CEAI, Vol.21, No. 1, pp. 51-60, 2019 Printed in Romania

# Identification and Robust Control of Heart Rate During Treadmill Exercise at Large Speed Ranges

**Ali Esmaeili** [*] **, Asier Ibeas** [*,**]**, Jorge Herrera** [**] **and Nazila Esmaeili** [***]

[*] *Department of Telecommunications and Systems Engineering*
*Universitat Autònoma de Barcelona, 08193 Bellaterra, Barcelona, Spain.*
*Email: Ali.Esmaeili@e-campus.uab.cat*

[**] *Dep. de Ingeniería, Facultad de Ciencias Naturales e Ingeniería ,*
*Universidad de Bogotá Jorge Tadeo Lozano, 110311, Bogotá DC, Colombia*

[***] *Faculty of Electrical Engineering and Information Technology,*
*Otto-von-Guericke-Universitat, 39106, Magdeburg, Germany*

**Abstract:** The objective of this paper is to design a heart rate (HR) controller for a treadmill so that the HR of an individual running on it tracks a pre-specified, potentially time-varying profile specified by doctors for the cardiac recovery of the person. Initially, a parameter estimation algorithm is presented with the aim of estimating the values of the parameters of a model relating the speed of the treadmill with the HR of an individual. The parameter estimation problem is formulated as an optimization one and solved by using Particle Swarm Optimization (PSO). Afterwards, a super-twisting sliding mode controller is designed to perform the robust control of treadmill's speed in the presence of potential unmodelled dynamics or parametric uncertainties. Numerical examples show that the estimation procedure is able to obtain accurate values for the system's parameters while the proposed control approach is able to obtain zero tracking error without chattering, definitely achieving the control objectives. In both cases the range of treadmill's speed goes from 2 to 14 km/h, range that is not usually employed in previous studies.

*Keywords:* Heart rate control, PSO Identification of nonlinear systems, Super-twisting sliding mode control

## 1. INTRODUCTION

The objective of this paper is to design a heart rate (HR) controller for a treadmill so that the HR of an individual running on it tracks a pre-specified, potentially time-varying profile specified by doctors for the cardiac recovery of the person. The controller modifies the speed of the treadmill so as to make individual's HR track such a profile. The ability to control the Heart Rate (HR) in treadmill exercise is of the great importance in design of exercise routines and it is one of the most significant vital signs to reflect cardiovascular events, (Hunt et al, 2016). The understanding of HR response with exercise may also lead to an improvement in developing training protocols for athletics, more efficient weight loss protocols for the overweight people, and in facilitating assessment of physical fitness and health of individuals, (Anselmio et al, 2017), (Argha et al, 2017), (Weippert et al, 2014). In addition, treadmill exercise has an important role for people recovering from cardiac disease or surgery as well as for people involved in weight loss programs, (Dan and Dragumir, 2012). The design of the controller follows two steps. The first one is the modeling of the HR response to the treadmill exercise while the second one designs the controller based on the previous model by means of a sliding mode control.

Several models have been proposed in the literature to model heart rate response to treadmill velocity, (Jang and Dae-Geun, 2016), (Cheng et al, 2008), (Peter and Huber, 1964). For instance, (Shtessel and Yuri, 2010) proposes a Hammerstein model composed of a static non-linearity defined by a look-up table followed by a linear dynamical system, while (Zhang et al, 2011) and (Scalzi et al, 2012) propose a nonlinear dynamical model. Anyway, nonlinearity must be present in the model due to the nonlinear response of the heart rate to the exercise. In this study, we use the nonlinear dynamical system proposed by, (Zhang et al, 2011), (Wang et al, 2007), (Rini et al, 2014), which is able to capture the dynamic behavior in a compact way by means of a reduced number of parameters, fact that is convenient for control purposes.

Heart rate treadmill models are parametrized by a number of parameters that capture the individual HR response to exercise, (Kranjec et al, 2014), (Mayer et al, 2018). It is vital, thus, to design parameter estimation procedures that allow us to have a personalized model for each individual from measured data. In this work, parameter estimation is set up as an optimization problem whose solution leads to the estimated model parameters. The optimization problem is solved by using a Particle Swarm Optimization (PSO) algorithm.

Particle swarm optimization was introduced by Kennedy and Eberhart in 1995, (Yan et al, 2013), (Shtessel and Yuri, 2010). At that time, the wide success of Evolutionary Algorithms (EAs) motivated researchers worldwide to develop and experiment with novel nature-inspired methods. Thus, besides the interest in evolutionary procedures that governed EAs, new paradigms from nature were subjected to investigation. The first PSO models introduced the novelty of using difference vectors among population members to sample new points in the search space, (Souravlias and Parsopoulos, 2016), (Xue et al, 2013), (Sugonthan, 1997). This novelty diverged from the established procedures of EAs, which were mostly based on sampling new points from explicit probability distributions. Additional advantages of PSO were its potential for easy adaptation of operators and procedures to match the specific requirements of a given problem, as well as its inherent decentralized structure that promoted parallelization, (Storn and Price, 1997). PSO has gained much attention nowadays and has wide applications in different fields such a fitness distance ratio, (Yan et al, 2013), adaptive mutation and inertia weight, (Zhang et al, 2011) and parameter identification in magnet synchronous motors, (Liu et al, 2008), hybrid neural network and the level of seismic inversion, (Yang et al,2017). In this study, the parameter estimation problem is solved by using the Particle Swarm Optimization (PSO) method, which is the first time where this method is used for the problem at hand. The parameter estimation will be focused in the range of treadmill speeds from 2 up to 14 km/h (2-14 km/h). In many studies, (Bansal et al, 2011), (Ibeas et al, 2016), (Shtessel and Yuri, 2010), the usual range of speed is (2-8 km/h), or (2-10 km/h). Therefore, the proposed methodology is applicable in the large range of speeds from 2 to 14 km/h. Despite this range is popular in many rehabilitation and training exercises, it is the first time considered in this problem.

On the other hand, a Sliding Mode Controller (SMC) approach is adopted to design the controller. The SMC has revealed very useful in the robust control of multiple systems, such as pneumatic cylinder as actuators for robot manipulators, (Paul et al, 1994) and the hydraulic dynamics of the manipulator, (Guo et al, 2008).However, this is the first time that SMC is used to control the HR during treadmill exercise since previous works used different control techniques such $H_\infty$ robust control approaches or model predictive controllers, (Scalzi et al, 2012), (Shtessel and Yuri, 2017), (GAO and Xuehui, 2016). Especial attention will be devoted to the chattering effect since this is a crucial aspect in biomedical applications, (Lu et al, 2016). To this end, a super-twisting based sliding mode controller will be designed instead of a traditional SMC, (Shtessel and Yuri, 2010), (GAO and Xuehui, 2016). The super-twisting approach will allow avoiding the undesired oscillations that a classical sliding mode controller may cause. Simulation results will show that our approach definitely improves the accuracy of the model parameter estimation for these treadmill speeds, (up to 14 km/h) and the SMC also improves the heart rate control which has not been considered in the past.

The rest of paper is organized as follows. In section (2) we introduce the problem formulation. In this part, we provide the model description. In the next section, the PSO algorithm proposed for the parameter estimation is introduced, while the advantages of this method are commented. In addition, Section (3) also contains the controller design procedure based on SMC. Finally, the last section presents the simulation examples and its comparison with alternative estimation and control methods.

