In: Systems Engineering Using Particle Swarm Optimisation ISBN 1-60021-119-4
Editors: Nadia Nedjah et al, pp. 85–112

Chapter 5

PIECEWISE REFERENCE-DRIVEN TAKAGI-SUGENO FUZZY MODELLING BASED ON PARTICLE SWARM OPTIMIZATION

Ernesto Araujo[*,1] ***and Leandro dos Santos Coelho***[**,2]

[*]Integration and Testing Laboratory – LIT,
Instituto Nacional de Pesquisas Espaciais – INPE,
Av. Astronautas, 1758, 12.227-010, So Jos Campos, SP, Brazil

[**]Automation and Systems Laboratory – LAS-PPGEPS,
Pontifcia Universidade Catlica do Paran – PUC-PR,
R. Imaculada Conceio, 1155, Curitiba, PR, Brazil

A nonlinear identification approach based on Particle Swarm Optimization (PSO) and T-S fuzzy model for describing dynamical behaviour of a thermal-vacuum system is proposed in this paper. Employed for space environmental emulation and satellite qualification, thermal-vacuum systems are inherently nonlinear and present time-delay characteristics. Despite the success of linear techniques of modelling and identification when dealing with nonlinear dynamic systems, they are usually not appropriate. Nevertheless, only recently has much ongoing research addressed the problem of nonlinear identification approach. In this sense, fuzzy systems based on Takagi-Sugeno (T-S) model have increasingly been employed for the identification of nonlinear systems. In order to find out an optimal nonlinear model swarm intelligence methodology (PSO) is employed as a method for optimizing the premise part of production rules while batch least mean squares technique is employed

[1]E-mail address: ernesto.araujo@lit.inpe.br, http://www.lit.inpe.br/ ernesto
[2]E-mail address: leandro.coelho@pucpr.br, http://www.produtronica.pucpr.br/leandro

for consequent part of production rules of a T-S fuzzy model. This hybrid PSO-TS fuzzy approach is employed here to generate piecewise, gain-scheduling sub-models concerning diverse operational conditions driven by the reference. This reference can additionally be associated to distinct goals, context, or other exogenous signals. Experimental data obtained from thermal-vacuum system is used for identification process. Numerical results indicate that the PSO succeeded in constructing a piecewise, gain-scheduling T-S fuzzy model for nonlinear identification in this particular application.

5.1 Introduction

Modelling of nonlinear dynamic processes from operating data is fundamental to diverse engineering problems. A model may be understood as any sort of abstract description that captures useful relevant features able to represent a system. Finding out a model for representing dynamical behaviour is of particular importance when dealing with nonlinear, time-delay thermal-vacuum chambers used for satellite qualification. Once in space, satellites are exposed, but not limited, to sunshine, Albedo radiation, earth radiation, shadow/eclipse conditions, and earthshine infrared. Thermal-vacuum chambers are used to reproduce as close as possible environmental conditions of expected post-launch environments which satellites will experience during their in-flight life, that is, their operational life [15].

Thermal-vacuum systems consist of a chamber, a shroud (set of pipes) which heats or cools off the environment, and auxiliary equipment (Figure 5.1). In the thermal-vacuum system used at Integration and Testing Laboratory (LIT) in the Brazilian National Institute for Space Research (INPE), the original controller was designed to control the temperature on the shroud (Figure 5.2). Nevertheless, requirements for the space sector establish that the controlled variable must be the temperature at the surface of the specimen under test. This system is characterized for being nonlinear, presenting time-delay, and working in diverse points of operation. Due to these characteristics operators in a hand made procedure currently conduct thermal-vacuum testing (Figure 5.3).

