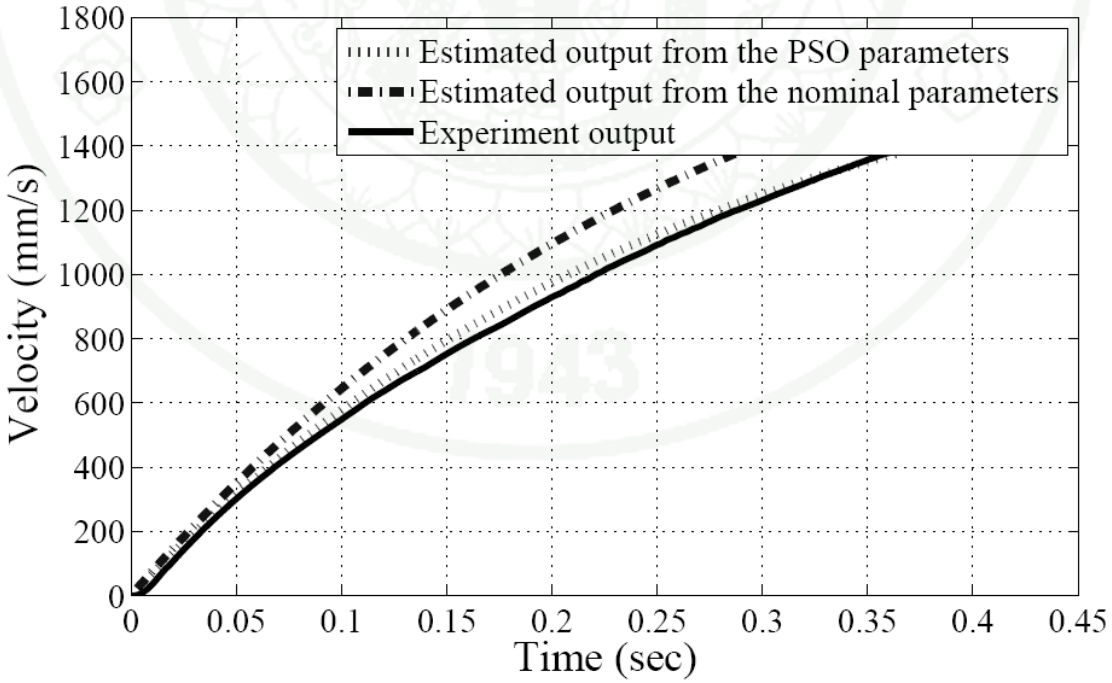


Figure 30 Velocity of the system outputs

Simulation results of sliding mode controllers

Table 17 Mean Absolute Error Comparison of Simulation Results

Controllers	Case 1 (mm)	Case 2 (mm)
Total Sliding Mode	0.08	1.07
Adaptive Sliding Mode	0.21	0.64
Improved Adaptive Sliding Mode	0.21	0.64

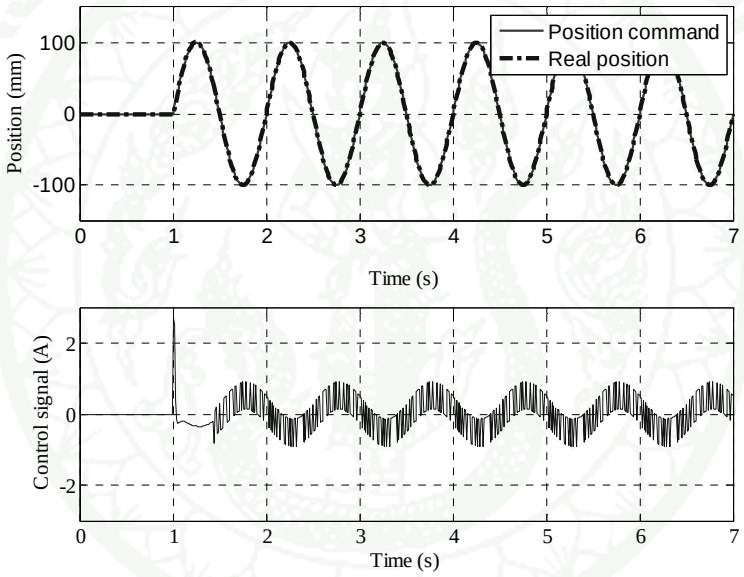


Figure 31 Simulation results of the total sliding mode control for Case 1 ($M = \bar{M}$)

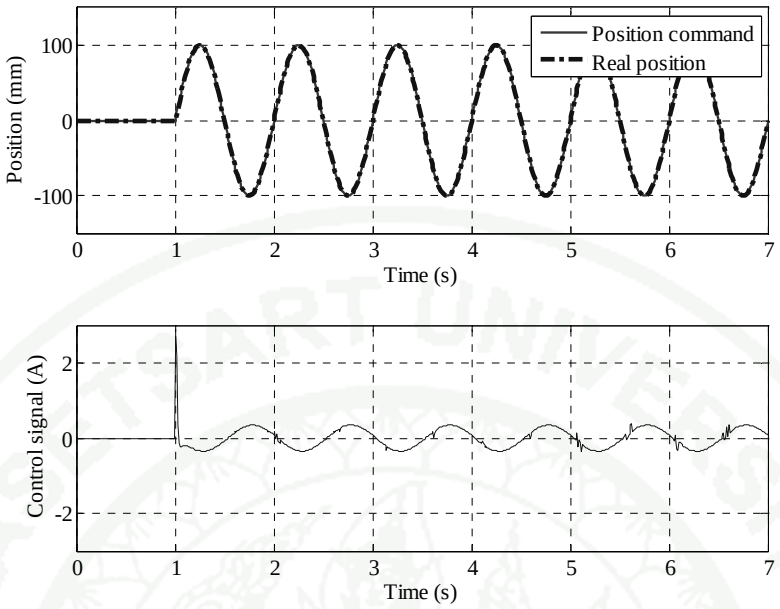


Figure 32 Simulation results of the adaptive sliding mode control for Case 1 ($M = \bar{M}$)

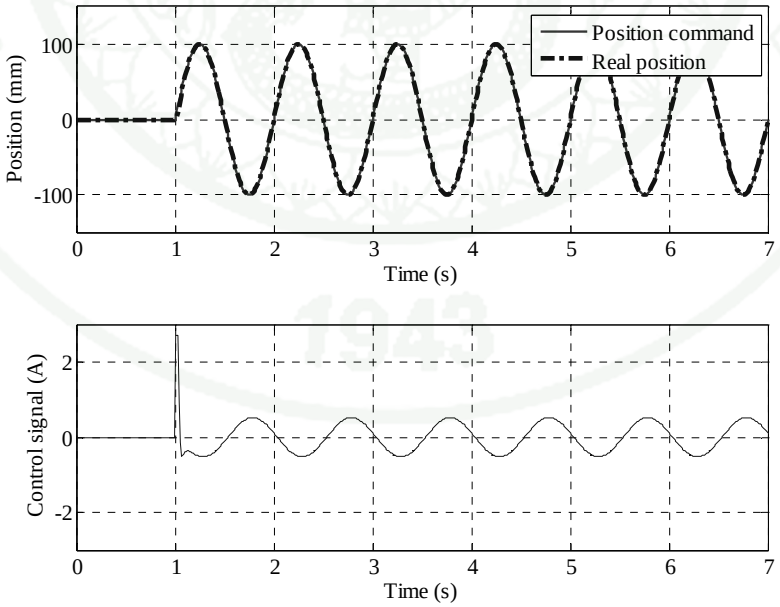


Figure 33 Simulation results of the improved adaptive sliding mode control for Case 1 ($M = \bar{M}$)

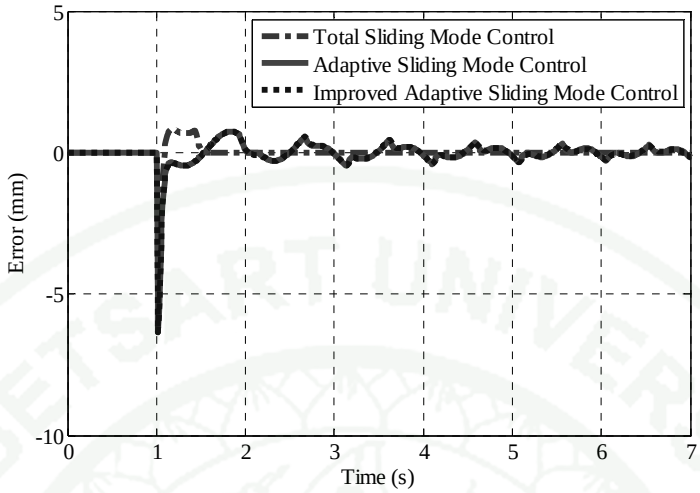


Figure 34 Comparison of simulation results of the Mean Absolute Errors for Case 1 ($M = \bar{M}$)

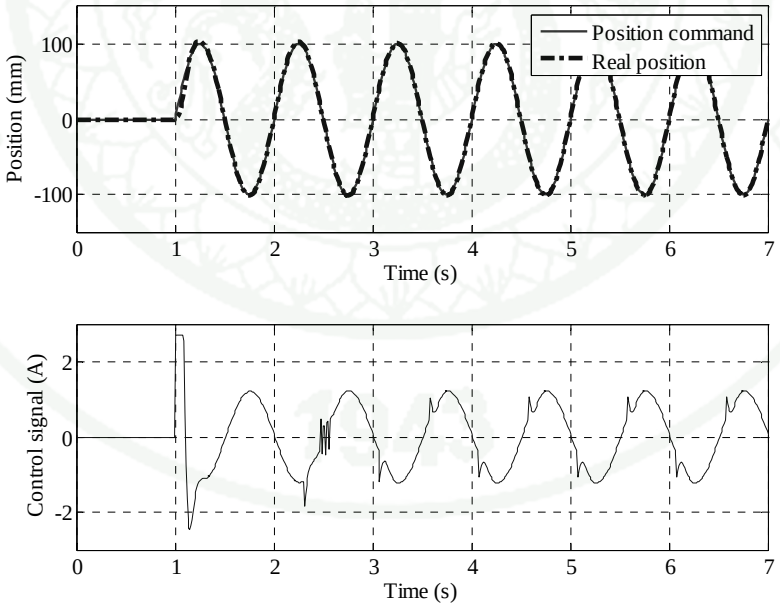


Figure 35 Simulation results of the total sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

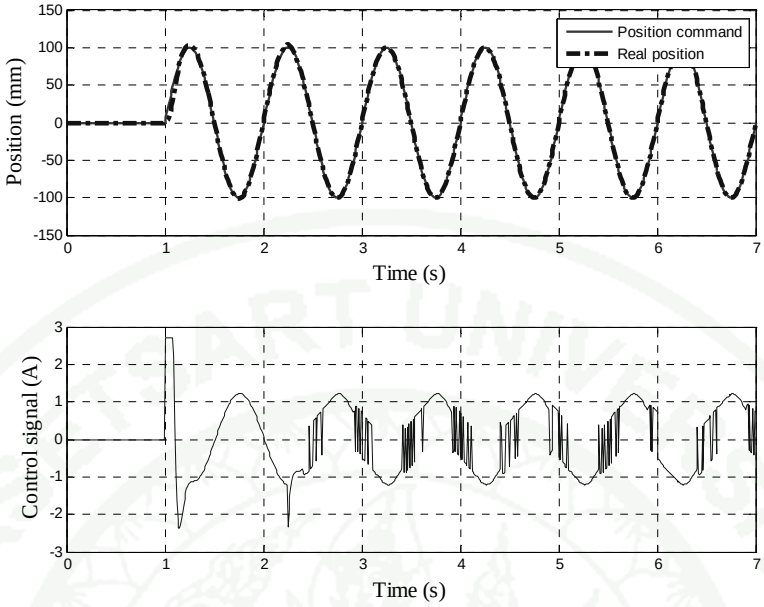


Figure 36 Simulation results of the adaptive sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

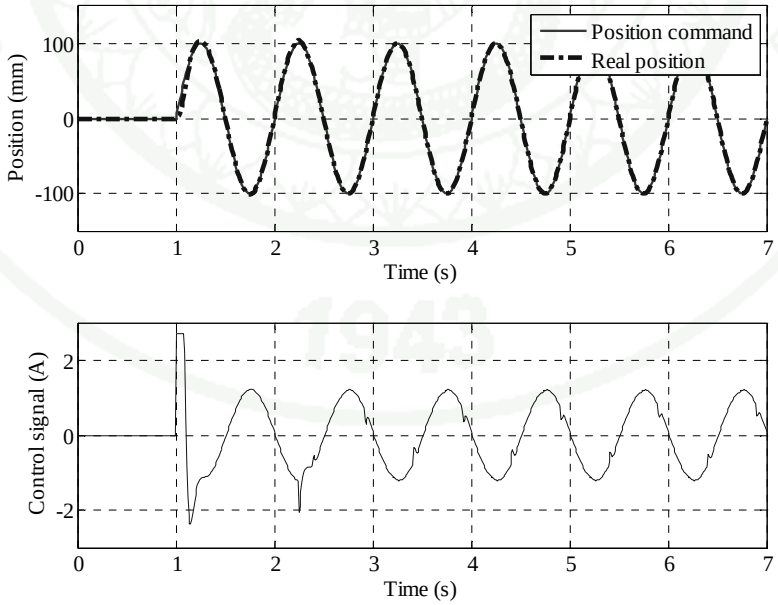


Figure 37 Simulation results of the improved adaptive sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