## 2. PROBLEM FORMULATION

The following nonlinear model describes the relationship between speed and heart rate during treadmill exercise, (Esmaeili and Ibeas, 2016), (Jang and Dae-Geun, 2016), (Su et al, 2010):

$$\dot{x}_1(t) = -a_1x_1(t) + a_2x_2(t) + a_3u^2(t) \tag{1}$$

$$\dot{x}_2(t) = -a_4x_2(t) + \phi(x_1(t)) \tag{2}$$

$$\phi(x_1(t)) = \frac{a_5x_1(t)}{1+\exp(-(x_1(t)-a_6))} \tag{3}$$

$$y(t) = x_1(t) + HR_{rest} \tag{4}$$

Where $x(0) = [x_1(t), x_2(t)] = [0,0]$ is the usual initial condition and $a_1$, ..., $a_6$ are positive scalars that are adjusted from real data to describe the particular response of each individual to exercise. The output $y(t)$ relates to the change of HR of the person, and $HR_{rest}$ is the value of the Heart Rate at rest. The control input $u(t)$ describes the speed of the treadmill. The component $x_1(t)$ describes the change of HR from the heart rate at rest mainly due to the central response to exercise, whereas the component $x_2(t)$ describes the slower and more complex local peripheral effects. The positive feedback signal $x_2$, or a dynamic disturbance input to the $x_1$ subsystem, may be treated as a reaction of HR to the effects from the peripheral local responses or factors. In this case, the metabolites from the peripheral local metabolism further accelerate the HR during exercise. For instance, in the case of the peripheral local metabolism, the accumulated metabolic by-products, such as adenosine, K+, H+, lactic acid and other metabolites, cause vasodilatation and hyperemia inactive muscles, (Su SW et al, 2010). Vasodilatation in the active muscles causes a reduction in total peripheral resistance which in turn causes a decrease in mean arterial blood pressure. In order to regulate the blood pressure, the cardiac output needs to be increased, meaning that stroke volume and HR are increased via the baroreceptor reflex, (Zhang et al, 2011).

The nonnegative nonlinear function $\phi(x_1)$ has the property that $\phi(x_1) << 1$ when $x_1$ is small, whereas when $x_1$ is much larger than $a_6$, $\phi(x_1(t))$ approaches the linear function $x_1(t)$. If $x_1$ is small and $a_6$ is large, the variable $x_1$ is multiplied by a small factor(i.e. $\frac{a_4}{1+exp(-x_1(t)-a_6)} \approx 0$ in the second equation of (1), so $x_2$ becomes nearly independent of $x_1$ . If $x_1(0) = x_2(0) = 0$ and the input $u(t)$ is small, the state $x_1(t)$ may not be large enough to make the factor $\frac{a_4}{1+exp(-x_1(t)-a_6)}$ significant, and $x_2(t)$ will remain close to zero. As a result, system (1) can be approximated by the system $x_1(t) = -a_1x_1(t) + a_3u^2(t)$ with $x_2(t) = 0$. On the other hand, if the input $u(t)$ is sufficiently large, the state $x_1(t)$ will be driven to a level that the factor $\frac{a_4}{1+exp(-x_1(t)-a_6)}$ is significant, and $x_2(t)$ is no longer independent of $x_1(t)$.

It is important to bear in mind that the objective of the paper is twofold. On the one hand to propose a parameter estimation method based on PSO, while, on the other hand, to design a SMC controller, both things used for the first time in this problem and also for speeds ranging from 2 to 14 km/h. In this way, the estimation of the model's parameters $X = (x_i) = [a_1 .....a_6]$ is formulated as an optimization problem. Hence, the optimization of a cost function will provide an estimation of the parameters. The PSO algorithm will be used to solve the so-obtained optimization problem.

## 3. PARAMETER ESTIMATION AND CONTROLLER DESIGN

This section contains the description of the PSO algorithm along with the derivation of the sliding mode controller.

*3.1 Parameter estimation of the model*

PSO algorithm was utilized for understanding the regulations dominating the swarms of birds and their sudden changes. PSO consists of a population (or swarm) of $M$ particles, each of which represents a $n$ dimensional potential solution of the optimization problem. In our approach, $n = 6$ is the number of parameters to be estimated. Particles are assigned random initial positions and they change their positions iteratively to reach the global optimal solution. It is desired to minimize the fitness function as the PSO iterations progress.

The parameter estimation problem is cast into an optimization one so that the minimum of the fitness cost function will provide an estimation of the parameters of the system. The Squared Error Loss (SEL) is the most common cost function to be optimized for speed estimation problems and it is also the easiest to work with from a mathematical point of view. The SEL is linked with variance and bias of an estimator, so that the cost function is formulated for the estimated parameter vector $\hat{X}_k$ at iteration step $k$ as :

$$I_k = Var(\hat{y}_k) + bias(\hat{X}_k) \tag{5}$$

where $\hat{y}_k$ is the estimated output. Both terms in Eq. (2) are nonnegative i.e. $(Var(\hat{y}_k) > 0, bias(\hat{X}_k) \geq 0)$, so that the minimum of the cost function $I_k$ is given by $I_k = 0$, $argminI_k = (0,0)$. Therefore, when the cost function $I_k$ vanishes then $(Var\hat{y}_k = 0)$ and $bias(\hat{X}_k) = 0$ implying that the estimation of the parameters is performed adequately. The minimization of such cost function is done by using the PSO algorithm.

Each particle evaluates its fitness (given by Eq. (5)) and every particle $i = [1, ..., M]$ has a memory to store the value of its best own position $Pbest\ _{id}$, which is defined as the position where the particle has minimum fitness. Besides, the best of $Pbest\ _{id}$ of all particles, called $Gbest_d$, is stored too. At each iteration $k$, the PSO modifies each dimension of the position $x_{id}$ in a particle by adding a velocity $v_{id}$ and moves the particle towards the linear combination of $Pbest\ _{id}$ and $Gbest_d$ according to:

$$v_{id}(k+1) = w\ v_{id}(k) + c_1\ rand_1(p_{id} - x_{id})\ + c_2\ rand_2(p_{gd} - x_{id}) \tag{6}$$

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k+1) \tag{7}$$

In fact, according to Eqs. (6)-(7), some new particles may be out of the search space so that a projection to the boundaries of such space is included in the algorithm, (Yan et al, 2013). Moreover, the most common approach to restrict the particle position in the search space is to set the violated components of the particle equal to the value of the violated boundary. In the problem at hand, the constraint violation appears when the algorithm provides a negative value for the parameters. Consequently, the projection algorithm takes the form:

$$x_{id} = \begin{cases} 0 \quad , & x_{id} < 0 \\ x_{id}, & otherwise \end{cases} \tag{8}$$

In this way, we can guarantee that the estimated parameters are nonnegative and the velocity and position of each particle are updated by the equations until a termination condition is met and the algorithm finally stops. The parameters $c_1$ and $c_2$ are the so-called cognitive and social parameters, respectively, and satisfy $0 < c_1, c_2 < 1$, (Rini et al, 2014). Finally, $w$ is the inertia weight selected as $0.4 \leq w \leq 0.9$, the range where the algorithm provides the best results, (Bansal et al, 2011).

Typically, the number of subsequent iterations without improvement of the best solution and/or the dispersion of the particles current (or best) positions in the search space has been used as indicators of search stagnation. Frequently, the aforementioned termination criteria are combined in forms such as:

$$IF \quad (|I_{k+1} - I_k| \leq \varepsilon) \quad OR \quad (k \geq k_{max}).\ Then\ Stop \tag{9}$$

where $I_k$ is is the function to be optimized and $k$ stands for the iteration number, respectively, and $\varepsilon$ is the corresponding user-defined tolerance. However, the search stagnation criterion can prematurely stop the algorithm even if the computational budget is not exceeded. Successful application of this criterion is based on the existence of a proper stagnation measure.