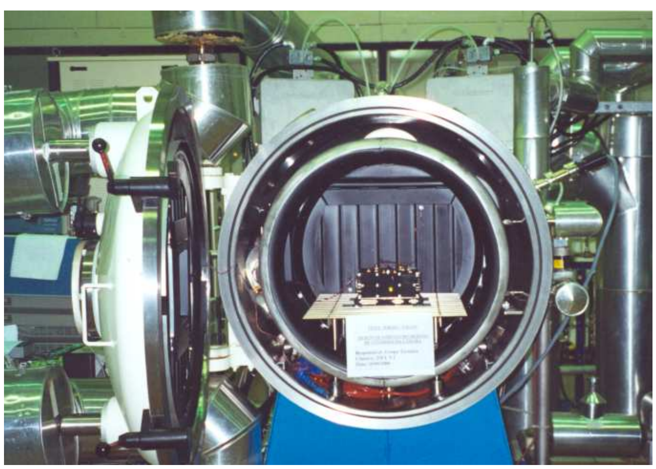

Figure 5.1. Thermal-vacuum chamber with passive load.

A first attempt for dealing with this problem was to use a feasible approach named

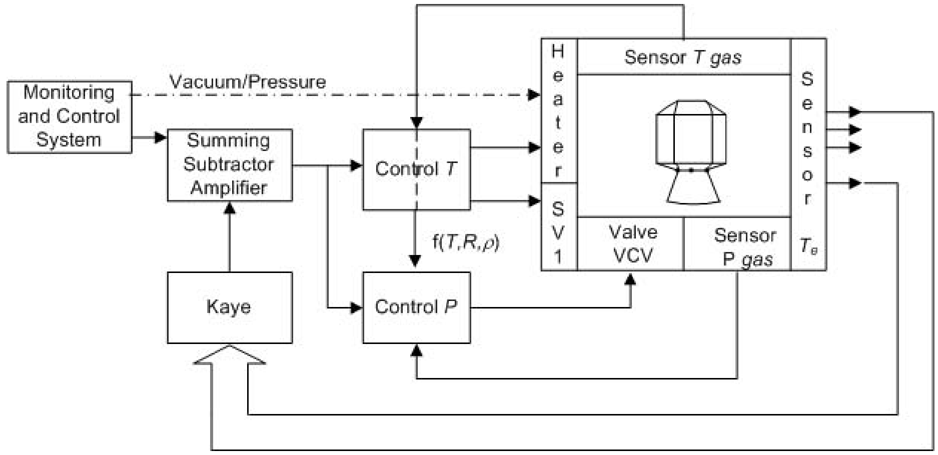

Figure 5.2. Original thermal-vacuum control system.

Fuzzy Reference Gain-Scheduling (FRGS) control system [1] [2] understood as well as a mechanism for decision support system [3]. This controller was designed by using heuristic and expertise obtained through knowledge engineering.

Despite the success in using this reference-driven fuzzy control system, an immediate alternative to set up the control parameters is by first finding out a fuzzy model for describing the thermal-vacuum system and later to use this information to achieve an optimized fuzzy control system.

The interest in obtaining a fuzzy model goes beyond control application. Additional advantages of identifying a model for thermal-vacuum systems are, for instance, the ability to detect loss of vacuum, presence of unknown heat sources or sinks, training of thermal-vacuum operators, development of a supervisor decision-support system for helping operators to control the whole system, and checking the instantaneous operator's behaviour or performance.

A question that arises is which modelling approach would be more suitable for automatically modelling thermal-vacuum systems. System identification is a relevant step in system analysis of nonlinear processes as well as model-based control design. System identification allows building mathematical models of dynamic systems based on measured data. The use of linear mathematical models through identification is adequate for several applications. Whilst such an approximation may be acceptable in many signal processing and control applications, there are several advantages that can be obtained if a nonlinear model identification technique is applied when dealing with a nonlinear model of the process [39]. Moreover, conventional modelling approach seems not appropriate for this task since thermal-vacuum systems are highly nonlinear, presents time-delay, and changes its dynamic behaviour in many different operational conditions.