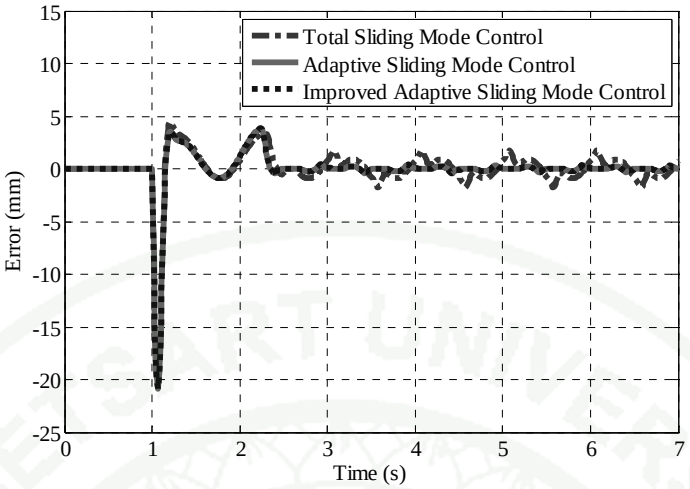


Figure 38 Comparison of simulation results of the Mean Absolute Errors for Case 2 ($M = \bar{M} + 3.5$ kg)

Experimental results of sliding mode controllers

Table 18 Mean Absolute Error Comparison of Experimental Results

Controllers	Case 1 (mm)	Case 2 (mm)
Total Sliding Mode	0.22	1.72
Adaptive Sliding Mode	0.25	1.00
Improved Adaptive Sliding Mode	0.24	0.96

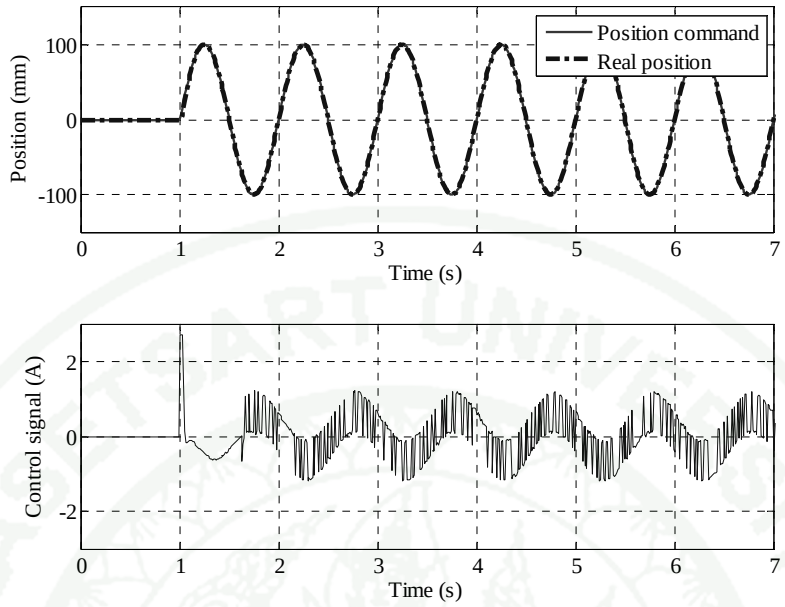


Figure 39 Experimental results of the total sliding mode control for Case 1 ($M = \bar{M}$)

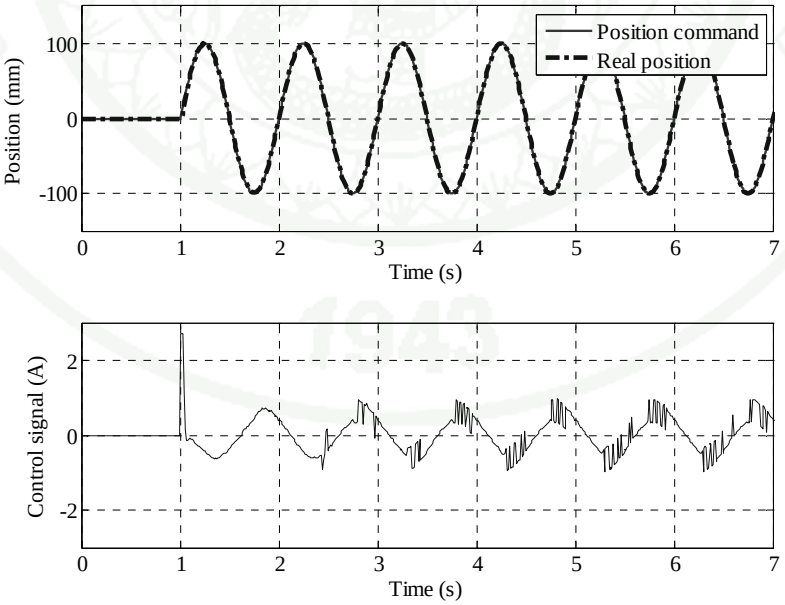


Figure 40 Experimental results of the adaptive sliding mode control for Case 1 ($M = \bar{M}$)

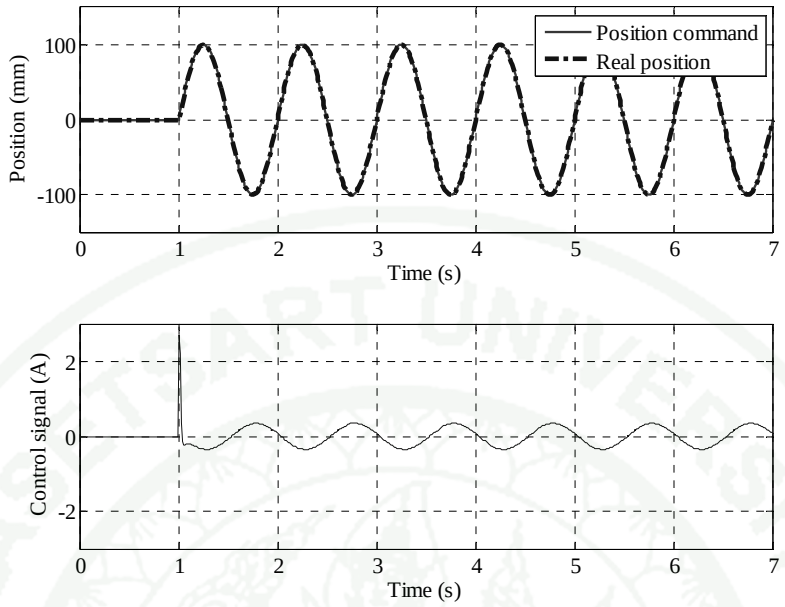


Figure 41 Experimental results of the improved adaptive sliding mode control for Case 1 ($M = \bar{M}$)

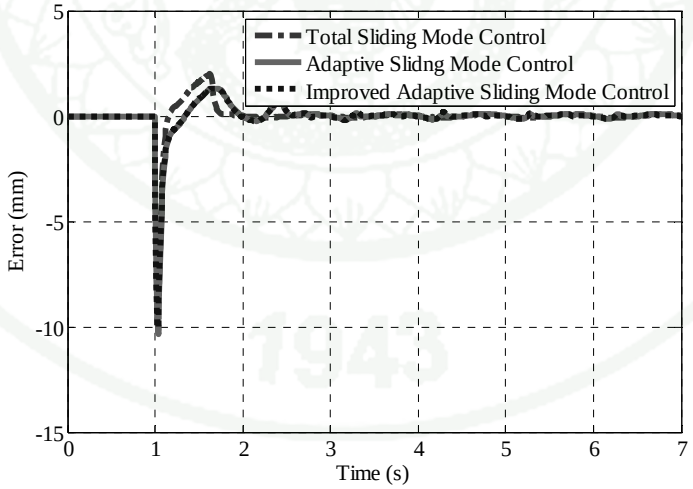


Figure 42 Comparison of experimental results of the Mean Absolute Errors for Case 1 ($M = \bar{M}$)

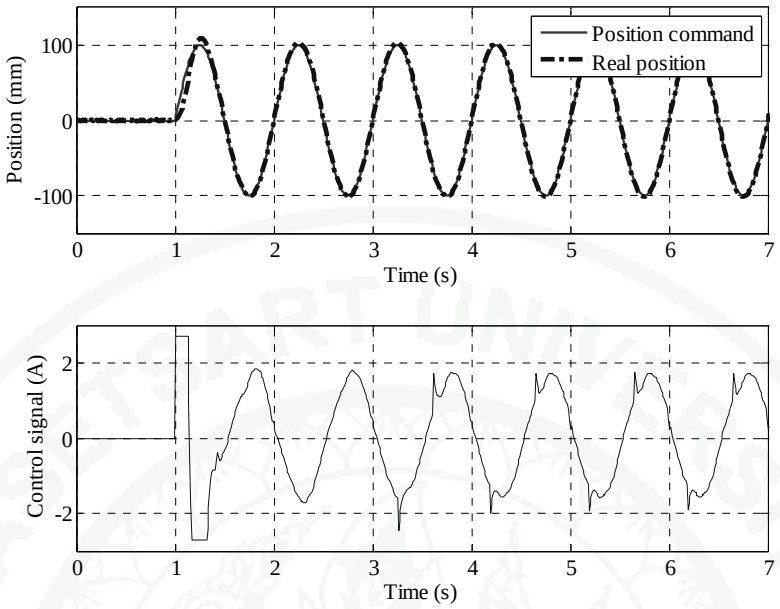


Figure 43 Experimental results of the total sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

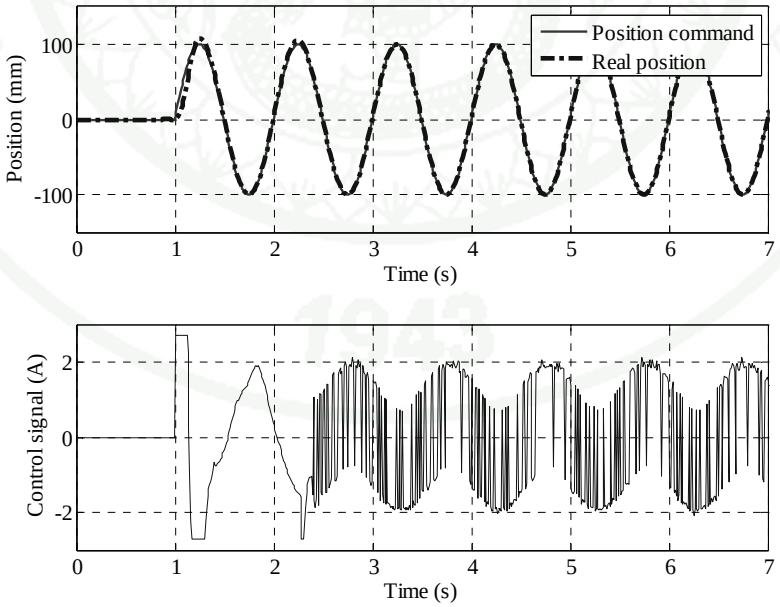


Figure 44 Experimental results of the adaptive sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

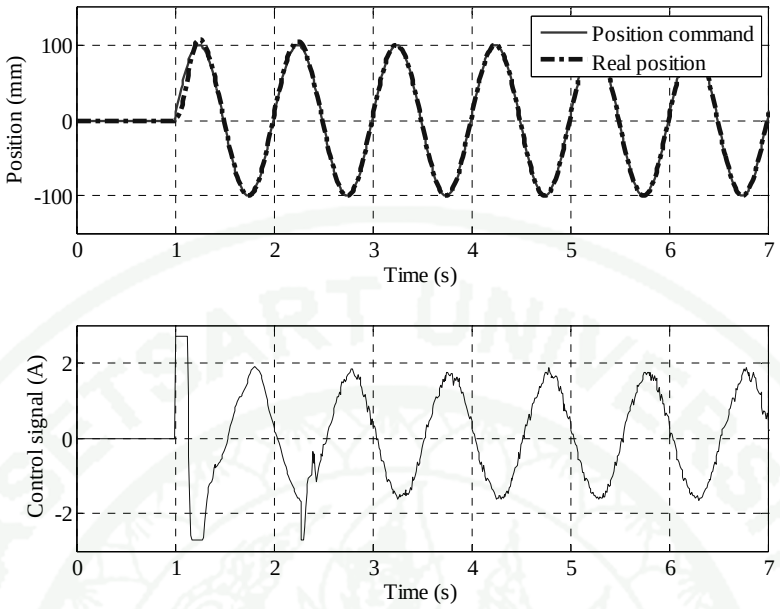


Figure 45 Experimental results of the improved adaptive sliding mode control for Case 2 ($M = \bar{M} + 3.5$ kg)

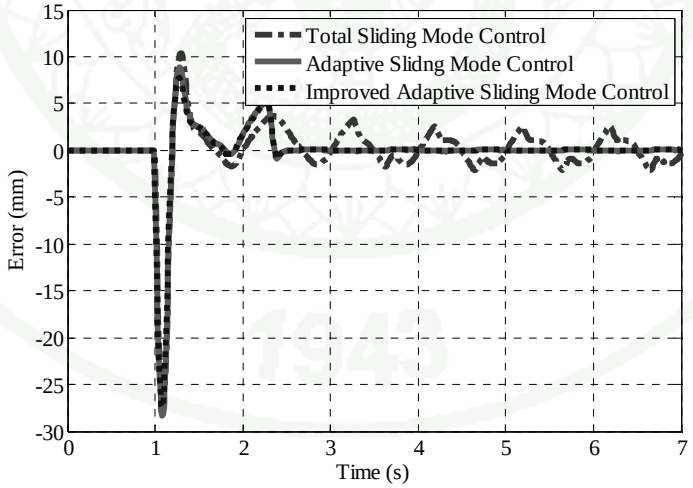


Figure 46 Comparison of experimental results of the Mean Absolute Errors for Case 2 ($M = \bar{M} + 3.5$ kg)

Discussion

Simulation results of parameter identification

The identified results from the simulation are shown in Table 9, Table 10 and Table 11 for test signal input of step, chirp and PRBS inputs, respectively. The verified results are calculated by

$$\text{Percent absolute error} = \frac{|\hat{x} - \bar{x}|}{\bar{x}} \times 100\% \tag{46}$$

where $\bar{x}$ and $\hat{x}$ are the nominal value of LPM motor parameters that is specified following in Table 3 and the identified result respectively. The verified results of using step, chirp and PRBS inputs are shown in Table 12, Table 13 and Table 14, respectively. The average results of percent absolute error of each test signal input are shown in Table 15 that the PRBS inputs can effectively identify the LPM motor parameters compared to the nominal values listed in Table 3. However, as mentioned early, PRBS is not suitable for identifying the LPM motor's parameters. Thus, the chirp input is the alternative test signal input to identify the LPM motor's parameters; only chirp inputs are used to obtain experimental results.