Figure 1 displays the pseudocode of the PSO algorithm developed in our study. The PSO search is carried out by the speed

FOR each particle $i$
    Initialize position $x_{id}$
    Initialize velocity $v_{id}$
  End FOR
End FOR
Iteration $k$=1
DO
  FOR each particle i
  Calculate fitness value
  IF fitness value is better than $p\_best_{id}$ in history set current fitness value as the $p\_best_{id}$
    END IF
  End FOR
Choose the particle having the best fitness value as the $g\_best_d$
FOR each particle $i$
  Calculate the velocity according to the equation
    $v_{id}(k+1)= w\ v_{id}(k)+c_1\ rand_1(p_{id} - x_{id})+c_2\ rand_2(p_g - x_{id})$
    update particle position according to the equation
    $x_{id}(k+1)=x_{id}(k)+v_{id}(k+1)$
  IF $x_{id}=\begin{cases} 0, & x_{id} < 0 \\ x_{id} & Otherwise \end{cases}$ End IF
End FOR
IF $[I_{k+1} - I_k] \leq \varepsilon\ OR\ (k \geq k_{max})\ ThenStop$
$k$=$k+1$ while maximum iteration or minimum error are not acceded

Fig. 1. PSO pseudocode employed to solve the optimization problem.

of the particle. During the development of several generations, only the most optimistic particles can transmit information to the other ones. One of the advantages of the PSO method is that it can be applied to optimization problems of large dimensions, often producing quality solutions more rapidly than alternative methods, (Yang et al, 2017), (Eswaran et al, 2017), (Hamed et al, 2017). Also, the algorithm is terminated after a given number of iterations, or once the fitness values of the particles (or the particles themselves) are close enough in some sense.

In the results section (Section 4), we describe the M-estimator as an alternative model to compare with our estimation results, showing that the proposed PSO method outperforms the M-estimator. Once the parameter estimation procedure has been described, the next step is to design a controller to make the heart rate track a pre-specified profile. The design of such a controller is carried out in the following section.

### *3.2 Super-twisting sliding mode control*

The design of control strategies for nonlinear systems has attracted considerable research interest in the recent past, (Shtessel and Yuri, 2017), (Belkaid et al, 2016). Sliding mode control (SMC), as an effective robust control scheme, has been successfully applied to a wide variety of systems, (Gao and Xuehui, 2016), (Abu-Rmileh et al, 2010), (Ebrahimi et al, 2018). This section contains the design of the sliding mode control for the system (1). Thus, define the tracking error as:

$$e = R - y \tag{10}$$

where $R$ denotes the reference signal (that is, the HR profile to be tracked) and $y$ is the output of our system. The role of the controller is to ensure that system's output accurately tracks the reference signal $R$. When the system is perturbed or uncertain, the finite time stabilization is not ensured, (Vaidynathan and Sundaropandian, 2017), (Swikir et al, 2016). Hence, a reaching law based discontinuous control is developed which rejects the uncertainties of the system and ensures that the control objectives are fulfilled. The uncertainties in our system can be modelled as:

$$\begin{aligned}
\dot{x}_1(t) &= -a_1x_1(t) + a_2x_2(t) + a_3u(t)^2 + f_{uncer1}(x)\\
\dot{x}_2(t) &= -a_4x_2(t) + \phi(x_1(t)) + f_{uncer2}(x)\\
\phi(x_1(t)) &= \frac{a_5x_1(t)}{1+\exp\left(-(x_1(t)-a_6)\right)}\\
y(t) &= x_1(t) + HR_{rest}
\end{aligned} \tag{11}$$

where $f_{uncer1}(x)$ and $f_{uncer2}(x)$ account for the unmodelled dynamics and parametric uncertainty in each of the model equations. On the other hand we need to consider the following assumptions.

*Assumption 1.* $f_{uncer1}(x)$ and $f_{uncer2}(x)$ are upper-bounded .
*Assumption 2.* One upper-bound for each one of these terms is known.

These are common assumptions in SMC, (Shtessel and Yuri, 2010). The sliding mode controller is composed of two parts:

$$u = u_{equiv} - u_{sliding} \tag{12}$$

where $u_{equiv}$ is the so-called equivalent control used to remove certain terms in (11)-(12) while the sliding term $u_{sliding}$ is the term used to counteract the uncertainties of the system and will be of the super-twisting type, (Shtessel and Yuri, 2017). This approach will also help us avoid the chattering effect, that would be very harmful in the control system. Initially, the equivalent control will be derived while the final control law will be obtained by incorporating the super-twisting sliding term according to (12).

The following sliding manifold with the integral term is proposed:

$$S(t) = e(t) + \lambda\int_0^t e(\tau)d\tau \tag{13}$$

where $\lambda$ are strictly positive constant. The equivalent control is obtained by derivating (13) with respect to time and then equating the so-obtained derivative to zero. In this way we have:

$$e(t) = R(t) - y(t) = R(t) - x_1(t) + HR_{rest} \tag{14}$$

$$\begin{aligned}
\dot{e}(t) &= \dot{R}(t) - \dot{x}_1(t) + H\dot{R}_{rest} && (15)\\
&= \dot{R}(t) + a_1x_1(t) - a_2x_2(t) - a_3u^2(t) + f_{uncer1}(t) && (16)
\end{aligned}$$

In this way, if we substitute the above expressions into (10) and simplify we obtain. Now, the derivative of the sliding manifold reads:

$$\begin{aligned}
\dot{S}(t) &= \dot{e}(t) + \lambda e(t) && (17)\\
&= \dot{R} - \dot{y} + \lambda e(t) = \dot{R} - \dot{x}_1 + \lambda e(t) && (18)\\
&= \dot{R} - (-a_1x_1 + a_2x_2 + a_3u^2 + f_{uncer1}) + \lambda e(t) && (19)
\end{aligned}$$

If $\dot{S}(t) = 0$ we have:

$$\dot{R} + a_1x_1 - a_2x_2 - a_3u^2 - f_{uncer1} + \lambda e(t) = 0 \tag{20}$$

Now, if we isolate $u^2$ we obtain:

$$a_3u^2 = \dot{R}(t) + a_1x_1 - a_2x_2 - f_{uncer1} + \lambda e(t) \tag{21}$$

$$u^2(t) = \frac{1}{a_3}(\dot{R}(t) + a_1x_1 - a_2x_2) + \lambda e(t) \tag{22}$$

The uncertain terms $f_{uncer1}$ do not appear in (22) since they are unknown. Therefore, they do not appear in the equivalent control part. The super-twisting sliding term is given by, (Swikir, 2016):

$$u_{sliding} = K|S|^{\alpha}sign(S) \tag{23}$$

It is important to point out that the total control command is given by (15) while being composed of the sum of (25) plus (26). Therefore, the value of both state variables $x_1$ and $x_2$ is needed to calculate the control law. The heart rate $x_1$ can be measured easily, as there exist multiple devices to measure the HR of an individual in real time. However, the peripherical effects $x_2$ cannot be measured. As a consequence, a state observer is needed in order to implement the control command in practice. The state observer is given by:

$$\dot{\hat{x}}_2(t) = -a_4\hat{x}_2(t) + \phi(x_1(t)) \tag{24}$$

With arbitrary initial condition $\hat{x}_2(0)$, since $x_2$ is infusible to obtain and $f_{uncer1}$ is unknown. The control law reads:

$$u(t) = \frac{1}{a_3}\left(\dot{R}(t) + a_1x_1(t) - a_2\hat{x}_2(t) + \lambda e(t)\right) - K|S|^{\alpha}sign(S) \tag{25}$$

Despite the observer works with arbitrary initial conditions, a judicious choice is given by $\hat{x}_2(0) = 0$ since at the beginning for the exercise, the peripherical effects are small and the initial value of the state variable is close to zero. In this way, the initial observation error would be zero and will maintain close to zero during all the observation.