Nonlinear system identification is more challenging and it has received less attention when compared to linear system identification. There are several approaches to nonlinear system identification. Frequency response methods, correlation analysis, regression methods, wavelets transformations, kernel methods, batch and recursive identification algorithms based on least mean square technique, fuzzy systems, evolutionary computation,

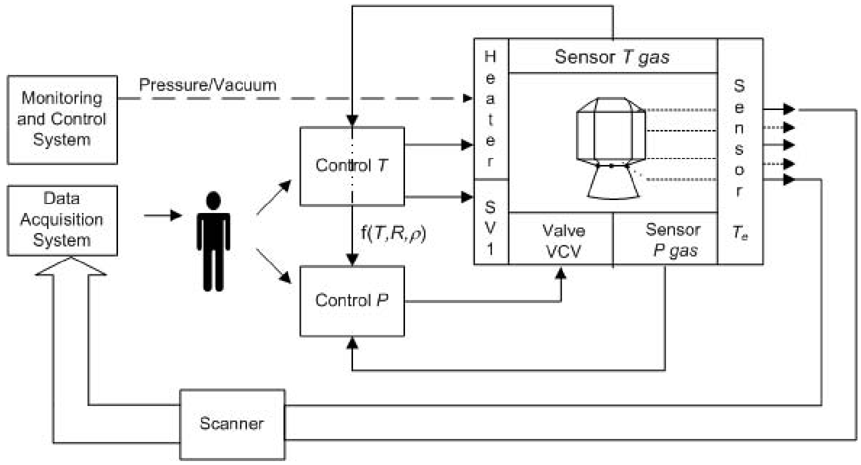

Figure 5.3. Current thermal-vacuum operation.

neural networks, intelligent hybrid systems, and NARMAX (Nonlinear Auto-Regressive Moving Average with eXogeneous inputs) models belong to this category [17] [27].

An alternative for copping with modelling this nonlinear and time-delay system is to employ fuzzy models since this approach exhibits both high nonlinearity and robustness to uncertainty in data. Further, fuzzy models are regarded as universal approximators. Advantages of using fuzzy modelling include its simple structure to describe nonlinear systems as well as its ability to represent human being behaviour [4]. Since human beings handle the thermal-vacuum system this fuzzy approach seems a natural method for both designing a control system and/or modelling the dynamical system. Fuzzy models are based on rules such as 'if premise then consequence' where premises evaluate the model inputs and consequences provide the value of the model output. In this paper, the Takagi-Sugeno (T-S) fuzzy model [37] [36] is employed. This approach consists of IF-THEN rules where the rule consequents are usually linear functions of the inputs.

Many optimization methods have been proposed to elicit fuzzy models through input-output data [5] [41]. In a previous work [29] a feasible identification process was carried out for describing the dynamics of the thermal-vacuum system when employing a hybrid intelligent approach. There, Particle Swarm Optimization (PSO) was employed as an auxiliary mechanism for achieving an optimal T-S fuzzy model. Working in synergy, PSO allows determining the premise space partition and to obtain membership functions, i.e., to extract the best shape, supports and cores membership functions, as well as to determine the statements in the consequent of the rules.

Particle Swarm Optimization (PSO), in turn, is a form of swarm intelligence for obtaining optimal solutions. Swarm intelligence is the emergent collective intelligence of groups of simple autonomous agents. Here, an autonomous agent is a subsystem that interacts with its environment but acts relatively with independence from all other agents [26]. The PSO method simulates social behaviour of organisms, such as bird-flocking and fish-schooling [13] [21]. The idea is that when a bird in a flock tries to find food it uses not only its own knowledge and experience but also its neighbours' experiences. The particles are flying through a hyperspace of possible solutions and remember the best position that

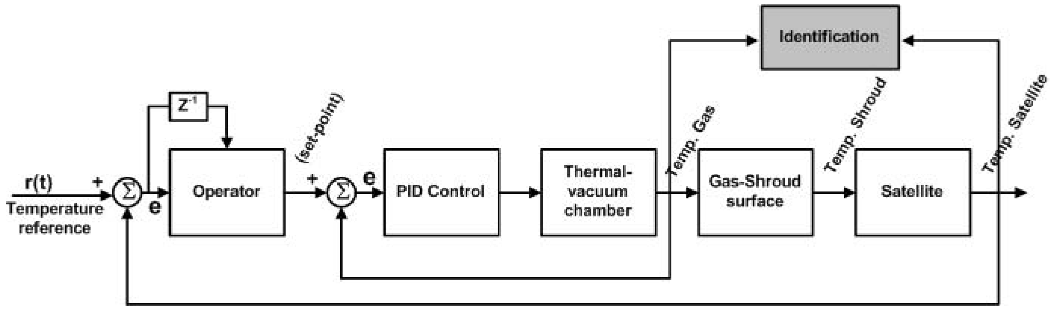

Figure 5.4. Block diagram of the thermal-vacuum system.

they have seen. In order to contribute to determine the best global solution members of the swarm communicate their best local positions to each other and adjust their own position and velocity computing also the correspondent information received from other members.