Experimental results of parameter identification

The conditions of the experiments are the same as those for the simulations with the same set of chirp inputs. The actual outputs (position and velocity) are measured using the optical linear encoder. The results of the identified parameters are summarized in Table 16. It is noticeable that the estimated values from the proposed method are slightly deviated from the nominal values shown in Table 3. Thus, the estimated values are calculated by Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| = \frac{1}{n}\sum_{i=1}^{n}|e_i| \tag{47}$$

where n is the number of sampling data, y_i and $\hat{y}_i$ are the output responses of i^{th} sampling data of the real system and estimated model, respectively.

The position and velocity of system outputs are compared. The plots of the best similarity are shown in Figure 29 and Figure 30. However, these deviations from the nominal values do not necessarily indicate the shortcoming of the proposed method since the nominal values themselves are usually not sufficient for high performance controller, especially with different loading conditions or different operating points. Therefore, it is not recommended to compare the identified parameters with the nominal parameters, as also noted by L. Liu *et al.* (2008), but to directly compare the dynamic performances between the identified parameters and the nominal values. Nonetheless, the results of the present work indicate the effectiveness of the proposed PSO method for parameter identification.

Simulation results of sliding mode controllers

The simulation results for the three control systems are shown in Figure 31, Figure 32, Figure 33, for Case 1 and in Figure 35, Figure 36, Figure 37 for Case 2. The computed Mean Absolute Errors are summarized in Table 17 and shown in Figure 34 for Case 1 and in Figure 38 for Case 2. According to Figure 31 to Figure 34 and Table 17, all three control systems have good tracking responses. However, when the total sliding mode control is used, large chattering is very noticeable as shown in Figure 31; whereas when the adaptive sliding mode control is used, the chattering is reduced but still noticeable as shown in Figure 32. When the improved adaptive sliding mode control is used, the chattering is totally eliminated as shown in Figure 33.

Consider Figure 35, Figure 36, Figure 37, Figure 38 and Table 17 for Case 2. For the total sliding mode control as shown in Figure 35, the tracking response performance is satisfactory although some chattering phenomena can be noticed. For the adaptive sliding mode control as shown in Figure 36, the tracking response performance is also satisfactory, but even more chattering phenomena are noticed. For

the improved adaptive sliding mode control as shown in Figure 37, the tracking response performance is also satisfactory, but the chattering phenomena are mostly eliminated. From the simulation results, the improved adaptive sliding mode control is apparently the most suitable choice in terms of performance and capability to reduce the chattering.

Experimental results of sliding mode controllers

The experiments are conducted under the same conditions and the obtained results are similar to those of the simulation results as shown in Figure 39, Figure 40, Figure 41 for Case 1 and in Figure 43, Figure 44, Figure 45 for Case 2. The computed Mean Absolute Error are summarized in Table 18 and shown in Figure 42 for Case 1 and in Figure 46 for Case 2.

For Case 1, the tracking response performance for all three control systems are comparable. The total sliding mode control has the largest chattering as shown in Figure 39, whereas the adaptive control can reduce some chattering as shown Figure 40. When the improved adaptive sliding mode control is used, the chattering is totally eliminated as shown in Figure 41. For Case 2, the tracking response performance for all three control systems are also comparable. However, the proposed improved adaptive sliding mode control outperforms the other two controls in terms of reducing the chattering as shown in Figure 43, Figure 44 and Figure 45 and. From the experimental results, the improved adaptive sliding mode control is clearly suitable for controlling the LPM motor and effective for reducing the chattering phenomena with good performance.

CONCLUSION AND RECOMMENDATIONS

Conclusion

Conclusion of parameter identification

This thesis proposed a PSO identification method to identify the LPM motor parameters. Both simulations and experiments are implemented to validate the effectiveness of this identification method. The results showed that the PSO identification method could be used to identify the mechanical parameters of the LPM motor model with good accuracy. The proposed method can be applied to identify parameters of other systems.

Conclusion of sliding mode controllers

This thesis presents the improved adaptive sliding mode control, which is successfully implemented to control the LPM motor with dSPACE (DS1104) controller card. The simulation and experimental results are conducted for two different cases and then evaluate the performance criterion with Mean Absolute Error (MAE). Although the performance of the improved adaptive sliding mode controller is similar to the adaptive sliding mode controller, the improved adaptive sliding mode control outperforms the total sliding mode control and the adaptive sliding mode control in terms of reducing the chattering phenomena. Finally, the idea of this proposed method of the improved adaptive sliding mode control can be easily applied to other control systems.

Recommendations

Recommendations of parameter identification

Possible future works are to identify other parameters and explore different types of input signals, as well as to determine to what extent the identified parameters could enhance the performance of the controller in comparison with using just the nominal parameter values from the datasheet in the controller.

Recommendations of sliding mode controllers

Although the performance of the improved adaptive sliding mode controller outperforms the other two controls in terms of reducing the chattering phenomena, the control parameters of the improved adaptive sliding mode controller are quite difficult to suitably adjust to control the LPM motor. Therefore, the control parameters of the improved adaptive sliding mode controller should be automatically adjust to get the higher performance in terms of reducing the chattering phenomena and keeping the good tracking responses.

LITERATURE CITED

Boldea, I. and S. A. Nasar. 1997. **Linear Electric Actuators and Generators.** Cambridge University Press, Cambridge, U. K..

Clerc, M. 2006. **Particle Swarm Optimization.** ISTE Publishing Company, London, U.K..

Chou, W. D., F. J. Lin and K. K. Shyu. 2003. Incremental motion control of an induction motor servo drive via a genetic-algorithm-based sliding mode controller. **Proceedings of IEE Control Theory and Applications** 150 (3): 209-211.

Gieras, J. F. and Z. J. Piech. 1999. **Linear Synchronous Motors: Transportation.** CRC Press, Boca Raton, FL.

Hasni, M., O. Touhami, M. Fadel and S. Caux. 2008. Synchronous machine parameter identification by various excitation signals. **Journal of Electrical Engineering** 90 (3): 219-228.

Hsu, S. C., C. Hao Liu, C. Huan Liu and N. J. Wang. 2001. Fuzzy PI controller tuning for a linear permanent magnet synchronous motor drive, pp. 1661-1666. *In* **Proceedings of IEEE Industrial Electronics Society** .

_____, W. D. Liu and C. H. Liu. 2000. Parameter auto-tuning for a linear permanent magnet synchronous motor drive, pp. 1031-1036. *In* **Proceedings of IEEE on Industrial Electronics Society** . Nagoya, Japan.

Kennedy, J. and R. Eberhart. 1995. Particle Swarm Optimization, pp. 1942-1948. *In* **Proceedings of IEEE International Conference on Neural Networks** . Perth, Australia.

Khalil, H. K. 1992. **Nonlinear systems.** Macmillan, New York.

Liu, T. H., Y. C. Lee and Y. H. Chang. 2004. Adaptive controller design for a linear motor control system. **IEEE Transactions on Aerospace and Electronic Systems** 40 (2): 601-616.

Liu, L. and D. A. Cartes. 2008. Permanent magnet synchronous motor parameter identification using particle swarm optimization. **International Journal of Computational Intelligence Reseach** 4 (2): 211-218.

Ljung, L. 1987. **System Identification: Theory for the User.** Prentice-hall, Englewood Cliffs.

Michalik, W. 1998. Standstill estimation of electrical parameters in motors with optimal input signals, pp. 407-413. *In* **Proceeeding of IEEE International Caracas Conference on Devices, Circuits and Systems** . Margarita, Venezuela.

Pacas, M., S. Vollwock, P. Szczupak and H. Zoubek. 2010. Methods for commissioning and identification in drives. **International Journal of Computation and Mathematics in Electrical and Electronic Engineering** 29 (1): 53-71.

Shyu, K. K., J. Y. Hung and J. C. Hung. 1999. Total sliding mode trajectory control of robotic manipulators, pp. 1062-1066. *In* **Proceedings of IEEE Industrial Electronics** .

Slotine, J. E. and W. Li. 1991. **Applied Nonlinear Control.** Prentice-Hall, Englewood Cliffs, NJ.

Sudhoff, S. D., J. Cale, B. Cassimere and M. Swinney. 2005. Genetic algorithm based design of a permanent magnet synchronous machine, pp. 1011-1019. *In* **Proceeding of IEEE on Electric Machines and Drives** . Texas, U.S..

Utkin, V. I., J. Guldner and J. Shi. 1999. **Sliding Mode Control in Electromechanical Systems.** Taylor & Francis, New York.

Wai, R. J. 2001. Total sliding-mode controller for PM synchronous servo motor drive using recurrent fuzzy neural network. **IEEE Transactions on Industrial Electronics** 48 (5): 926-944. *Cited* R. J. Wai 2000. Adaptive sliding-mode control for induction servomotor drive. **Proceedings of IEE Electric Power Applications** 147 (6): 553-561.

APPENDICES

Appendix A

Difficulty calculation

The difficulty calculation can be referred in M. Clerc (2006).

Function $\sum_{\mathbf{d=1}}^{\mathbf{D}} \mathbf{x_d}$

Let us call p the probability of success and e the required precision. One has successively:

$$\begin{aligned} p &= p\left(\sum_{d=1}^{D} x_d < \epsilon\right) = \int_0^{\epsilon} p\left(\sum_{d=1}^{D-1} x_d < \epsilon - u\right) du \\ &= \frac{\epsilon^D}{D!} \end{aligned} \tag{A.1}$$

the last equality being obtained easily by recurrence. But this is valid only if all the components are picked at random from the interval $\left[\ 0 \quad 1\ \right]$. If the real interval is $\left[\ 0 \quad R\ \right]$, this result must be multiplied by $\frac{1}{R}^D$. Finally, we obtain:

$$\mathit{difficulty} = \ln(\mathrm{D!}) - \mathrm{D}\ln(\epsilon) + \mathrm{D}\ln(\mathrm{R}) \tag{A.2}$$

Function $\sum_{\mathbf{d=1}}^{\mathbf{D}} \mathbf{x_d^2}$

Here, calculation is even simpler provided its formulae are known. Effectively we want the probability of $\sum_{d=1}^{D} x_d^2 < \epsilon$ for $0 \leq x_d \geq 1$. It is therefore enough to work out the ratio of the volume of the hypersphere of dimension D and radius $\sqrt{\epsilon}$ and of the volume of the hypercube of edge 2. It is given by the traditional formula:

$$\begin{cases} \dfrac{\pi^{D'}}{D'!}\sqrt{D}\frac{1}{2^D} \text{ si } \mathrm{D} = 2\mathrm{D}' \\ \pi^{\mathrm{D}'}\dfrac{2^{\mathrm{D}}}{\mathrm{D}'!}\sqrt{\mathrm{D}}\frac{1}{2^{\mathrm{D}}} \text{ si } \mathrm{D} = 2\mathrm{D}' + 1 \end{cases} \tag{A.3}$$

As before, if the hypercube is of edge $2R$, it is necessary to multiply by $\frac{1}{R}^D$.

Function $\sum_{\mathbf{d=1}}^{\mathbf{D}} \sqrt{\mathbf{x_d}|\sin(\mathbf{x_d})|}$

Here direct analytical determination is tricky. It would certainly be possible to use an expansion of a finite series, but let's take a lazier method of estimation, which nevertheless requires the use of a computer.

We take a very small search space $\left[\ 0 \quad r\ \right]^D$, such that there are nevertheless points in which the function has a greater value than the tolerance level, ϵ. For $D = 10$, one can take $r = 0.005$.

We can at random draw a great number of items (10^8 in the example), each time we evaluate the function, in order to see whether we obtain a value less than ϵ or not. We deduce an estimate from the success rate, τ. In the example, one obtains $\tau = 0.999732$. Note that it is necessary to use a good pseudo random number generator. For example, the function $rand$ in the programming language C is not always appropriate.

One then calculates that on the search space $\left[\ 0 \quad r\ \right]^D$, the success rate would be only $\tau' = \tau \left(\frac{r}{R}\right)^D$. The measurement of corresponding difficulty is thus:

$$difficulty = -\ln(\tau) - \mathrm{D}\ln(\mathrm{r}) + \mathrm{D}\ln(\mathrm{R}) \tag{A.4}$$

Note that this estimate is a little pessimistic as soon as $R \geq \pi$, since there are then several global minima (every point where $\sin(x_d) = 0$, for all x_d). The number of these points is $n = (1 + Ent(R/\pi))^D$, but the further one moves from the origin of the coordinates, the more the corresponding minimum is "pointed" and the less, therefore, its existence reduces the difficulty of the problem. The fact is, moreover, that PSO never finds them before the origin of the coordinates itself (as long as, of course, this is in the search space).