*Assumption 3.* The observation error at the initial time is bounded and an upper-bound for it is known.

The switching gain $K$ has to be selected so as to guarantee the stability and reference tracking of the closed-loop system. In order to obtain a guideline for its tuning we consider the following Lyapunov function candidate:

$$V(t) = \frac{1}{2}S^2 \tag{26}$$

Its time-derivative is given by:

$$\begin{aligned}\dot{V}(t) &= S\dot{S} = S\left(\dot{R} - \dot{x}_1(t) + \lambda e(t)\right)\\ &= S\left(a_2\hat{x}_2(t) - a_2x_2(t) - Ka_3|S|^{\alpha}sign(S) - f_{uncer1}(x)\right)\\ &= S\left(a_2\tilde{x}_2(t) - Ka_3|S|^{\alpha}sign(S) - f_{uncer1}(x)\right)\\ &= -Ka_3|S|^{\alpha+1} + S\left(a_2\tilde{x}_2(t) - f_{uncer1}(x)\right) \qquad (27)\end{aligned}$$

where $\tilde{x}_2(t) = \hat{x}_2(t) - x_2(t)$, represents the observation error. In order to ensure the appropriate operation of the controller, the time derivative (27) should be negative-definite, fact that is achieved if:

$$Ka_3 > a_2|\tilde{x}_2(t) - f_{uncer1}(x)| \qquad (28)$$

Condition (28) can be further elaborated in the following way. The dynamics of the observation error is obtained by Eq. (27), whose result is:

$$\dot{\tilde{x}}_2(t) = -a_4\tilde{x}_2(t) - f_{under2}(x) \qquad (29)$$

The solution to this equation is given by:

$$\tilde{x}_2(t) = e^{-a_4t}\tilde{x}_2(0) - \int_0^t e^{-a_4(t-\tau)}f_{uncer2}(x)d\tau \qquad (30)$$

If the uncertain function $f_{uncer2}$ is upper-bounded, i.e. $\sup|f_{uncer2}| < \infty$, fact that holds since according to Assumption 1 the uncertainty terms are bounded, then (30) can be upper-bounded accordingly as:

$$|\tilde{x}_2(t)| \leq e^{-a_4t}|\tilde{x}_2(0)| + \frac{1}{a_4}\sup|f_{uncer2}(x)|\left(1 - e^{-a_4t}\right) \quad (31)$$

$$\leq |\tilde{x}_2(0)| + \frac{1}{a_4}\sup|f_{uncer2}(x)| \qquad (32)$$

for all $t \geq 0$. In this way, (28) is satisfied if the following condition holds:

$$Ka_3 > \left(a_2|\tilde{x}_2(0)| + \frac{a_2}{a_4}\sup|f_{uncer2(x)}| + \sup|f_{uncer1}(x)|\right) \qquad (33)$$

since

$$\begin{aligned}&a_2|\tilde{x}_2(0)| + \frac{a_2}{a_4}\sup|f_{uncer2(x)}| + \sup|f_{uncer1}(x)|\\ &\geq a_2|\tilde{x}_2(t)| + \sup|f_{uncer1}(x)|\\ &\geq |a_2\tilde{x}_2(t) - f_{uncer1}(x)| \qquad (34)\end{aligned}$$

Thus, the switching gain $K$ must be selected to fulfill (33), a condition that depends on an upper-bound of the observation error and upper bounds of the uncertainties. In the end, we must bear in mind that we are working with the square of the control signal so that the actual speed command is given by:

$$u_{actual} = \sqrt{max(0,u)} \qquad (35)$$

In the case of recovery and training programs, the controller is able to make the heart rate follow the predefined profile set up. In the next section, the simulation results showing the performance obtained by the estimation and control algorithms are presented.

## 4. SIMULATION RESULTS

This Section is composed of three subsections. First, the estimation results obtained by means of the PSO algorithm are discussed in section 4.1. Secondly, the control results obtained by using the super-twisting control law are presented in section 4.2, while the comparison between the PSO and other parameter estimation procedures and SMC comparison with PID is presented in Section 4.3.

### 4.1 PSO parameter estimation results

In this part, we will apply the PSO parameter estimation algorithm to the data described in tables below corresponding to ten subjects. These data are numerical data used for testing the algorithm and they do not correspond to real subjects. The parameters of the PSO algorithm are given by $c_1 = 0.87$, $c_2 = 0.67$, $w = 58$, $rand_1 = 0.1$, $rand_2 = 0.5$ and $\varepsilon \geq 0.15$ then tolerance is small. The actual parameters of the ten subjects are given in Tables 1 and 2 and the estimated ones obtained from the PSO proposed approach are in Tables 3 and 4.

Table 1. Actual parameters of subjects 1 - 5.

| Parameters | Subject 1 | Subject 2 | Subject 3 | Subject 4 | Subject 5 |
|---|---|---|---|---|---|
| a1 | 2.512 | 2.791 | 2.683 | 2.592 | 2.611 |
| a2 | 25.92 | 25.74 | 25.25 | 25.41 | 25.84 |
| a3 | 0.81 | 0.85 | 0.79 | 0.8 | 0.81 |
| a4 | 0.9021 | 0.9087 | 0.909 | 0.9011 | 0.9108 |
| a5 | 0.038 | 0.041 | 0.035 | 0.039 | 0.042 |
| a6 | 5.37 | 5.51 | 5.43 | 5.65 | 5.29 |
| HRrest | 64 | 69 | 66 | 68 | 69 |

Table 2. Actual parameters of subjects 6 - 10.

| Parameters | Subject 6 | Subject 7 | Subject 8 | Subject 9 | Subject 10 |
|---|---|---|---|---|---|
| a1 | 3.12 | 3.7 | 3.05 | 3.25 | 3.9 |
| a2 | 21.25 | 21.95 | 21.1 | 21.55 | 21.8 |
| a3 | 1.5 | 1.6 | 1.3 | 1.4 | 1.7 |
| a4 | 1.9 | 1.8 | 1.8 | 1.25 | 1.9 |
| a5 | 1.01 | 1.21 | 1.16 | 1.09 | 1.25 |
| a6 | 8.15 | 8.35 | 8.5 | 8.6 | 8.3 |
| HRrest | 62 | 67 | 64 | 68 | 61 |

Table 3. Estimated parameters obtained by running the PSO algorithm of subjects 1 - 5.

| Parameters | Subject 1 | Subject 2 | Subject 3 | Subject 4 | Subject 5 |
|---|---|---|---|---|---|
| a1 | 2.508 | 2.789 | 2.681 | 2.588 | 2.615 |
| a2 | 25.88 | 25.69 | 25.2 | 25.48 | 25.71 |
| a3 | 0.849 | 0.855 | 0.788 | 0.798 | 0.809 |
| a4 | 0.9011 | 0.9079 | 0.9088 | 0.9019 | 0.9098 |
| a5 | 0.04 | 0.043 | 0.039 | 0.032 | 0.04 |
| a6 | 5.41 | 5.49 | 5.47 | 5.59 | 5.59 |

Table 4. Estimated parameters obtained by running the PSO algorithm of subjects 6 - 10.

| Parameters | Subject 6 | Subject 7 | Subject 8 | Subject 9 | Subject 10 |
|---|---|---|---|---|---|
| a1 | 2.95 | 3.15 | 2.8 | 2.95 | 3.5 |
| a2 | 20.8 | 21.5 | 20.5 | 20.85 | 21.25 |
| a3 | 1.1 | 1.2 | 0.95 | 1.15 | 1.2 |
| a4 | 1.5 | 1.3 | 1.2 | 0.95 | 1.5 |
| a5 | 0.8 | 0.9 | 1.0 | 0.85 | 1.0 |
| a6 | 7.75 | 7.9 | 8.2 | 8.1 | 7.9 |

As it can be noticed from Tables 1, 2, 3 and 4, the estimated parameters obtained by the proposed PSO approach are close to the actual ones. These results mean that the PSO algorithm performs very well.