Numerical results indicated that PSO succeeded in constructing a single global T-S fuzzy model through nonlinear identification in this particular application. Despite the suitability of this method another question that comes up is concerning its performance when taking in account the specificity of the distinct operational conditions. Due the high nonlinearities present in thermal-vacuum chambers, instead of using a single model to represent the entire range of operational conditions driven by the reference, an alternative is to model the system by using piecewise, gain-scheduling, reference-driven fuzzy T-S models. Modelling nonlinear systems with gain-scheduling methodology involves applying several linear models of similar structure over a partitioned input space [42] [38]. In this sense, T-S models are used to represent the best local models to operation in different regions of thermal-vacuum system. The best model for each region will be obtained in accordance with a basic T-S fuzzy model representation of the process by means of finite set of models. This approach can be interpreted as a switching multiple model approach [9] but with an interpolative scheme between linearized models. Computing several suitable mathematical models for the system, thus, can be useful for forecasting its behaviour under different operating conditions, as well as for designing the control law that will make the whole system perform in a desired way.

This paper explores the ability of PSO to derive the parameters of premise part for generating piecewise fuzzy systems for a nonlinear system working in distinct operational conditions established by an exogenous input. The consequent part of production rules of T-S fuzzy system is accomplished by least mean squares approach. This paper focus on the nonlinear identification for modelling the relationship between the temperature on satellite (output) and the controlled temperature of the gas inside shroud (input) which is used to change the temperature in the interior of the chamber. The simplified diagram that depicts the operational characteristic of the thermal-vacuum chamber and the identification block is presented in Figure 5.4).

5.2 FUZZY GAIN-SCHEDULING MODELLING

In the data-driven modelling community, two main paradigms have emerged: global versus divide and conquer. Global modelling builds a single functional model on the basis of the dataset. Divide and conquer techniques divide a complex problem into simpler ones whose particular solutions can be combined to provide a global solution for the original problem [6]. The main reason for using multiple models is to ensure the existence of at least one model with parameters sufficiently close to the unknown plant. One alternative for implementing this proceeding is to use in multiple models piecewise, gain-scheduling fuzzy systems [42] [38]. In doing so, the global nonlinear system is obtained by interpolating these sub-models generated according to diverse points of operation.

Consider, for instance, a nonlinear system described as:

$$\begin{aligned} \dot{x} = & \quad f(x(t), u(t), \theta) \\ y = & \quad g(x(t), u(t), \theta) \end{aligned} \tag{5.1}$$

where f is a nonlinear function, g is a measurement function, x is the state vector, u is the control input vector, q is the vector of possibly time varying parameters, and y is the output vector.

When dealing with a system that presents different regions of operation, the modelling problem may be described as a linearization scheduling in which eq. (5.1) is linearized with respect to a suitable set of pre-established variables. Thus, a set of linear systems or approximation for a nonlinear plant is obtained through scheduling variables.

In linearization scheduling problem, the nonlinear system may be rewriting, for instance, through linearization transformation or through Taylor expansion along a trajectory or over operational points (x_i, u_i) corresponding to ai regions in such a way that eq. (5.1) becomes:

$$\begin{aligned} \dot{x}_L = & \quad A_i x_L + B_i u_L \\ y = & \quad C_i x_L \end{aligned} \tag{5.2}$$

where:

$$A_i = \frac{\partial(f)}{\partial(x)}\Big|_{(x_i,u_i,\theta_i)}, B_i = \frac{\partial(f)}{\partial(u)}\Big|_{(x_i,u_i,\theta_i)}, C_i = \frac{\partial(g)}{\partial(x)}\Big|_{(x_i,u_i,\theta_i)},$$

and $x_L(t)$ is a state vector of the linearized system, and $x_i(t)$, $u_i(t)$ is the trajectory satisfying $\dot{x}_0(t) = f(x_0(t), u_0(t), \theta_0)$.