Appendix B

Adaptive sliding mode theory

This proof can be referred in R. J. Wai (2000).

Theorem

If the perturbed LPM motor control system is controlled by using adaptive sliding mode control law, then asymptotic stability of the adaptive sliding mode control can be guaranteed.

Proof Choosing a Lyapunov function candidate.

$$V(S(t),\tilde{\rho}(t)) = \frac{1}{2}\left[S^2(t) + \lambda\tilde{\rho}^2(t)\right] \tag{B.1}$$

where $S(t)$ is sliding surface; $\tilde{\rho}(t)$ is defined as $\hat{\rho}(t) - \rho(t)$; ρ and $\hat{\rho}$ are the control gain and the estimated control gain; λ is the learning rate. Taking the derivative of the Lyapunov function can obtain

$$\dot{V}(S(t),\tilde{\rho}(t)) = S(t)\dot{S}(t) + \lambda\tilde{\rho}(t)\dot{\tilde{\rho}}(t) \tag{B.2}$$

Substituting (37), (38) into (B.2) then

$$\begin{aligned}
\dot{V} &= S(t)[U_b + C_{2n}^{-1}W(t)] + \lambda\tilde{\rho}(t)\dot{\tilde{\rho}}(t) \\
&= -\hat{\rho}C_{2n}{-}1|S(t)| + C_{2n}^{-1}S(t)W(t) + \lambda(\hat{\rho}(t) - \rho)\dot{\hat{\rho}}(t) \\
&= -\hat{\rho}C_{2n}{-}1|S(t)| + C_{2n}^{-1}S(t)W(t) + (\hat{\rho}(t) - \rho)C_{2n}{-}1|S(t)| \\
&= -\rho C_{2n}^{-1}|S(t)| + C_{2n}^{-1}S(t)W(t) \\
&\leq -\rho C_{2n}^{-1}|S(t)| + C_{2n}^{-1}|S(t)||W(t)| \\
&= -C_{2n}^{-}1|S(t)|[\rho - |W(t)|] \leq 0
\end{aligned} \tag{B.3}$$

Since $\dot{V}(S(t),\tilde{\rho}(t)) \leq 0$, $\dot{V}(S(t),\tilde{\rho}(t))$ is negative semidefinite (i.e. $\dot{V}(S(t),\tilde{\rho}(t)) \leq \dot{V}(S(0),\tilde{\rho}(0))$), which implies $S(t)$ and $\tilde{\rho}(t)$ are bounded. Let function $\Xi = -\dot{V}(S(t),\tilde{\rho}(t)) = C_{2n}^{-1}|S(t)|[\rho - W(t)]$, and integrate function Ξ with respect to time to obtain

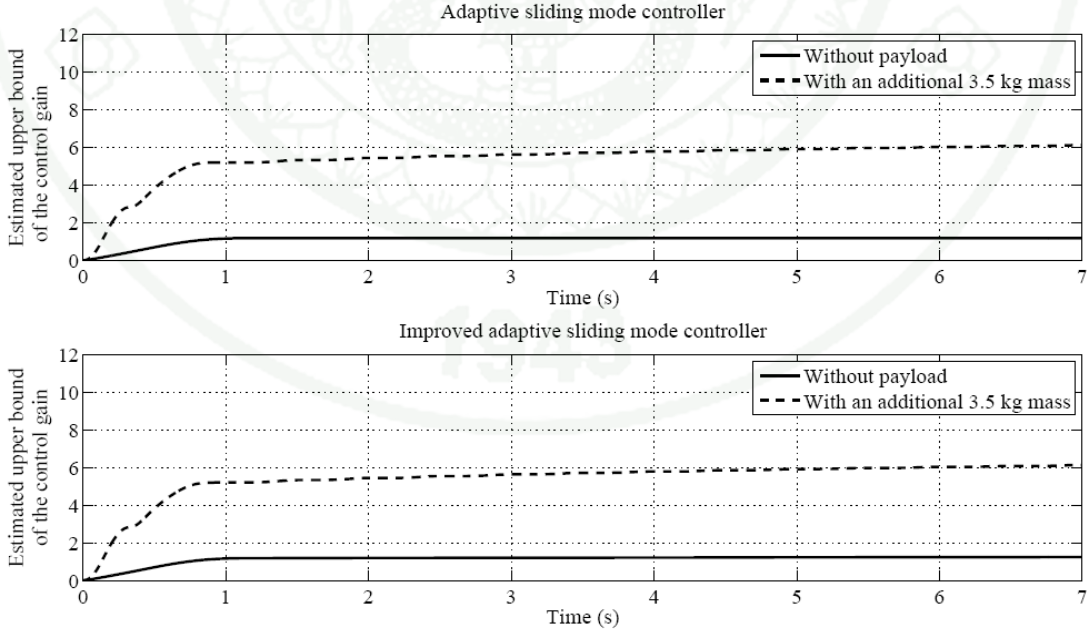
$$\int_0^t \Xi(t)d\tau = V(S(0),\tilde{\rho}(0)) - V(S(t),\tilde{\rho}(t)) \tag{B.4}$$

Because $V(S(0),\tilde{\rho}(0))$ is bounded, and $V(S(t),\tilde{\rho}(t))$ is non increasing and bounded, the following result is obtained:

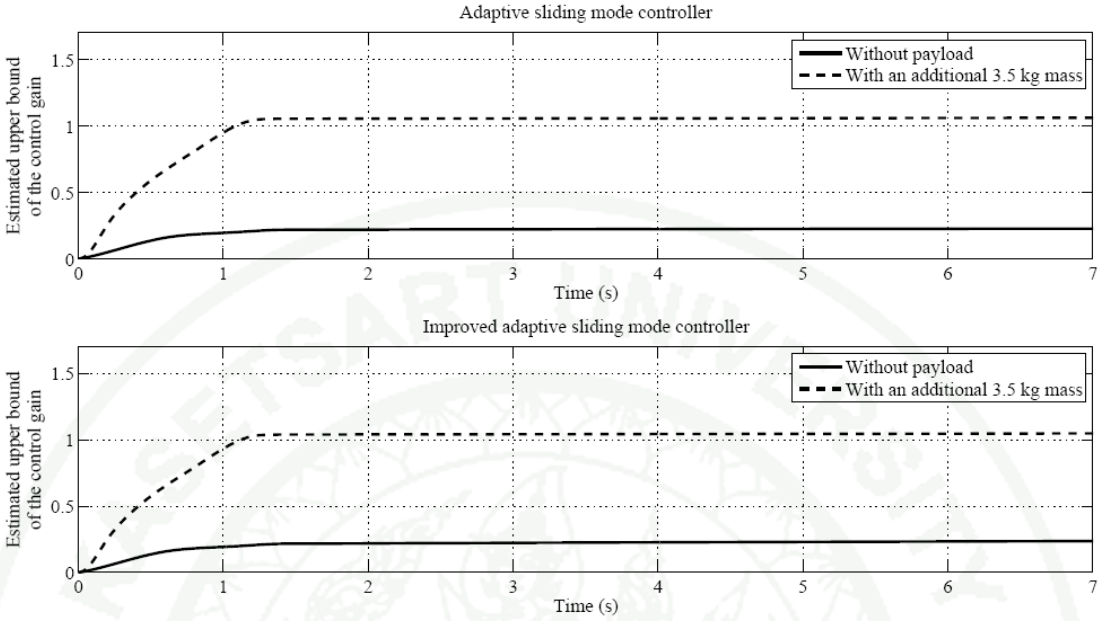
$$\lim_{t\to\infty}\int_0^t \Xi(t)d\tau < \infty \tag{B.5}$$

Also, since $\dot{\Xi}(t)$ is bounded, by Barbalat's lemma in J. E. Slotine and W. Li (1991), it can be shown that $\lim_{t\to\infty}\Xi(t) = 0$. That is, $S(t) \to 0$ as $t \to \infty$.

The simulation and experiment results for the control gain ρ of the adaptive sliding mode controller and the improved adaptive sliding mode controller are shown in Appendix Figure B1 and Appendix Figure B2. The plots clearly show that the control gain is bounded as expected.



Appendix Figure B1 Simulation results of the estimated upper bound of the control gain ρ



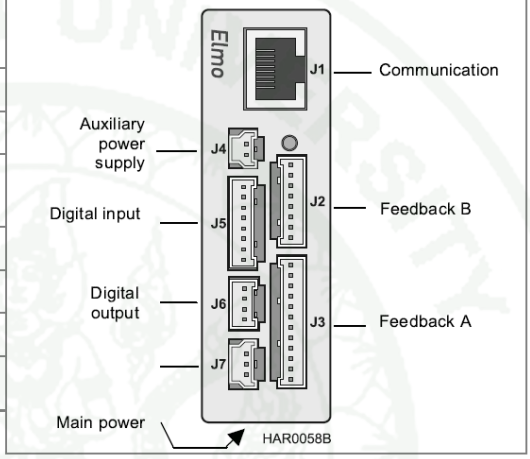
Appendix Figure B2 Experiment results of the estimated upper bound of the control gain ρ

Appendix C

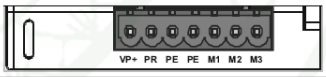
Motor driver information

Here, this provides necessary information about Elmo's motor driver. For full information, please refer in Elmo documents:

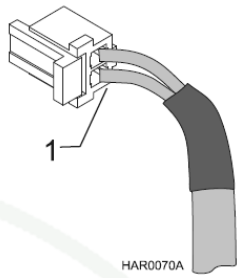
Harmonica Installation Guide

Type	Function	Port	Connector Location
8-pin RJ-45	Communication (CAN or RS-232)	J1	
8-pin Molex	Feedback B	J2	
12-pin Molex	Feedback A	J3	
2-pin Molex	Auxiliary power supply	J4	
8-pin Molex	Digital input	J5	
4-pin Molex	Digital output	J6	
3-pin Molex	Analog input	J7	
7-pin Phoenix	Main power	J8	

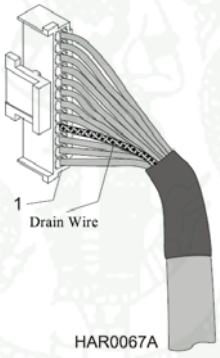
Appendix Figure C1 Harmonica Connectors

Pin	Function	Cable		Pin Positons
VP+	Power input, positive	Power		
PR	Power input, common	Power		
PE	Protective earth	Power		
		AC Motor Cable	DC Motor Cable	
PE	Protective earth	Motor	Motor	
M1	Motor phase	Motor	N/C	
M2	Motor phase	Motor	Motor	
M3	Motor phase	Motor	Motor	
	When connecting several motors, all must be wired in an identical manner.			

Appendix Figure C2 Power cable connector (J8)

Pin	Signal	Function	Pin Position
1	+24VDC	+24 VDC auxiliary power supply	
2	RET24VDC	Return (common) of the 24 VDC auxiliary power supply	

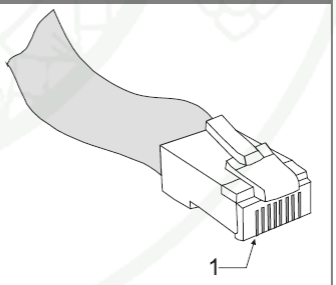
Appendix Figure C3 Auxiliary power cable connector (J4)



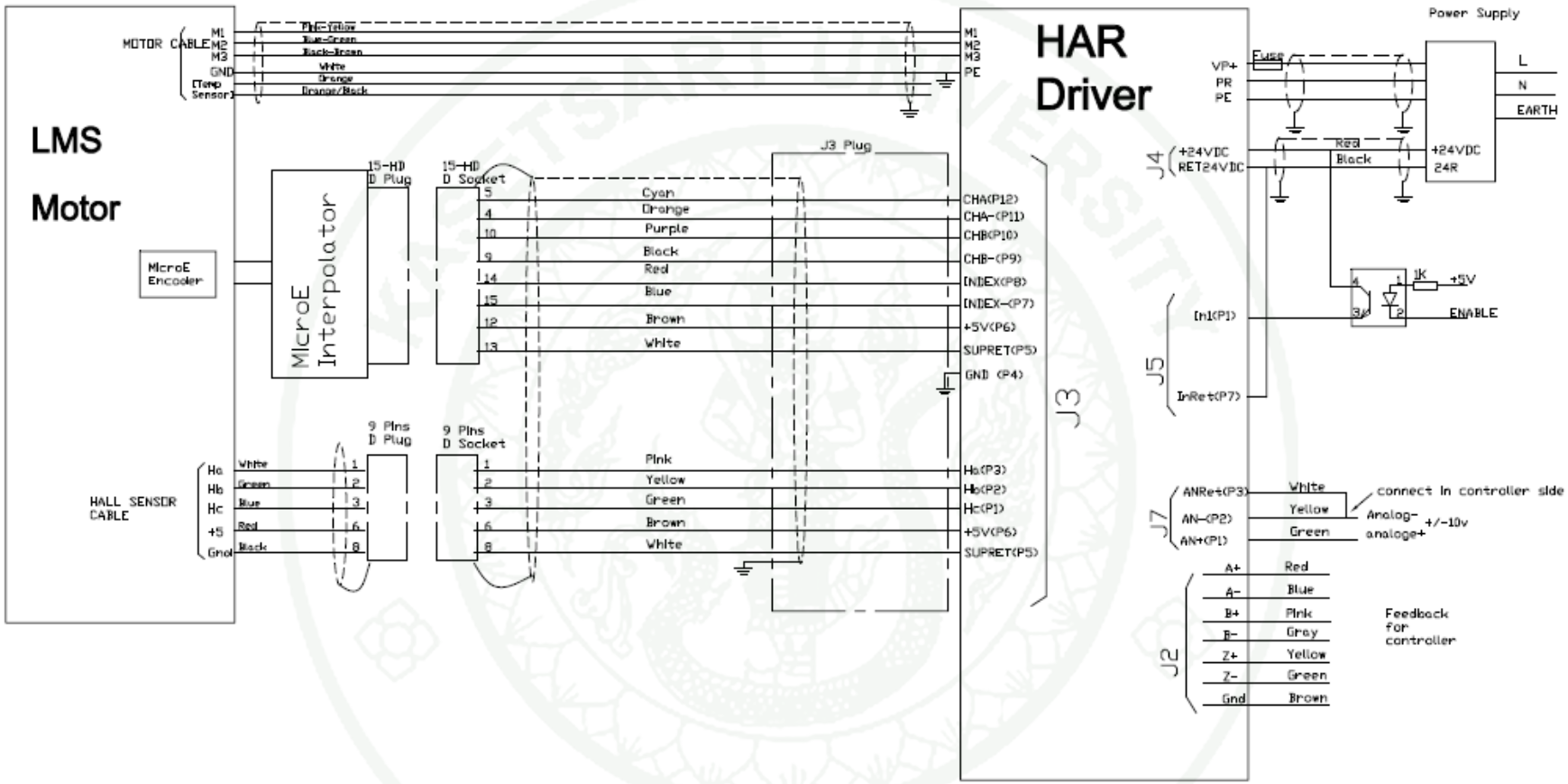
Appendix Figure C4 Main feedback connector