Figure 2 shows the actual heart rate corresponding to one of our subjects (subject No.3) along with the output of the estimated model obtained by using the PSO algorithm described in Section 3.1. As it can be observed in this figure, the estimated model captures the dynamics of the heart rate of each person. It means that the PSO algorithm has a good response. Moreover, PSO is able to provide accurate parameter values and able to reproduce the behavior of the heart rate. Figure 3 displays the HR generated by the original model parametrized by the parameters in Table 1 along with the values obtained when the estimated parameters values are used to obtain the HR response according to (1)-(4). We can observe some mistmatch between the actual output and the estimated one in Figure 3, which is due to the uncertain dynamics that can not be adequately captured by the model parameters. This unmodeled dynamics will be counteracted by means of the sliding mode control.

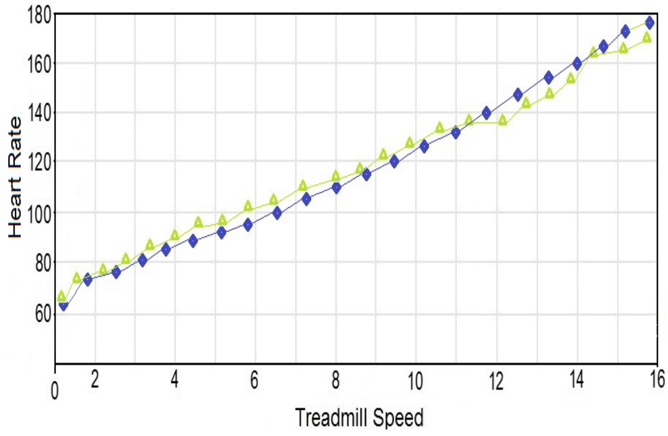


Fig. 2. Heart rate response with using PSO at speed of (2-14 km/h)

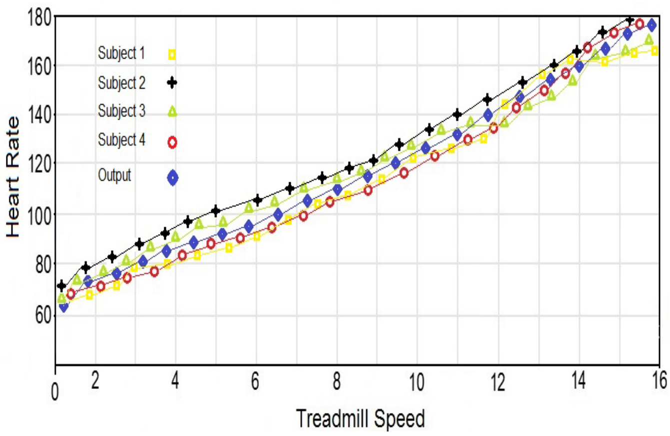


Fig. 3. Heart rate response with using PSO at speed of (2-14 km/h)- for 4 random subjects

The following figures (Fig. 4 and Fig. 5) display the evaluation of the estimated parameters for subjects No.1 and No.8 in Table 1 and the estimated onse. These figures show that after a small number of iterations the estimated parameters are close to the actual ones, the fact that is displayed numerically in Tables 2, 3 and 4.

The following Figure 6 displays the actual value of parameters for the ten subjects. Despite the value of the actual parameters with exhibit a large variability, showing that the algorithm works in a variety of situations the parameters. Thus, we have Figure 7, which is the relative error of the parameters with high variability. As it can be observed, the proposed PSO algorithm is able to achieve a superb estimation since the relative error is

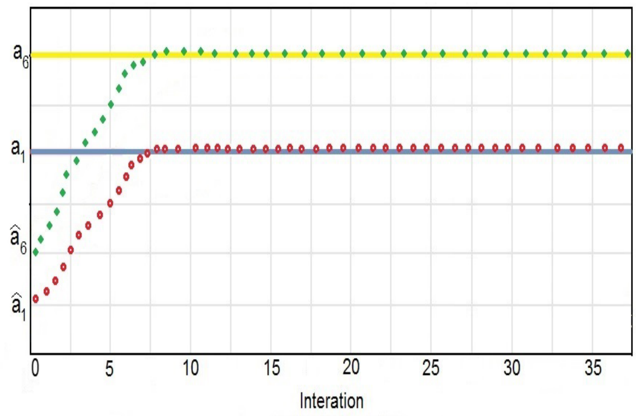


Fig. 4. Evaluation of estimated parameter subject No.1

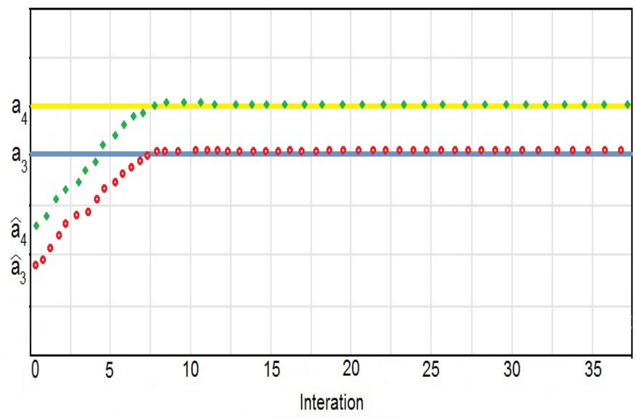


Fig. 5. Evaluation of estimated parameter subject No.8

given by $\frac{a-a_{estimation}}{a} \times 100$ which is very low and algorithm with this high variabilities has a superb response.

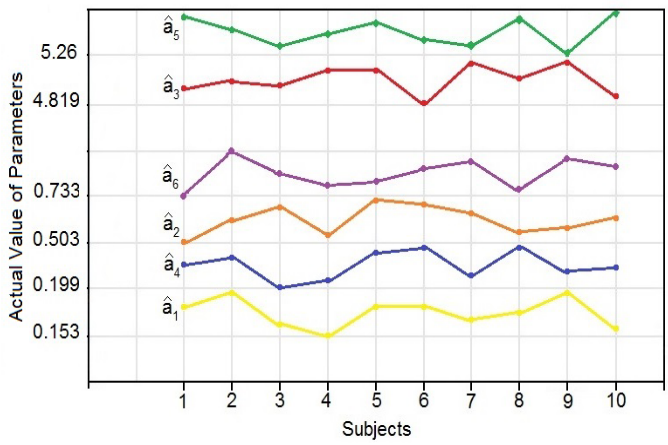


Fig. 6. Actual value of parameter.