While this approach is largely employed in nonlinear control systems, named gain-scheduling controllers [31] [23] [24] [32], here this approach is applied to model a system in which its dynamical behaviour is established according to distinct operational conditions driven by the reference (set-points). If the embedded idea behind gain scheduling approach is to design a global system by using associated local linearized plant models, fuzzy gain-scheduling approach interpolates these sub-models through membership functions (Figure 5.5).

For discrete, linear, time-invariant, single-input-single-output, controllable, and observable systems the matrices A, B and C are chosen in such a way that the eq. (5.2) can be rewritten as:

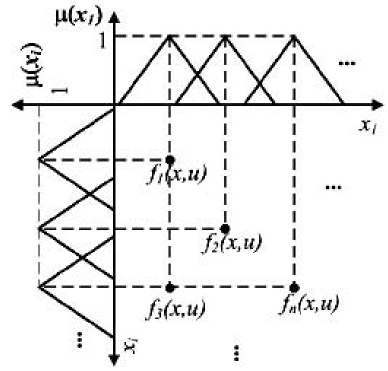

Figure 5.5. Gain-scheduling fuzzy system.

$$y_p(k+1) = \sum_{i=0}^{n-1} a_i y_p(k-i) + \sum_{j=0}^{m-1} b_i u_p(k-j), \tag{5.3}$$

where ai and bj are constant unknown parameters. Each p model in eq. (5.3) will represent a point of operation as a subset in the Tagaki-Sugeno fuzzy model.

5.2.1 Fuzzy Models

A fuzzy system is a *nonlinear mapping* from the inputs space vector to a scalar output space represented as a function, $f : X \rightarrow Y$, where X and Y is universe of discourses. This mapping is accomplished by a set of IF-THEN rules in the form "IF <X is A> THEN <Y is B>" that defines the input-output space, X x Y, and an associated fuzzy inference mechanism. Different from the classical crisp mathematical function (including crisp interval function), in fuzzy function each element inside a fuzzy region assumes a degree of fulfilment between 0 and 1 (Figure 5.6b). Each rule defines a fuzzy region as depicted in Figure 5.6a, that is called granule, patch, or cluster, according to diverse fields of research. The fuzzy subset of the input space X is A and B is the fuzzy subset of the output space, and are also known as membership function or linguistic terms. Along with fuzzy rules, fuzzy set is another element that is used for partitioning the universe of discourse and defining the number of overlapping fuzzy regions. There will be as many fuzzy regions as the number of fuzzy sets in each universe of discourse. For example, if there are three fuzzy sets in the input space X, that is A_1, A_2, A_3, and two fuzzy sets in the output space, that is B_1, B_2, then there may be six fuzzy regions. However, these fuzzy regions are not all available; there must be a set of rules mapping a fuzzy set in the input space into the output space, as shown in Figure 5.6b.

For a multi-input single-output (MISO) fuzzy model the rules have the form:

$$R^{(j)} : IF\ (z_1\ IS\ A_1^j)\ AND\ \ldots\ AND\ (z_m\ ISA_m^j)\ THEN\ (y\ IS\ B^j) \tag{5.4}$$

where the input vector of the premise is given by $z = [z_1, \ldots, z_m]^T$, $i = 1, \ldots, m$; y is the output vector of the conclusion; A_i^j are linguistic input terms; B^j is a linguistic output term. The resulting fuzzy function or fuzzy relation is given by the aggregation of the set of fuzzy