	Incremental Encoder		Interpolated Analog (Sine/Cosine) Encoder		Resolver	
	HAR XX/YYY_		HAR XX/YYYI		HAR XX/YYYR	
Pin	Signal	Function	Signal	Function	Signal	Function
1	HC	Hall sensor C input	NC	-	NC	-
2	HB	Hall sensor B input	NC	-	NC	-
3	HA	Hall sensor A input	NC	-	NC	-
4	SUPRET	Supply return	SUPRET	Supply return	SUPRET	Supply return
5	SUPRET	Supply return	SUPRET	Supply return	SUPRET	Supply return
6	+5V	Encoder/Hall +5 V supply voltage, 5 V @ 200 mA maximum	+5V	Encoder/Hall +5 V supply voltage, 5 V @ 200 mA maximum	NC	-
7	INDEX-	Index complement	INDEX-	Index complement	VREF-	Vref cmplmnt Vref 3.5RMS S= 1/TS, 50mA Max.
8	INDEX	Index	INDEX	Index	Data	Vref 3.5 vrms, S=1/TS, 50mA
9	CHB-	Channel B cmplmnt	CHB-	Channel B complement	CHB-	Cos B complement
10	CHB	Channel B	CHB	Cos B	CHB	Cos B
11	CHA-	Channel A cmplmnt	CHA-	Sine A complement	CHA-	Sine A complement
12	CHA	Channel A	CHA	Sin A	CHA	Sin A

Appendix Figure C5 Main feedback signals

Pin	Signal	Function	Pin Location
1	–	–	
2	–	–	
3	Tx	RS-232 transmit	
4	–	–	
5	COMRET	Communication return	
6	Rx	RS-232 receive	
7	–	–	
8	–	–	

Appendix Figure C6 RS-232 communication connector (J1)



Appendix Figure C7 LPM motor wiring diagram for force/current control mode

Appendix D

Publication papers

An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor

Tithiwat Therdbankerd*, Peerayot Sanposh†, Nattapon Chayopitak‡ and Hideaki Fujita§
* TAIST Tokyo Tech, ICTES Program
*† Intelligent Mechatronics, Automation, Robotics, and Control Laboratory,
Department of Electrical Engineering, Kasetsart University
50 Phahonyothin Rd., Ladyao, Jatujak, Bangkok, 10903, Thailand
Email: tithiwat@msn.com*, peerayot.s@ku.ac.th†
‡ Industrial Control and Automation Laboratory, National Electronics and Computer Technology Center
112 Thailand Science Park, Phahonyothin Rd., Klong 1, Klong Luang, Pathumthani, 12120, Thailand
Email: nattapon.chayopitak@nectec.or.th
§ Department of Electrical and Electronic Engineering, Tokyo Institute of Technology
S3-18, 2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8550, Japan
Email: hide@ee.titech.ac.jp

***Abstract*—Linear permanent magnet motors have been widely used to provide linear motion for high-performance applications without using conventional gears, screws or crack shafts. One approach to effectively control the linear permanent magnet motor is to use the total sliding mode control based on adaptive algorithm which is robust against uncertainty parameter variations and external disturbances. However, one drawback from using such approach is the chattering phenomena once the system has reached the sliding surface. Hence, an improved adaptive sliding mode control is proposed to reduce the chattering. For comparison, the simulations and experiments are implemented for three different controllers: the conventional total sliding mode controller, the adaptive sliding mode controller and the proposed improved adaptive sliding mode controller. From the simulation and experimental results, the improved adaptive sliding mode control is clearly suitable for controlling the LPM motor and effective for reducing the chattering phenomena with good performance.**

***Index Terms*—sliding mode control, adaptive control, linear permanent magnet motors, motion control.**

I. Introduction

Linear motors are electric motors that can operate without conventional gears, screws or crack shafts to provide direct linear motion. Linear Permanent Magnet (LPM) motors, particularly, have been widely used in many applications demanding faster speed and better accuracy such as computer-controlled machine tools, semiconductor manufacturing equipments and inspection machines.

In order to achieve high performance of the LPM motors, many control algorithms have been considered such as Fuzzy-PI control [4], adaptive control [6] and sliding mode control [9]. In particular, the sliding mode control has simple structure and robustness against uncertain perturbations and external disturbances.

Traditional sliding mode control has two major weaknesses: (i) the control system is not robust while the system is still on the reaching phase before entering the sliding surface, and (ii) the system is suffered from the chattering phenomena after reaching the sliding surface. One approach to solve the first problem is the total sliding mode control [8] where the reaching phase is eliminated by initializing the control system on the sliding surface. The adaptive sliding control [10] is proposed to alleviate the second problem by utilizing online an parameter tuning technique to reduce the chattering.

Although the adaptive sliding mode control has simple structure and can effectively reduce some of the chattering phenomena, it is still susceptible to large external disturbances and large parameter variations. The total sliding mode with Neural Networks [11] and the sliding mode control with real-time Genetic algorithm [2] are later proposed to address such problems; however, these methods require high-performance processor for complicate computation.

Therefore, this paper proposes an improved adaptive sliding mode control using the saturation function to effectively reduce the chattering phenomena. The proposed technique is based on the adaptive sliding mode control to maintain the simple structure and computational

Appendix Figure D1 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page1

efficiency. The simulations and experiments have been conducted to control the LPM motor drive system under large parameter variations to demonstrate the effectiveness of the proposed method.

II. Description Linear Permanent Magnet Motor Control

The mathematical model of the LPM motor shown in Fig. 1, in dq domain of the synchronous rotating frame is given by [1]

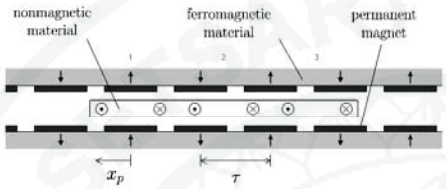


Fig. 1. LPM motor structure.

$$\dot{i}_d = -\frac{R_s}{L_d} i_d + \frac{\pi}{\tau}\frac{L_q}{L_d}\dot{x}_p i_q + \frac{1}{L_d} v_d \quad (1)$$

$$\dot{i}_q = -\frac{L_d}{L_q}\dot{x}_p i_d - \frac{R_s}{L_q} i_q - \frac{\pi}{\tau}\frac{\lambda_f}{L_q}\dot{x}_p + \frac{1}{L_q} v_q \quad (2)$$

$$F_e = K_f i_q \quad (3)$$

where x_p and $\dot{x}_p$ are the displacement and velocity of the LPM motor; p and τ are the number of pole pairs and pole pitch; F_e is the motor force; v_d, v_q, i_d and i_q are the dq-components of voltages and currents in the dq-axis reference frame, respectively; R_s, L_d, L_q and λ_f are the phase winding resistance, d-axis and q-axis inductances, and the permanent magnet flux linkage, respectively; K_f is the motor force constant defined as

$$K_f = \frac{3}{2}\frac{\pi}{\tau} p \lambda_f \quad (4)$$

The position movement of the LPM motor is given by

$$F_e = M\ddot{x}_p + B\dot{x}_p + F_L \quad (5)$$

where

F_L : the external force
M : the total mass of the moving part
B : the viscous friction coefficient
$\ddot{x}_p$: the acceleration

Since the force produced by the LPM motor only depends on i_q, it is customary to control $i_d = 0$ [3], resulting in i_q chosen as a control variable U.

III. Total Sliding Mode Control

A. Baseline Model Control

The total sliding mode control, as shown in Fig. 2, has a simple structure and can be implemented in a real-time control system [8]. Following [10] the control law of the total sliding mode control is given as

$$U = U_{\mathrm{BMC}} + U_{\mathrm{b}} \quad (6)$$

where

U : Control Variable
U_{BMC} : Baseline Model Control
U_{b} : Curbing Control

The first term of the total sliding mode is called the Baseline Model Control (BMC) that is used to specify the desired performance of the nominal model. By rearranging (3), (4) and (5), we obtain

$$\ddot{x}_p(t) = -\frac{B}{M}\dot{x}_p(t) + \frac{K_f}{M} i_q^*(t) - \frac{1}{M} F_L \quad (7)$$

$$\triangleq C_1\dot{x}_p(t) + C_2 U(t) + C_3 F_L \quad (8)$$

where $C_1 = -\frac{B}{M}$, $C_2 = \frac{K_f}{M} > 0$, and $C_3 = -\frac{1}{M}$; $U(t) = i_q^*(t)$ is a control variable. By neglecting parameter variation and external force disturbance, (8) can be rewritten to obtain the nominal model as

$$\ddot{x}_p(t) = C_{1n}\dot{x}_p(t) + C_{2n} U(t) \quad (9)$$

where $C_{1n} = -\frac{\bar{B}}{\bar{M}}$ and $C_{2n} = \frac{\bar{K}_f}{\bar{M}}$ are the nominal values of C_1 and C_2, indicated by the overbar symbol. For example $\bar{K}_f$, $\bar{M}$ and $\bar{B}$ are the nominal values of K_f, M and B, respectively. When the uncertainties or the external force disturbances present, these parameters are deviated from their nominal values. The dynamic position movement (9) can be modified to

$$\begin{aligned}\ddot{x}_p(t) &= (C_{1n} + \triangle C_1)\dot{x}_p(t) \\ &\quad + (C_{2n} + \triangle C_2) U(t) + (C_{3n} + \triangle C_3) F_L \\ &= C_{1n}\dot{x}_p(t) + C_{2n} U(t) + W(t) \end{aligned} \quad (10)$$

where $C_{3n} = -\frac{1}{\bar{M}}$ is the nominal value of C_3; $\triangle C_1$, $\triangle C_2$, $\triangle C_3$, and F_L are the uncertainties; and $W(t)$ is the lumped uncertainty, explicitly given by

$$W(t) = \triangle C_1\dot{x}_p(t) + \triangle C_2 U(t) + (C_{3n} + \triangle C_3) F_L \quad (11)$$

Moreover, the lumped uncertainty should be bounded by

$$|W(t)| < \rho \quad (12)$$

The selection of ρ affects the chattering phenomena and the task for adjusting ρ is non-trivial. If ρ is too large,

Appendix Figure D2 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page2

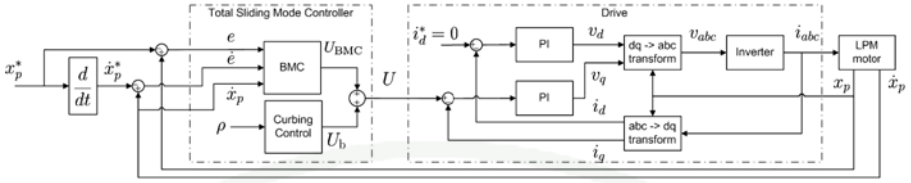


Fig. 2. Total Sliding Mode control system.

the chattering bound will be large. If ρ is too small, the system performance might be unstable. From (9), the BMC control law is given by

$$U_{\mathrm{BMC}} = U_{\mathrm{c}} + U_{\mathrm{s}} \tag{13}$$

where

$$U_{\mathrm{c}} = -C_{2n}^{-1} C_{1n} \dot{x}_p \tag{14}$$

$$U_{\mathrm{s}} = C_{2n}^{-1} [\ddot{x}_p - K_p e - K_v \dot{e}] \tag{15}$$

in which U_{c} is used to cancel the nonlinear terms in the model; U_{s} is used to specify the desired system performance; K_p and K_v are nonzero positive constants. Define the tracking error as $e \triangleq x_p - x_p^*$, where x_p^* is the position command. By rearranging (9), (13), (14) and (15), we obtain the controlled nominal system

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \tag{16}$$

where K_p and K_v can be chosen to specify the desired second order system performance. However, only BMC cannot guarantee the desired system performance due to the potential lumped uncertainty $W(t)$. To ensure the desired system performance, we need the second term U_{b} of the control law as described in the next section.