### 4.2 *Control Results*

In this part, we choose one person (subject No.3) to show the result. We want to highlight that similar good results are obtained for all the parameters. The controller parameters are $\alpha = 0.5$ and $K = 10$. In Fig. 8, the actual Heart rate (HR)

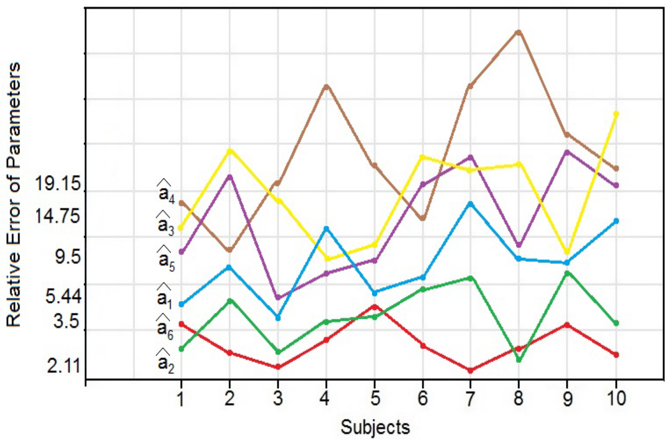


Fig. 7. Relative Error of parameter.

and the reference signal are shown, In this figure (Fig. 8), the output and the reference signal are super-impressed implying that the control objective has been achieved. The zoom of first 150 seconds in the previopus figure (Fig. 8) is shown in Figure 9. On the other hand, Figure 10 shows the speed calculated from the SMC given by Eq. (10). In this case, the tracking error after the reaching phase is very small, despite the changes in the reference signal and it shows that how the SMC had a great response regards to the absence of chattering in the output and in the control command.

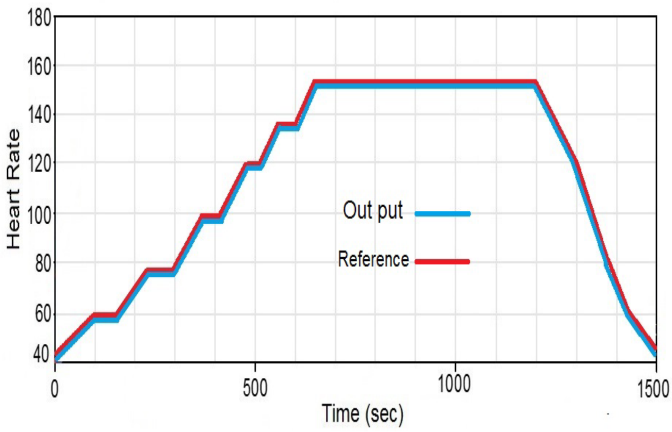


Fig. 8. Heart Rate provided by the SMC controller.

*4.3 Estimation and control comparisons*

Previously in some studies, researchers used different methods such as the M-estimator to solve the parameter estimation problem, (Peter and Huber, 1964). This method is effective, from the statistical point of view, to obtain an adequate estimation of the parameters. The comparison with the M-estimator procedure shows that PSO has better behavior than the previous approach, while the accuracy of the estimation parameter is increased by using PSO.

In this subsection, the theory of M-estimation is introduced for comparison purposes. The previous researchers used robust performance of estimation for two main reasons, namely: 1)there may be outliers in the data, that are sample values considered very different from the majority of the sample and 2)the

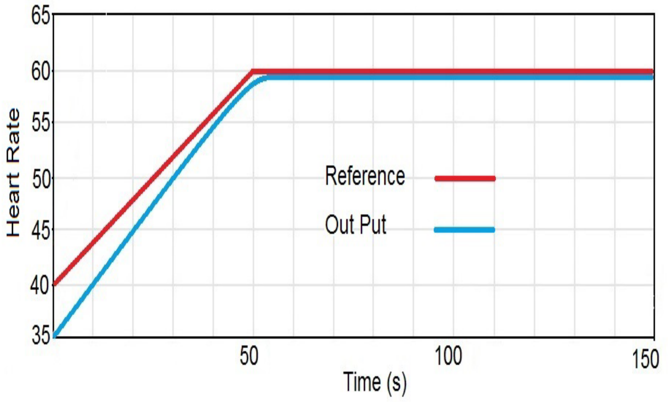


Fig. 9. The first 150 seconds HR provided by the SMC controller.

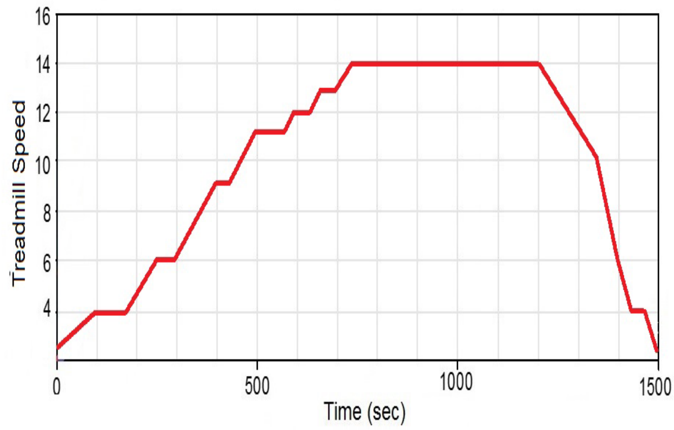


Fig. 10. Speed provided by the SMC controller.

data may depart from the underlying distribution assumptions, (Cheng et al, 2008), (Peter and Huber, 1964), (Maronna, 1976). This method is good for estimating the parameters, which is the reason why we compare our approach with this one. The class of M-estimators contains the maximum likelihood estimator (ML) as a special case. If we assume that the data come from the model distribution $F(\mu,\sigma)$ then the log-likelihood can be written as:

$$\sum_{i=0}^{n}\{log(f_0(\frac{x_i-\mu}{\sigma})-log\sigma) \tag{36}$$

The first order condition for the M-estimator of $\mu$ is then given by:

$$\frac{1}{n}\sum_{i=0}^{n}\psi_M(\frac{x_i-\mu_M}{\sigma})=0 \tag{37}$$

while the M-estimator of scale verifies

$$\frac{1}{n}\sum_{i=0}^{n}\rho_M(\frac{x_i-\mu}{\sigma_M})=1 \tag{38}$$

with $\psi_M(u)$ being the so-called score function, and $\rho_M(u)=\psi_M(u)u$. Under regularity conditions the ML estimators have a 100 percent efficiency, meaning that their asymptotic variance equals the inverse of the Fisher information, the lower bound of the Cramer-Rao inequality, (Tian et al,2014), (Peter and Huber, 1964). The parameters are coming from the particular subject (subject No.3) of Table 1, estimated by solving Eq. (17) with respect to $x_i$. The estimated parameters obtained by using the M-estimator are applied again to parametrize Eq. (1)-(4). Figure 11 demonstrates that in many points the output of PSO intersects with the reference signal and it means that PSO estimation is really close to the reference. On the other hand, the M-estimator displays an output that intercepts at some points with the reference, it has many variations in most of the points and this is not as effective as PSO. Whereas, we are calculating the fit-in error (since it is an error coming from a difference in the output of the models and actual data) in this part, that is only in open loop and it is only for estimation purposes. The fit-in error is calculated as:

$$J_{1,k} = [r_k - p_k] \tag{39}$$

$$J_{2,k} = [r_k - m_k] \tag{40}$$

where : $r_k = reference$

$p_k = PSO\ output$

$m_k = M - estimator\ parameter\ output$

Fig. 12 shows the fit-in results of the output in the PSO and M-estimator. We want to show that PSO outperforms the M-estimator. As it can be seen at the beginning of the process in Figure 12, PSO after 500 seconds starts definitely better than the M-estimator and it slightly goes better afterward. Both of these models have a good response, but PSO performs better during the fit-in error process. On the other hand, PSO has a lower error against the M-estimator.