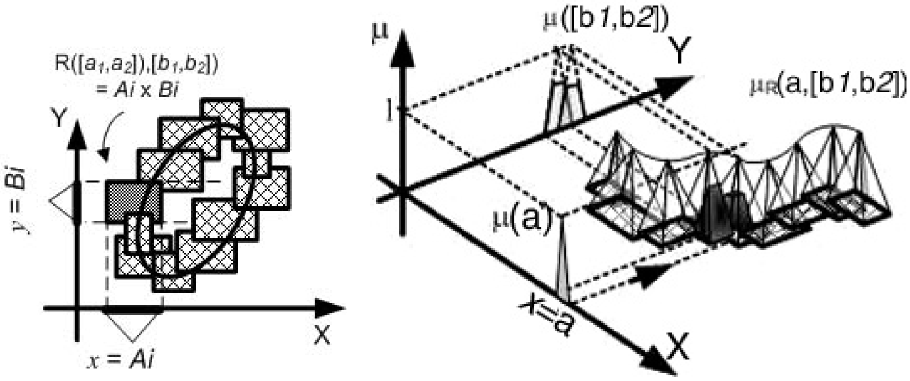

Figure 5.6. Nonlinear input-output mapping when fuzzy regions assume a degree of fulfilment between 0 and 1.

rules, R^j. The aggregation operator for generating the fuzzy relation is associated to a *t-conorm*, usually the *max* operation.

In the presence of a singleton (crisp) input, for instance, this data is first associated to one membership function, *Ai*. When a linguistic term is associated to the input data this process is named *fuzzyfication*. The resulting *degree of membership* is obtained where this singleton crosses the membership function. When there are diverse conditional statements in the premise one degree of membership among many from different input linguistic input variable must be chosen. For this task a t-norm is employed to perform the fuzzy conjunction. For example, in a Cartesian space usually a min operator or a product operator carries out this task. The resulting degree of fulfilment is propagated and weights the consequent of each active rule. When there is more than one active rule a *t-conorm* is employed to perform the fuzzy disjunction. In a Cartesian space usually a max operator carries out this task. A *defuzzyfication* process accomplishes the final resulting where the ordinary value is the centre of area.

Basically there are three categories of fuzzy systems models: relational fuzzy model, linguistic fuzzy model also known as Mamdani fuzzy model, and the interpolative (linear, first order functions) model also known as Takagi-Sugeno fuzzy model. The structure for identification chosen in this approach is the Takagi-Sugeno fuzzy model.

5.2.2 Takagi-Sugeno Fuzzy Models

The essential idea of T-S fuzzy model is the partitioning of the input space into fuzzy areas and the approximation of each area through a linear model in such a way that a global nonlinear model is computed. It is characterized as a set of IF-THEN rules where the consequent part are linear sub-models describing the dynamical behaviour of distinct operational conditions meanwhile the antecedent part is in charge of interpolating these sub-systems. The “IF statements” define the premise part that is featured as linguistic terms while the THEN functions constitute the consequent part of the fuzzy system characterized, but not limited, as linear polynomial terms. The global model is then obtained by the interpolation between these various local models. This model can be used to approximate a highly nonlinear function through simple structure using a small number of rules. The following general form represents the T-S model:

$$R^{(j)}: \; IF \; (z_1 \; IS \; A_1^j) \; AND \; \ldots \; AND \; (z_m \; IS \; A_m^j) \; THEN \; (y \; = \; b_0^j + b_1^j x_1^j + b_{q_j}^j x_{q_j}^j) \tag{5.5}$$

The input vector of the premise is given by $z = [z_1, \ldots, z_m]^T$, $i = 1, \ldots, m$, and A_i^j are linguistic terms (labels) of fuzzy sets. The fuzzy sets pertaining to a rule form fuzzy regions within the input space, A_1^j x A_2^j x ... x A_m^j. The element $x = [x_1^j, \ldots, x_{q_j}^j]^T$ represents the input vector to the consequent part of R^j that comprises q_j terms; $y_i = y_j(x^j)$ denotes the j-th rule output which is a linear polynomial of the consequent input terms u_1^j; and $b = [b_0^j, b_1^j, \ldots, b_{q_j}^j]^T$ are the polynomial coefficients that form the consequent parameter set. Usually the input vector, z, is related to the elements of x, that is, $z(x)$, or even $z = x$.