B. Curbing Control

The second term of the total sliding mode control is called Curbing Control that is used to eliminate uncertain perturbation effects from the parameter variation and external force disturbance. The curbing control law is given by

$$U_{\mathrm{b}} = -\rho C_{2n}^{-1} sgn(S(t)) \tag{17}$$

where $\rho > 0$ is the control gain as defined in (12), $sgn(\cdot)$ is the sign function and $S(t)$ is the sliding surface as designed in [8], [10], [11]. From (16), we obtain the state variable form

$$\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} \tag{18}$$

or

$$\dot{\mathbf{E}} = \mathbf{A}\mathbf{E} \tag{19}$$

where

$$\mathbf{E} = \begin{bmatrix} e & \dot{e} \end{bmatrix}^T \tag{20}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -K_p & -K_v \end{bmatrix} \tag{21}$$

Now, considering the sliding surface:

$$S(t) = C(\mathbf{E}) - C(\mathbf{E_0}) - \int_0^t \frac{\partial C}{\partial \mathbf{E}} \mathbf{A}\mathbf{E} dt \tag{22}$$

where $C(\mathbf{E})$ is the vector to be designed such that $\frac{\partial C}{\partial E} = \begin{bmatrix} 0 & C_{2n}^{-1} \end{bmatrix}$, and $\mathbf{E_0}$ is the initial state of $\mathbf{E}$. The objective of Curbing Control is to maintain the controlled system on the sliding surface, $S(t) = 0$, at all time. From (22), it can be seen that this sliding surface has no reaching phase because $S(t) = 0$ when $t = 0$, compared with the traditional sliding mode controller [5], [7], [9].

IV. Adaptive Sliding Mode Controls

A. Adaptive Sliding Mode Control

The adaptive sliding mode control [10], Fig. 3, is a method to adjust ρ automatically in real-time for the total sliding mode controller. The control law of the adaptive sliding mode control is given by

$$U = U_{\mathrm{BMC}} + \hat{U}_{\mathrm{b}} \tag{23}$$

where $\hat{U}_{\mathrm{b}}$ is the adaptive curbing control

$$\hat{U}_{\mathrm{b}}(t) = -\hat{\rho}(t) C_{2n}^{-1} sgn(S(t)) \tag{24}$$

and

$$\dot{\hat{\rho}}(t) = \frac{1}{\lambda} C_{2n}^{-1} |S(t)| \tag{25}$$

where $\hat{\rho}$ is the estimated value of ρ, and $\lambda > 0$ is the learning rate of the adaptive algorithm. For this control, the control gain parameter, ρ, is selected the

Appendix Figure D3 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page3

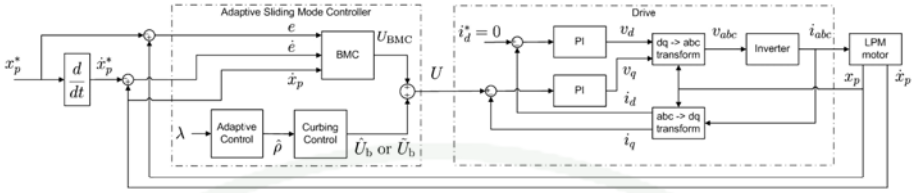


Fig. 3. Adaptive Sliding Mode control system.

upper bound in real-time and effect from pre-selecting too large ρ by the total sliding mode controller is eliminated. If the nominal values of K_f, M and B are deviated from the nominal values, the upper bound of $|W(t)|$. And ρ in (12) will be automatically adjusted to be greater than $|W(t)|$. However, some chattering phenomena still appears using the adaptive sliding mode control.

B. Improved Adaptive Sliding Mode Control

To reduce the chattering phenomena, this paper presents a method to improve the adaptive sliding mode control by using a continuous approximation method [5], [7], [9]. The proposed improved adaptive sliding mode control law is given by

$$U = U_{\rm BMC} + \tilde{U}_{\rm b} \tag{26}$$

where $\tilde{U}_{\rm b}$ is the proposed improved adaptive sliding mode control given by

$$\tilde{U}_{\rm b} = -\hat{\rho}(t) C_{2n}^{-1} sat\left(\frac{S(t)}{\epsilon}\right) \tag{27}$$

The sign function of $\hat{U}_{\rm b}$ in (24) is replaced by the high slope saturation function. The saturation function with high slope is defined as

$$sat\left(\frac{y}{\epsilon}\right) = \begin{cases} \dfrac{y}{\epsilon} & , \text{ if } |y/\epsilon| \le 1 \\ sgn(\dfrac{y}{\epsilon}) & , \text{ if } |y/\epsilon| > 1 \end{cases} \tag{28}$$

where ϵ is a non-zero positive constant. A good approximation requires a small value of ϵ [5], [7]. For this control, the chattering phenomena can be reduced when the nominal values of K_f, M and B are deviated from the nominal values.

Fig. 4. Experimental setup.

V. Simulation and Experimental Results

The simulations and experiments, as shown in Fig. 4, are implemented for three control systems: (i) the total sliding mode control, (ii) the adaptive sliding mode control, and (iii) the proposed improved adaptive sliding mode control. These control systems have been applied to control the three-phase LPM motor. The simulation are implemented on a Pentium 4, 3.0 GHz PC with MATLAB & SIMULINK and the dSPACE (DS1104) controller card. All simulations and experiments are tested for two cases: (i) without payload and (ii) with an additional 3.5 kg payload, i.e.

$$\begin{aligned} &\text{Case 1}: \quad M = \bar{M} \\ &\text{Case 2}: \quad M = \bar{M} + 3.5 \text{ kg} \end{aligned} \tag{29}$$

The sinusoidal function is used as the position reference input x_p^*. The performance criterion for the three control systems is the Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |x_{pi} - x_{pi}^*| = \frac{1}{n}\sum_{i=1}^{n} |e_i| \tag{30}$$

where n is the number of data points, x_{pi} and x_{pi}^* are the actual position and the position reference input respectively. The nominal values of the LPM motor and control parameters are listed in Table I.

Appendix Figure D4 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page4

TABLE I
THE LPM MOTOR AND CONTROL PARAMETERS

LPM motor parameters	Values
K_f	10.86 N/A
$\bar{M}$	1.4 kg
$\bar{B}$	2 Ns/m
Control parameters	**Values**
K_p	2500
K_v	100
ρ	3
λ	0.01
ϵ	0.002

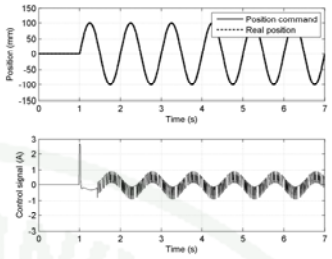
A. Simulation Results

The simulation results for the three control systems are shown in Fig. 5 for Case 1 and in Fig. 6 for Case 2; the computed Mean Absolute Errors are summarized in Table II and shown in Fig. 5(d) for Case 1 and in Fig. 6(d) for Case 2. According to Fig. 5 and Table II, all three control systems have good tracking responses. However, when the total sliding mode control is used, large chattering is very noticeable as shown in Fig. 5(a); whereas when the adaptive sliding mode control is used, the chattering is reduced but still noticeable as shown in Fig. 5(b). When the improved adaptive sliding mode control is used, the chattering is totally eliminated as shown in Fig. 5(c).

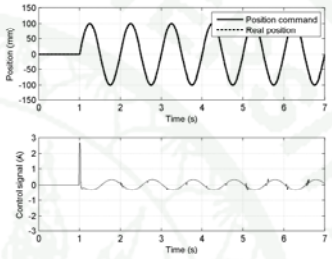
Consider Fig. 6 and Table II for Case 2. For the total sliding mode control as shown in Fig. 6(a), the tracking response performance is satisfactory although some chattering phenomena can be noticed. For the adaptive sliding mode control as shown in Fig. 6(b), the tracking response performance is also satisfactory, but even more chattering phenomena are noticed. For the improved adaptive sliding mode control as shown in Fig. 6(c), the tracking response performance is also satisfactory, but the chattering phenomena are mostly eliminated. From the simulation results, the improved adaptive sliding mode control is apparently the most suitable choice in terms of performance and capability to reduce the chattering.

TABLE II
MEAN ABSOLUTE ERROR COMPARISON OF SIMULATION RESULTS

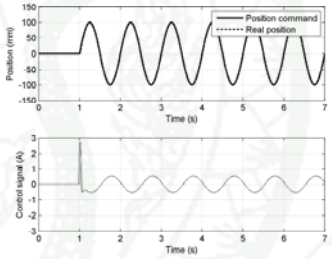
Control system	Case 1 (mm)	Case 2 (mm)
Total Sliding Mode	0.08	1.07
Adaptive Sliding Mode	0.21	0.64
Improved Adaptive Sliding Mode	0.21	0.64



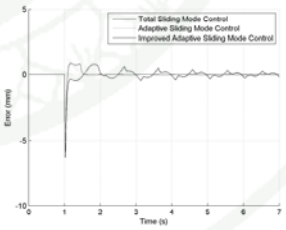
(a) The total sliding mode control.



(b) The adaptive sliding mode control.



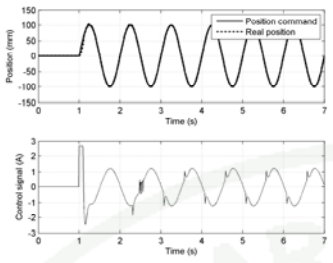
(c) The improved adaptive sliding mode control.



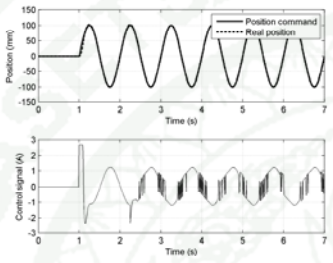
(d) Comparison of the Mean Absolute Errors.

Fig. 5. Simulation results for Case 1 ($M = \bar{M}$).

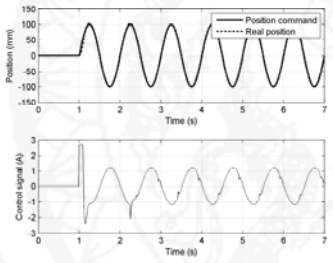
Appendix Figure D5 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page5



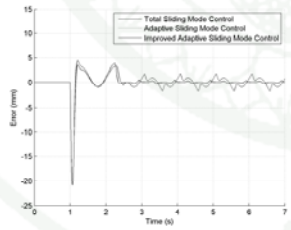
(a) The total sliding mode control.



(b) The adaptive sliding mode control.



(c) The improved adaptive sliding mode control.



(d) Comparison of the Mean Absolute Errors.

Fig. 6. Simulation results for Case 2 ($M = \bar{M} + 3.5$ kg).

B. Experimental results

The experiments are conducted under the same conditions and the obtained results are similar to those of the simulation results as shown in Fig. 7 for Case 1 and in Fig. 8 for Case 2; the computed Mean Absolute Error are summarized in Table III and shown in Fig. 7(d) for Case 1 and in Fig. 8(d) for Case 2.

For Case 1, the tracking response performance for all three control systems are comparable. The total sliding mode control has the largest chattering as shown in 7(a), whereas the adaptive control can reduce some chattering as shown Fig. 7(b). When the improved adaptive sliding mode control is used, the chattering is totally eliminated as shown in Fig. 7(c). For Case 2, the tracking response performance for all three control systems are also comparable. However, the proposed improved adaptive sliding mode control outperforms the other two controls in terms of reducing the chattering as shown in Figs. 8(a), 8(b) and 8(c). From the experimental results, the improved adaptive sliding mode control is clearly suitable for controlling the LPM motor and effective for reducing the chattering phenomena with good performance.

TABLE III
MEAN ABSOLUTE ERROR COMPARISON OF EXPERIMENTAL RESULTS

Controllers	Case 1 (mm)	Case 2 (mm)
Total Sliding Mode	0.22	1.72
Adaptive Sliding Mode	0.25	1.00
Improved Adaptive Sliding Mode	0.24	0.96

VI. CONCLUSIONS

This paper presents the improved adaptive sliding mode control that is successfully implemented to control the LPM motor with dSPACE (DS1104) controller card. The simulation and experimental results are conducted for two different cases and then evaluate the performance criterion with Mean Absolute Error (MAE). Although the performance of the improved adaptive sliding mode controller is similar to the adaptive sliding mode controller, the improved adaptive sliding mode control are less chattering phenomena than the total sliding mode control and the adaptive sliding mode control. Finally, the idea of this proposed can be easily applied to other control systems.

Appendix Figure D6 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page6

VII. ACKNOWLEDGEMENT

This research is financially supported by Thailand Advanced Institute of Science and Technology - Tokyo Institute of Technology (TAIST-Tokyo Tech), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology (Tokyo Tech), and Kasetsart University (KU).

REFERENCES

[1] I. Boldea and S. A. Nasar, *Linear Electric Actuators and Generators,* Cambridge, U.K.: Cambridge University Press, 1997.

[2] W. D. Chou, F. J. Lin and K. K. Shyu, "Incremental motion control of induction motor servo drive via a genetic-algorithm-based sliding mode controller," *Proceedings of IEE Control Theory and Applications,* vol. 150, no. 3, pp. 209–211, May 2003.

[3] J. F. Gieras and Z. J. Piech, *Linear Synchronous Motors: Transportation and Automation Systems,* Boca Raton, FL: CRC Press, 1999.

[4] S. C. Hsu, C. Hao Liu, C. Huan Liu and N. J. Wang,"Fuzzy PI controller tuning for a linear permanent magnet synchronous motor drive," *Proceedings of IEEE Industrial Electronics,* vol. 3, pp. 1661–1666, 2001.