Finally, in Fig. 13 the value of the cost function (Eq. (5)) in Particle Swarm Optimization and M-estimator are displayed. The cost function (2) in the PSO after 5 iterations reduces faster than the M-estimator. So the fact that can be interpreted as that estimation is performed faster in PSO method against the M-estimator, and this fact is reflected in the quality of estimation.

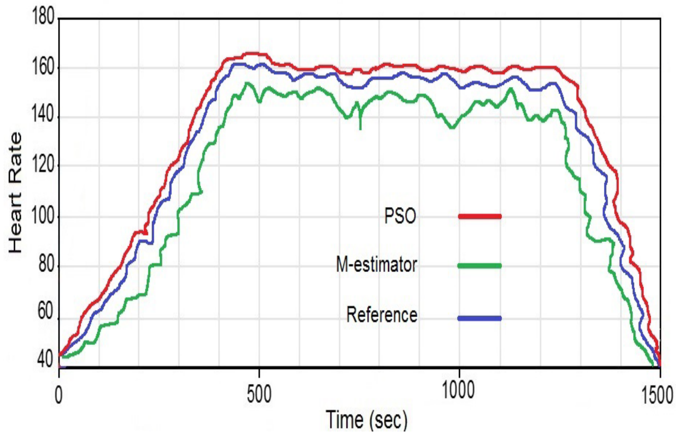


Fig. 11. Comparing heart rate tracking result with PSO and M-estimator.

On the other hand, PID is a common approach in the control of systems. For this reason, it is used to solve the control problem in many studies. The SMC control will be compared with the PID controller implemented in (Girard et al, 2016). Since PID controllers are widely used in practice we will show the results achieved by the proposed controller in this scenario. In Fig.

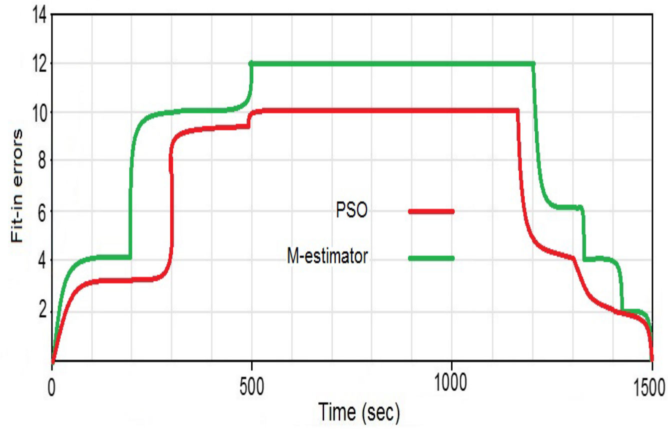


Fig. 12. Fit-in error for the PSO and M estimators.

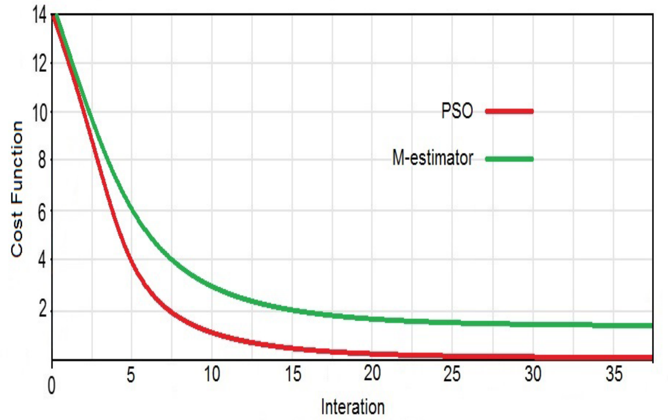


Fig. 13. Comparing error with PSO and M-estimator.

14, the comparison of the actual HR obtained by using the SMC and the PID controller is shown. As it is clear, the SMC works much better than PID showing that it is able to obtain a zero tracking error despite the presence of uncertainties in the system's model. Overall, the presented method is able to obtain an appropriate and superb closed-loop behaviour.

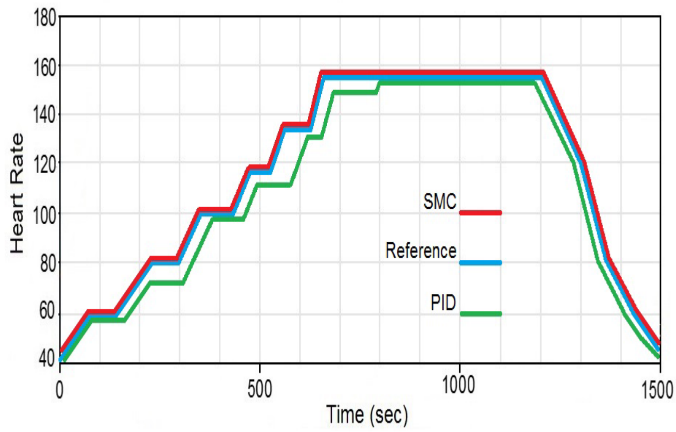


Fig. 14. Heart rate Comparision with SMC and PID.

## 5. CONCLUSIONS

This paper has considered the design of a sliding mode controller for the HR control during treadmill exercise. Initially, a Particle Swarm Optimization algorithm has been proposed to obtain an accurate estimation of the model parameters. Secondly, a super-twisting based sliding control law has been designed for the system in order to counteract the remaining potential unmodelled dynamics or parametric uncertainty in the system. In both situations, the range of treadmill speeds goes from 2 up to 14 km/h, range not usually employed in previous studies. Simulation results show how the proposed PSO algorithm is able to obtain accurate estimations of the model parameters while the super-twisting sliding controller is capable of obtaining zero tracking error without chattering.

## ACKNOWLEDGEMENTS

This work was partially supported by the Spanish Ministry of Economy and Competitiveness through grant DPI2016-77271-R and by the University of the Basque Country (UPV/EHU) through grant PPG17/33.

## REFERENCES

Abu-Rmileh, A., Garcia-Gabin, W., and Zambrano, D. (2010). Internal model sliding mode control approach for glucose regulation in type 1 diabetes. Biomedical Signal Processing and Control, 5(2), 94-102.

Anselmino, M., Scarsoglio, S., Saglietto, A., Gaita, F., and Ridolfi, L. "A computational study on the relation between resting heart rate and atrial fibrillation hemodynamics under exercise." PloS one 12.1 (2017): e0169967.

Argha, A., Su, S. W., and Celler, B. G. (2017). Heart rate regulation during cycle-ergometer exercise via event-driven biofeedback. Medical and biological engineering and computing, 55(3), 483-492.

Bansal, J. C., Singh, P. K., Saraswat, M., Verma, A., Jadon, S. S., and Abraham. "Inertia weight strategies in particle swarm optimization." Nature and Biologically Inspired Computing (NaBIC),IEEE, 2011. 2011 Third World Congress on.

Belkaid, A., Gaubert, J. P., and Gherbi, A. (2016). An improved sliding mode control for maximum power point tracking in photovoltaic systems. Journal of Control Engineering and Applied Informatics, 18(1), 86-94.

Cheng TM, Savkin AV, Celler BG, Su SW, Wang L. Heart rate regulation during exercise with various loads: identification and nonlinear H infinity control. InIFAC World Congress 2008. 2008 IFAC.