[5] H. K. Khalil, *Nonlinear systems,* New York: Macmillan, 1992.

[6] T. H. Liu, Y. C. Lee and Y. H. Chang, "Adaptive controller design for a linear motor control system," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 40, no. 2, pp. 601–616, April 2004.

[7] J. E. Slotine and W. Li, *Applied Nonlinear Control,* Englewood Cliffs, NJ: Prentice-Hall, 1991.

[8] K. K. Shyu, J. Y. Hung and J. C. Hung, "Total sliding mode trajectory control of robotic manipulators," *Proceedings of IEEE Industrial Electronics,* vol. 3, pp. 1062–1066, 1999.

[9] V. I. Utkin, J. Guldner and J. Shi, *Sliding Mode Control in Electromechanical Systems,* New York: Taylor & Francis, 1999.

[10] R. J. Wai, "Adaptive sliding-mode control for induction servo-motor drive," *Proceedings of IEE Electric Power Applications,* vol. 147, no. 6, pp. 553–561, November 2000.

[11] R. J. Wai, "Total sliding-mode controller for PM synchronous servo motor drive using recurrent fuzzy neural network," *IEEE Transactions on Industrial Electronics,* vol. 48, no. 5, pp. 926–944, October 2001.

Appendix Figure D7 An improved Adaptive Sliding Mode Controller Design for Linear Permanent Magnet Motor page7

Appendix Figure D8 Certificate of best presentation of an improved adaptive sliding mode controller design for linear permanent magnet motor

Appendix Figure D9 Certificate of best student-paper of an improved adaptive sliding mode controller design for linear permanent magnet motor

Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization

Tithiwat Therdbankerd*, Peerayot Sanposh*, Nattapon Chayopitak† and Hideaki Fujita‡
* Intelligent Mechatronics, Automation, Robotics, and Control Laboratory,
Department of Electrical Engineering, Kasetsart University
50 Phahonyothin Rd., Ladyao, Jatujak, Bangkok, 10903, Thailand
Email: tithiwat@msn.com, peerayot.s@ku.ac.th

† Industrial Control and Automation Laboratory, National Electronics and Computer Technology Center
112 Thailand Science Park, Phahonyothin Rd., Klong 1, Klong Luang, Pathumthani, 12120, Thailand
Email: nattapon.chayopitak@nectec.or.th

‡ Department of Electrical and Electronic Engineering, Tokyo Institute of Technology
S3-18, 2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8550, Japan
Email: hide@ee.titech.ac.jp

***Abstract*—Accurate and effective parameter identification is an important engineering task in high performance control system design. One emerging approach to effectively identify such nonlinear or dynamic unknown parameters is to use Particle Swarm Optimization (PSO) algorithm. Linear Permanent Magnet (LPM) motor is a high performance actuator employed in many applications that require direct linear motion without mechanical transmission for high acceleration and accurate positioning. Therefore, accurate motor parameters are necessary to effectively control the LPM motors. This paper proposes a simple PSO based method with chirp inputs to identify the LPM motor's parameters. The simulations and experiments are conducted to verify the results and determine the effectiveness of the proposed method.**

***Index Terms*—Particle Swarm Optimization, Parameter Identification, Linear Permanent Magnet Motor**

I. Introduction

Linear motors are electric motors that can operate without conventional gears, screws or crack shafts to provide direct linear motion. Linear Permanent Magnet (LPM) motors, particularly, have been widely used in many applications that require faster speed and better accuracy such as computer-controlled machine tools, semiconductor manufacturing equipments and inspection machines [1], [3].

In general, accurate parameters are significant for high-performance control system designs. In order to acquire the accurate parameters of the LPM motors, several parameter identification techniques have been investigated for different system. In particular, the use of Genetic Algorithm to identify the parameters of Permanent Magnet Synchronous Motor (PMSM) is reported in [11]. In [7], Particle Swarm Optimization (PSO) has been proposed to directly identify the parameters of PMSM.

Although, PMSM and LPM motors share several magnetic features and many design and control concepts are transferable, unfortunately this is not the case for parameter identification techniques. The accuracy of parameter identification techniques rely on the appropriate choice of the test input signals [8], such as the well-known pseudo random binary signal (PRBS), that can excite as many modes for extracting as many features of the system. Several studies, for instance [4], [9], [10], have reported successful application of PRBS inputs for PMSM.

However, PRBS inputs are not suitable for parameter identification of linear motors with finite traveling track length. To be effective, PRBS is generated in random sequence for extended excitation time resulting in long travel distance; this problem is not encountered for PMSM systems since the motor essentially has unlimited travel distance, in other words infinite rotation. Since, linear motor drive systems found in manufacturing applications usually have limited-length traveling tracks (e.g. 0.5-2 m), PRBS is essentially not suitable for parameter identification of the LPM motor system as the input command for extended travel distance may cause the mover to hit the end of the track resulting in physical damages.

Literature on parameter identification for linear motor is limited. A parameter auto-tuning technique for identifying the parameters of the LPM motor is proposed in [5], but the method is complicated to implement. This paper proposes a parameter identification method based on PSO using chirp signal and simple procedure of placing two different payloads to the system to effectively identify the mechanical parameters of the LPM motor. The parameter identification method based on PSO has advantages of fast convergence and simplicity when appropriate test input signal is chosen. The proposed method may be used with any systems with any input signals, but the chirp signal is recommended for linear motor systems. The simulations and experiments have been conducted to demonstrate the effectiveness of the proposed method.

II. Description of Linear Permanent Magnet Motor

The mathematical model of the LPM motor, as shown in Fig. 1, in the dq-axis reference frame is given by [1]

Appendix Figure D10 Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization page1

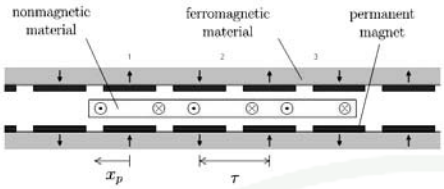


Fig. 1. The LPM motor structure.

$$\dot{i}_d = -\frac{R_s}{L_d} i_d + \frac{\pi}{\tau}\frac{L_q}{L_d}\dot{x}_p i_q + \frac{1}{L_d} v_d \quad (1)$$

$$\dot{i}_q = -\frac{L_d}{L_q}\dot{x}_p i_d - \frac{R_s}{L_q} i_q - \frac{\pi}{\tau}\frac{\lambda_f}{L_q}\dot{x}_p + \frac{1}{L_q} v_q \quad (2)$$

$$F_e = \frac{3}{2}\frac{\pi}{\tau} p \lambda_f \triangleq K_f i_q \quad (3)$$

where x_p and $\dot{x}_p$ are the position and velocity of the LPM motor; p is the number of pole pairs; τ is the pole pitch; F_e is the motor force; v_d and v_q are the voltages in the dq-axis reference frame; i_d and i_q are the currents in the dq-axis reference frame; R_s, L_d, L_q and λ_f are the phase winding resistance, the d- and q-axis inductances, and the permanent magnet flux, respectively; K_f is the motor force constant.

The mechanical system of the LPM motor is composed of the mover and its payload described by

$$F_e = M\ddot{x}_p + B\dot{x}_p + F_L \quad (4)$$

where F_L is the external force; M is the total mass of the moving part; B is the viscous friction coefficient; and $\ddot{x}_p$ is the acceleration.

Since the force produced by the LPM motor only depends on i_q, it is customary to control $i_d = 0$ [3], resulting in i_q chosen as the input of the mechanical system. Hence, (3) and (4) can be rewritten into

$$K_f i_q = M\ddot{x}_p + B\dot{x}_p + F_L \quad (5)$$

Normally, parameters are provided by motor manufacturers but they are usually not sufficient for high-performance applications, due to parameter variations and various disturbances. Some parameters often fluctuate during operation and are difficult to measure in practice. For example, the motor force constant, K_f, the viscous friction coefficient, B, and the external force, F_L, cannot be directly measured whereas M cannot be easily determined if the LPM motor is already installed in a larger system such as a gantry robot. Therefore, this paper proposes the PSO method to effectively identify such mechanical parameters: K_f, M, B, and F_L in (5).

III. Particle Swarm Optimization

In 1995 Kennedy and Eberhart first introduced PSO, a stochastic optimization algorithm that imitates the process of animals' group communication behavior [6]. The information sharing among the group's population, known as a swarm, helps the group determine a "better" solution quickly whenever a member of the swarm, known as a particle, "discovers" a better route to the target. The PSO algorithm can be described by the updating rules of each particle [2] as

$$\begin{aligned} v_i^{k+1} &= w v_i^k + c_1\gamma_1 \times (p_{\text{best}_i} - x_i^k) \\ &\quad + c_2\gamma_2 \times (g_{\text{best}} - x_i^k) \\ x_i^{k+1} &= x_i^k + v_i^{k+1} \end{aligned} \quad (6)$$

where c_1 and c_2 are positive constants, defined as the cognitive coefficients; w is the inertia weight factor; γ_1 and γ_2 are random numbers in $[0, 1]$; x_i and v_i are the position and velocity of i^{th} particle; p_{best_i} is the personal best position of particle; g_{best} is the global best position of the entire population; $k = 1, 2, \ldots$ indicates the number of iteration.

The flowchart of the PSO algorithm is shown in Fig. 2 and the details of each step are as follows:

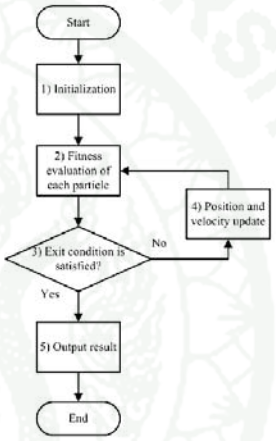


Fig. 2. The PSO algorithm flowchart.

1) *Initialization*. In this step, all parameters to be optimized are initialized. The position and velocity of each particle are randomly assigned in the search space; g_{best} and p_{best_i} are set at the allowable maximum value.
2) *Fitness evaluation of each particle*. For each step k, the fitness or performance of each particle is calculated. If the new calculated value is better than the current p_{best_i}, then new p_{best_i} is updated by the new calculated value; if the best of p_{best_i} is better than the current g_{best}, then g_{best} is replaced by the new best of p_{best_i}.
3) *Exit condition*. If the current g_{best} satisfies the desired exit condition, i.e. the g_{best} is lower than the minimum error criterion or either the algorithm reaches the maximum iteration, then go to step 5. If not, go to step 4.
4) *Position and velocity update*. The position and the velocity of each particle are updated according to (6).

Appendix Figure D11 Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization page2

If the updated position and velocity particles are ended up outside the search space, their values are set at the boundaries [2], i.e. at the maximum or minimum value.

5) *Output result*. The best solution of the optimization process, g_{best}, is the output of this step.

IV. PSO Approach for LPM motor Parameter Identification

The method investigated here follows the approach proposed by [7]. Consider a general dynamical system that represents the "real system" (e.g. the LPM motor)

$$\dot{x} = f(p, x, u) \quad (7)$$
$$y = g(p, x) \quad (8)$$

where x is the vector of state variables; u is the vector of system inputs; p is the vector of unknown parameters to be identified; and y is the vector of output variables. Next consider an estimate model that is used to predict the behavior of the "real system"

$$\dot{\hat{x}} = f(\hat{p}, \hat{x}, u) \quad (9)$$
$$\hat{y} = g(\hat{p}, \hat{x}) \quad (10)$$

where $\hat{x}$ is the vector of state variables of the estimated model; $\hat{y}$ is the vector of output variables of the estimated model; and $\hat{p}$ is the vector of the estimated unknown parameters; Note that u is the same for (7) and (9) while $f(\cdot)$ and $g(\cdot)$ also represent the same system structure. The fitness function for evaluation should be carefully selected.

The block diagram of the PSO identification method is shown in Fig. 3. First, an input signal, u, is fed into both real and estimated systems to obtain the outputs of position and velocity. Then, evaluate these outputs by the fitness function for optimization. It is suggested in [2] that the fitness function should be formulated as the sum of squares when high precision result is desired, such as

$$C(\hat{p}) = \int_0^t (y - \hat{y})^T W (y - \hat{y}) dt \quad (11)$$

where W is a positive definite matrix. The fitness function, $C(\hat{p})$, depends on $\hat{p}$ to identify the parameter vector, p. Hence, the optimal $\hat{p}$ is determined when the corresponding value of the fitness function trends to zero, i.e. $C(\hat{p} \approx p) \to 0$. From (5), (7) and (8), pick $y = (x_p, \dot{x}_p)$ and $u = i_q$ to rearrange the system dynamic as a double integrator, then the fitness function becomes

$$C(\hat{p}) = \sum_{k=1}^{n} \left(w_{x_p}(x_p(k) - \hat{x}_p(k))^2 + w_{\dot{x}_p}(\dot{x}_p(k) - \dot{\hat{x}}_p(k))^2 \right) \quad (12)$$

where x_p and $\hat{x}_p$ are the position responses of the real system and the estimated model; $\dot{x}_p$ and $\dot{\hat{x}}_p$ are the velocity responses of the real system and the estimated model; w_{x_p} and $w_{\dot{x}_p}$ are the weighting factors; and n is the number of the sampling data.

This concept can be used for offline or online identification. For offline identification, the exit condition is reached when the estimated parameter vector approaches a target criterion ($\hat{p} \approx p$). For online identification, the exit condition could be when $\hat{p} \approx p$ or when the maximum iteration number is reached. In this paper, only offline identification is implemented to identify the mechanical parameters of the LPM motor.