Cheng TM, Savkin AV, Celler BG, Su SW, Wang L. Nonlinear modeling and control of human heart rate response during exercise with various work load intensities. Biomedical Engineering, IEEE Transactions on. 2008 Nov;55(11):2499-508.

Dan, A. M., and Dragomir, T. L. (2012, May). A model of the control function in the case of constant effort of the cardiovascular system. In Applied Computational Intelligence and Informatics (SACI), 2012 7th IEEE International Symposium on (pp. 169-173). IEEE.

Ebrahimi, N., Ozgoli, S., and Ramezani, A. (2018). Model-free sliding mode control, theory and application. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 0959651818780597.

Esmaeili, A., and Ibeas, A. (2016, October). Particie Swarm Optimization modelling of the heart rate response in treadmill exercise. In System Theory, Control and Computing (ICSTCC), 2016 20th International Conference on (pp. 613-618). IEEE.

Eswaran, T., and V. Suresh Kumar. "Particle swarm optimization (PSO)-based tuning technique for PI controller for management of a distributed static synchronous compensator (DSTATCOM) for improved dynamic response and power quality." Journal of Applied Research and Technology 15.2 (2017): 173-189.

Gao, Xuehui. Variable gain super-twisting sliding mode control for Hammerstein system with Bouc-Wen hysteresis nonlinearity. In: Control Conference (CCC), 2016 35th Chinese. IEEE, 2016. p. 3369-3372.

Girard, C., Ibeas, A., Vilanova, R., and Esmaeili, A. (2016, May). Robust discrete-time linear control of heart rate during treadmill exercise. In Electrical Engineering (ICEE), 2016 24th Iranian Conference on (pp. 1113-1118). IEEE.

Guo, H., Liu, Y., Liu, G., and Li, H. (2008). Cascade control of a hydraulically driven 6-DOF parallel robot manipulator based on a sliding mode. Control Engineering Practice, 16(9), 1055-1068.

Hamed, E., Anushya, A., Alzoubi, R., and Vincy, B. A. (2017, August). An analysis of particle swarm optimization for feature selection on medical data. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 227-231). IEEE.

Hunt, Kenneth J., and Simon E. Fankhauser. "Heart rate control during treadmill exercise using input-sensitivity shaping for disturbance rejection of very-low-frequency heart rate variability." Biomedical Signal Processing and Control 30 (2016): 31-42.

Ibeas, A., Esmaeili, A., Herrera, J., and Zouari, F. (2016, October). Discrete-time observer-based state feedback control of heart rate during treadmill exercise. In System Theory, Control and Computing (ICSTCC), 2016 20th International Conference on (pp. 537-542). IEEE.

Jang, D. G., Ko, B. H., Sunoo, S., Nam, S. S., Park, H. Y., Bae, S. K. (2016, August). A preliminary study of a running speed based heart rate prediction during an incremental treadmill exercise. In Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the (pp. 5323-5326). IEEE.

Kranjec, Beguš, Geršak , Drnovšek. "Non-contact heart rate and heart rate variability measurements: A review." Biomedical Signal Processing and Control 13 (2014): 102-112.

Liu, W., Liu, L., Chung, I. Y., and Cartes, D. A. (2011). Real-time particle swarm optimization based parameter identification applied to permanent magnet synchronous machine. Applied Soft Computing, 11(2), 2556-2564.

Lu, C. H., Wang, W. C., Tai, C. C., and Chen, T. C. (2016). Design of a heart rate controller for treadmill exercise using a recurrent fuzzy neural network. Computer methods and programs in biomedicine, 128, 27-39.

Maronna, R.A. (1976). Robust M-estimators of Multivariate Location and Scatter, The Annals o Statistics 4, 51-67.

Meyer, Daniel, and Veit Senner. "Evaluating a heart rate regulation system for human–electric hybrid vehicles." Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology 232, no. 2 (2018): 102-111.

Paul, Arun K., J. E. Mishra, and M. G. Radke. "Reduced order sliding mode control for pneumatic actuator." IEEE Transactions on Control Systems Technology 2, no. 3 (1994): 271-276.

Peter J. Huber, Robust Estimation of a Location Parameter , The Annals of Mathematical Statistics, Vol. 35, No. 1. (Mar., 1964), pp.73-101.

Rini DP, Shamsuddin SM, Yuhaniz SS. Particle swarm optimization: Lari A, Khosravi A, Rajabi F. Controller design based on $\mu$ analysis and PSO algorithm. ISA transactions. 2014 Mar 31;53(2):517-23.

Shtessel, Yuri B. Super-twisting adaptive sliding mode control: A Lyapunov design. In: Decision and Control (CDC), 2010 49th IEEE Conference on. IEEE, 2010. p. 5109-5113.

Shtessel, Yuri B., Jaime A. Moreno, and Leonid M. Fridman. "Twisting sliding mode control with adaptation: Lyapunov design, methodology and application." Automatica 75 (2017): 229-235.

Souravlias D, Parsopoulos KE (2016) Particle swarm optimization with neighbourhood-based budget allocation. Int J Mach Learn Cybern 7(3):451–477.

Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11:341–359

Scalzi, S., Tomei, P., and Verrelli, C. M. (2012). Nonlinear control techniques for the heart rate regulation in treadmill exercises. IEEE Transactions on Biomedical Engineering, 59(3), 599-603.

Su SW, Huang S, Wang L, Celler BG, Savkin AV, Guo Y, Cheng TM. Optimizing heart rate regulation for safe exercise. Annals of biomedical engineering. 2010 Mar 1;38(3):758-68.

Suganthan PN (1999) Particle swarm optimizer with neighborhood operator. In: Proceedings of the IEEE congress on evolutionary computation, Washington, DC, pp 1958–1961

Swikir, Abdalla, and Vadim Utkin. "Chattering analysis of conventional and super twisting sliding mode control algorithm." Variable Structure Systems (VSS), 2016 14th International Workshop on. IEEE, 2016.

Tian, Xiaomin, and Shumin Fei. "Robust control of a class of uncertain fractional-order chaotic systems with input nonlinearity via an adaptive sliding mode technique." Entropy 16.2 (2014): 729-746.

Vaidyanathan, Sundarapandian. "Super-Twisting Sliding Mode Control of the Enzymes-Substrates Biological Chaotic System." Applications of Sliding Mode Control in Science and Engineering. Springer International Publishing, 2017. 435-450.

Wang, X., Wang, S., and Ma, J. J. (2007). An improved co-evolutionary particle swarm optimization for wireless sensor networks with dynamic deployment. Sensors, 7(3), 354-370.

Weippert, M., Behrens, M., Rieger, A., Behrens, K. Sample entropy and traditional measures of heart rate dynamics reveal different modes of cardiovascular control during low intensity exercise,(2014). Entropy, 16(11), 5698-5711.

Xue, B., Zhang, M., and Browne, W. N. (2013). Particle swarm optimization for feature selection in classification: A multi-objective approach. IEEE transactions on cybernetics, 43(6), 1656-1671.

Yan, X., Wu, Q., Liu, H., Huang, W. (2013). An improved particle swarm optimization algorithm and its application. International Journal of Computer Science Issues (IJCSI), 10(1), 316.

Yang, H., Xu, Y., Peng, G., Yu, G., Chen, M., Duan, W., and Wang, X. (2017). Particle swarm optimization and its application to seismic inversion of igneous rocks. International Journal of Mining Science and Technology, 27(2), 349-357.

Zhang, Yudong, and Lenan Wu. "Crop classification by forward neural network with adaptive chaotic particle swarm optimization." Sensors 11.5 (2011): 4721-4743.