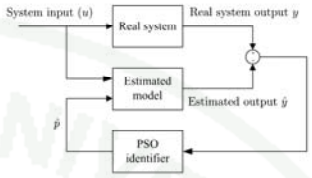


Fig. 3. Block diagram of the PSO identification method.

V. Simulation and Experiment

A. Implementation

For the PSO identification method, the accuracy of the estimated parameters depends on the appropriate system input that should sufficiently excite the LPM motor model. Consider (5) when no payload is attached to the system so that $M = \bar{M}$; then

$$i_q = \frac{\bar{M}}{K_f}\ddot{x}_p + \frac{B}{K_f}\dot{x}_p + \frac{F_L}{K_f} \triangleq p_1\ddot{x}_p + p_2\dot{x}_p + p_3 F_L \quad (13)$$

where p_1, p_2 and p_3 are the motor parameters without payload; and $\bar{M}$ is the nominal value of the moving mass without payload. Next consider when a payload is attached to the original system so that $M \to M' = \bar{M} + \tilde{M}$

$$i_q = \frac{M'}{K_f}\ddot{x}_p + \frac{B}{K_f}\dot{x}_p + \frac{F_L}{K_f} \triangleq p_1'\ddot{x}_p + p_2'\dot{x}_p + p_3' \quad (14)$$

where p_1', p_2' and p_3' are the motor constant with additional mass; and $\tilde{M}$ is a known adding mass of the payload. Note that the motor force constant K_f does not depend on the payload hence it remains unchanged. Then each parameter may be calculated as follows:

$$p_1' = \frac{M'}{K_f} \Rightarrow K_f = \frac{\tilde{M}}{p_1' - p_1} \quad (15)$$

Once K_f is known, B and F_L can be determined accordingly from (13), i.e.

$$B = p_2 K_f \ , \quad F_L = p_3 K_f \quad (16)$$

B. Simulation Results

The simulations are implemented using MATLAB & SIMULINK. All simulations are tested by adjusting two

Appendix Figure D12 Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization page3

weighting factors between $[0, 1]$. Each simulation run is performed for three cases:

$$\begin{aligned} \text{Case 1} &: M = \bar{M} \\ \text{Case 2} &: M = \bar{M} + 1.72 \text{ kg} \\ \text{Case 3} &: M = \bar{M} + 2.75 \text{ kg} \end{aligned} \tag{17}$$

The PSO identification method is simulated with the set of measured motor parameters shown in Table I and the PSO configuration parameters are listed in Table II. The chirp signal, used as a test input is a sinusoidal function with varying frequency ω over a time period $0 \le t \le t_f$ whose equation is given by

$$u(t) = A\cos(\frac{\omega_1 t + (\omega_2 - \omega_1)t^2}{2t_f}) \tag{18}$$

where A is the amplitude; ω_1 and ω_2 are the lower and upper bound frequencies; and t_f is the final time. Since there is no traveling length limitation in simulation, PRBS [8] is also implemented for comparison with the results from the chirp signal.

TABLE I
LPM MOTOR PARAMETERS

LPM motor parameters	Nominal values
Motor force constant, K_f	10.83 N/A
Nominal moving mass, $\bar{M}$	1.4 kg
Viscous friction coefficient, B	5 Ns/m
External frictional force, F_L	0.05 N

TABLE II
PSO CONFIGURATION PARAMETERS

PSO identification parameters	Values
Population size	20
Maximum iteration	150
Inertia weight factor, w	0.7
Cognitive coefficient1, c_1	1.43
Cognitive coefficient2, c_2	1.43
Random numbers, (γ_1, γ_2)	Randomly chosen between [0,1] for each calculation
Chirp signal parameters	Values
Lower bound frequency, ω_1	0.1 Hz
Upper bound frequency, ω_2	100 Hz
Final time, t_f	20 sec
PRBS parameters	Values
Amplitude	± 1
Frequency	5 Hz

The simulation procedures are conducted as the diagram shown in Fig. 3 as follows:

1) *Initialization*. The parameters listed in Table I and II are initialized.
2) *Simulation*. The input signal of the chirp or PRBS inputs is fed to the real system and estimated model and then the simulation outputs are obtained.
3) *PSO identifier*. The PSO identifier algorithm identifies the motor constant parameters p_1, p_2 and p_3.
4) *Repeat for different values of payload*. Repeat steps 1 - 3 to identify the motor constant parameters for three cases (17).
5) *Identify the unknown parameters*. Calculate K_f, B, F_L from (15) - (16).
6) *Adjust the weighting factors*. The weighting factors are adjusted to explore the system's sensitivity.

The identified results from the simulation are shown in Table III and Table IV. It is apparent that the chirp and PRBS inputs can effectively identify the LPM motor parameters compared to the nominal values listed in Table I. However, as mentioned early, PRBS is not suitable for identifying the LPM motor's parameters; only chirp inputs are used to obtain experimental results.

TABLE III
SIMULATION RESULTS USING CHIRP SIGNAL INPUT

No.	(w_{z_p}, w_{z_p})	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	(0.0, 1.0)	10.830	1.400	5.000	0.050
2	(0.1, 0.9)	10.830	1.400	5.000	0.050
3	(0.2, 0.8)	10.234	1.332	3.486	0.104
4	(0.3, 0.7)	10.606	1.383	4.717	0.000
5	(0.4, 0.6)	10.897	1.390	4.921	0.339
6	(0.5, 0.5)	11.014	1.423	5.261	0.045
7	(0.6, 0.4)	10.786	1.403	4.981	0.000
8	(0.7, 0.3)	10.119	1.340	3.931	0.047
9	(0.8, 0.2)	10.682	1.380	4.865	0.000
10	(0.9, 0.1)	11.007	1.427	5.206	0.000
11	(1.0, 0.0)	10.957	1.413	5.130	0.000

TABLE IV
SIMULATION RESULTS USING PRBS SIGNAL INPUT

No.	(w_{z_p}, w_{z_p})	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	(0.0, 1.0)	10.830	1.400	5.000	0.050
2	(0.1, 0.9)	10.828	1.399	4.993	0.066
3	(0.2, 0.8)	10.819	1.398	5.013	0.050
4	(0.3, 0.7)	10.828	1.399	4.985	0.089
5	(0.4, 0.6)	10.724	1.363	4.945	0.126
6	(0.5, 0.5)	10.827	1.399	4.974	0.161
7	(0.6, 0.4)	10.813	1.394	4.983	0.076
8	(0.7, 0.3)	10.821	1.398	5.002	0.050
9	(0.8, 0.2)	10.816	1.399	5.003	0.050
10	(0.9, 0.1)	10.814	1.394	4.929	0.221
11	(1.0, 0.0)	10.827	1.399	4.996	0.058

C. Experimental Results

Fig. 4. Experiment setup.

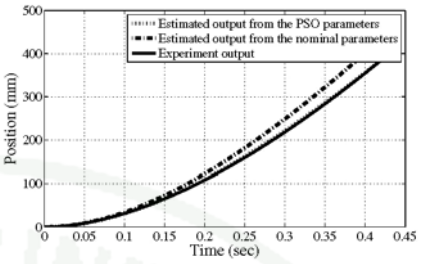
The PSO parameter identification method is implemented on a PC for the experiment setup, shown in Fig. 4, where the LPM motor is controlled in a force/current control mode. The conditions for the experiments are the same as those for the simulations with the same set of chirp inputs. The actual

Appendix Figure D13 Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization page4

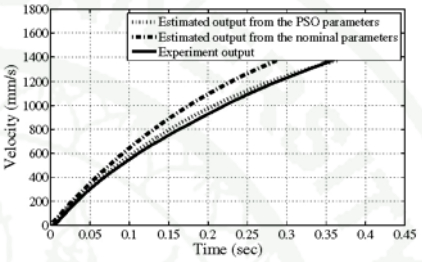
outputs (position and velocity) are measured using the optical linear encoder. The results of the identified parameters are summarized in Table V. It is noticeable that the estimated values from the proposed method are slightly deviated from the nominal values shown in Table I. The plots of the position and velocity responses compared with the simulated results are shown in Fig. 5. However, these deviations from the nominal values do not necessarily indicate the shortcoming of the proposed method since the nominal values themselves are usually not sufficient for high performance controller, especially with different loading conditions or different operating points. Therefore, it is not recommended to compare the identified parameters with the nominal parameters, as also noted by [7], but to directly compare the dynamic performances between the identified parameters and the nominal values. Nonetheless, the results of the present work indicate the effectiveness of the proposed PSO method for parameter identification.

TABLE V
EXPERIMENTAL RESULTS USING CHIRP SIGNAL INPUT

No.	$(w_{x_p}, w_{\dot{x}_p})$	K_f (N/A)	M (kg)	B (Ns/m)	F_L (N)
1	(0.0, 1.0)	10.812	1.538	6.271	0.057
2	(0.1, 0.9)	10.892	1.551	6.328	0.072
3	(0.2, 0.8)	10.845	1.545	6.362	0.061
4	(0.3, 0.7)	10.806	1.540	5.760	0.143
5	(0.4, 0.6)	10.799	1.538	6.520	0.026
6	(0.5, 0.5)	10.820	1.541	6.083	0.116
7	(0.6, 0.4)	10.958	1.560	5.257	0.247
8	(0.7, 0.3)	10.702	1.523	6.299	0.057
9	(0.8, 0.2)	11.261	1.487	2.967	1.432
10	(0.9, 0.1)	10.571	1.499	6.213	0.058
11	(1.0, 0.0)	10.534	1.472	6.503	0.000



(a) Position of the system outputs.



(b) Velocity of the system outputs.

Fig. 5. Comparison between system outputs.

VI. CONCLUSIONS

This paper proposed a PSO identification method to identify the LPM motor parameters. Both simulation and experiment are conducted to validate the effectiveness of this identification method. The results showed that the PSO identification method can be used to identify the mechanical parameters of the LPM motor model with good accuracy. The proposed method can also be applied to identify parameters of other systems.

Possible future works are to identify other parameters and explore different types of input signals, as well as to determine to what extent the identified parameters could enhance the performance of the controller in comparison with using just the nominal parameter values from the datasheet in the controller.

VII. ACKNOWLEDGEMENT

This research is financially supported by Thailand Advanced Institute of Science and Technology - Tokyo Institute of Technology (TAIST-Tokyo Tech), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology (Tokyo Tech), and Kasetsart University (KU).

REFERENCES

[1] I. Boldea and S. A. Nasar, *Linear Electric Actuators and Generators*, Cambridge, U.K.: Cambridge University Press, 1997.
[2] M. Clerc, *Particle Swarm Optimization*, London, U.K.: ISTE Publishing Company, 2006.
[3] J. F. Gieras and Z. J. Piech, *Linear Synchronous Motors: Transportation and Automation Systems*, Boca Raton, FL: CRC Press, 1999.
[4] M. Hasni, O. Touhami, R. Ibtiouen, M. Fadel and S. Caux, "Synchronous machine parameter identification by various excitation signals," *Journal of Electrical Engineering*, vol. 90, no. 3, pp. 219-228, 2008.
[5] S. C. Hsu, W. D. Liu and C. H. Liu, "Parameter auto-tuning for a linear permanent magnet synchronous motor drive," *Proceeding of IEEE on Industrial Electronics Society*, Nagoya, Japan, pp. 1031–1036, October 2000.
[6] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceeding of IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1948, November 1995.
[7] L. Liu, W. Liu and D. A. Cartes, "Permanent magnet synchronous motor parameter identification using particle swarm optimization," *International Journal of Computational Intelligence Research*, vol. 4, no. 2, pp. 211-218, 2008.
[8] L. Ljung, *System Identification: Theory for the User*, Englewood Cliffs, NJ: Prentice-hall, 1987.
[9] W. Michalik, "Standstill estimation of electrical parameters in motors with optimal input signals," *Proceeding of IEEE International Caracas Conference on Devices, Circuits and Systems*, Margarita, Venezuela, pp. 407–413, March 1998.
[10] M. Pacas, S. Vollwock, P. Szczupak and H. Zoubek, "Methods for commissioning and identification in drives," *International Journal of Computation and Mathematics in Electrical and Electronic Engineering*, vol. 29, no. 1, pp. 53–71, 2010.
[11] S. D. Sudhoff, J. Cale, B. Cassimere and M. Swinney, "Genetic algorithm based design of a permanent magnet synchronous machine," *Proceeding of IEEE on Electric Machines and Drives*, Texas, U.S., pp. 1011–1019, May 2005.

Appendix Figure D14 Parameter Identification of a Linear Permanent Magnet Motor Using Particle Swarm Optimization page5

CIRRICULUM VITAE

NAME : Mr. Tithiwat Therdbankerd

BIRTH DATE : June 6, 1984

BIRTH PLACE : Bangkok, Thailand

EDUCATION :

YEAR	**INSTITUTE**	**DEGREE/DIPLOMA**
2006	Kasetsart University	B.Eng. (Electrical Engineering)
2010	Kasetsart University	M.Eng. (Information and Communication Technology for Embedded Systems)

POSITION/TITLE : -

WORK PLACE : -

SCHOLARSHIP/AWARDS : Full scholarship of Master degree from Thailand Advanced Institute of Science and Technology – Tokyo Institute of Technology, TAIST-Tokyo Tech (2008-2010)

Certificate of Best Presentation from International Conference on Information and Communication Technology for Embedded System (IC-ICTES2010)

Certificate of Best Student-Paper from International Conference on Information and Communication Technology for Embedded System (IC-ICTES2010)

Teacher Assistant scholarship of Kasetsart University (2009 – 2010)