Given the input vectors z and x^j, $j = 1, \ldots, M$, the final output of the fuzzy system is inferred by taking the weighted average of the local outputs $y_j(x^j)$

$$y = \sum_{j=1}^{M} v_j(\underline{z}) \cdot y_j(\underline{x}^j) \tag{5.6}$$

where M denotes the number of rules and $v_j(z)$ is the normalized firing strength of $R(^j)$, which is defined as

$$v_j(\underline{z}) = \frac{\mu_j(\underline{z})}{\sum_{j=1}^{M} \mu_j(\underline{z})} \tag{5.7}$$

and

$$\mu_j(\underline{z}) = \mu_{A_1^j}(z_1) \cdot \mu_{A_2^j}(z_2) \cdots \mu_{A_m^j}(z_m). \tag{5.8}$$

Linguistic labels A_i^j may be, for instance, associated with Gaussian membership functions,

$$\mu_{A_1^j}(z_i) = \exp\left[-\frac{1}{2}\frac{(z_i - m_{ij})^2}{\sigma_{ij}^2}\right] \tag{5.9}$$

where m_{ij} and s_{ij} are the centres (mean value) and the spreads (standard deviations) of the Gaussian function, respectively, that defines the core and the support of membership functions.

When dealing with fuzzy model identification, instead of static functions (5.5), input space X is replaced by a finite number of past inputs and past outputs of the system representing the system dynamics [5]. In doing so, the T-S models employ regression type of rules that maps the current state and input variable into the output variable and eq. (5.5) is related to eq. eq. (5.3) in the following form:

$$\begin{aligned} &R^{(j)}: \; IF \; y(k) \; IS \; A_1^j \; AND \; \ldots \; AND \; y(k-n+1) \; A_n^j \\ &AND \; u(k) \; IS \; B_1^j \; AND \; \ldots \; AND \; u(k-m+1) \; B_m^j \\ &THEN \; \hat{y}_j(k+1) \; = \; \textstyle\sum_{i=0}^{n-1} \; a_p^j \; y(k-i) \; + \; \sum_{p=0}^{n-1} \; b_p^j \; u(k-p) \; + \; c^j \end{aligned} \tag{5.10}$$

The objective of the optimization process consists of determining (tuning) these unknown parameters, a_i, b_i, c_i, A_i, and B_i, (represented as θ) when using measured input-output data so that a performance measure based on the output errors is minimized

$$min_{\theta} \sum_{k=1}^{N} \|\widehat{y}(k+1) - y(k+1)\|. \tag{5.11}$$

In eq. (5.11), $\widehat{y}(k+1)$ is the estimate output (or fuzzy function approximation) used for computing the square error when compared with the actual output, $y(k+1)$. This activity corresponds to the parameter-learning task and, consequently, the parameter estimation process. The identification problem in T-S fuzzy modelling involves not only parameter identification but structure identification as well. The structure identification, in turn, consists of determining the premise space partition and extracting the number of rules and determining the structure of the output elements (equations), respectively.

The identification of T-S system is realized in this paper based on PSO for premise part optimization while the consequent part optimization is realized by batch least mean squares method [27]. The batch least mean squares method requires the whole data set of the input and output (all training data) and is implemented off-line.

5.3 Particle Swarm Optimization (PSO)

The PSO originally developed by Kennedy and Eberhart in 1995 is a population-based swarm algorithm. PSO is a stochastic global optimization technique making use of a population of particles, where the position and velocity of each particle represents a solution to the problem being optimized. The PSO has been shown to be effective in optimising multidimensional discontinuous problems in a variety of fields [8] [11] [35].

Each particle in PSO keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far. This value is called *pbest*. Another "best" value that is tracked by the global version of the particle swarm optimizer is the overall best value and its location obtained so far by any particle in the population. This location is called *gbest*. The PSO concept consists of, in each time step, changing (accelerating) the velocity of each particle flying toward its pbest and gbest locations (global version of PSO). Acceleration is weighted by random terms, with separate random numbers being generated for acceleration toward pbest and gbest locations, respectively.

Similarly to genetic algorithms [16], an evolutionary algorithm approach, PSO is a swarm intelligence optimization tool based on a population, where each member is seen as a particle, and the position and the velocity of each particle is a potential solution to the problem under analysis. Each particle in PSO has a randomized velocity associated to it, which moves through the space of the problem. However, unlike genetic algorithms, PSO does not have operators, such as crossover and mutation. PSO does not implement the survival of the fittest individuals; rather, it implements the simulation of social behaviour.

The global version of PSO algorithm [11] [22] follows the steps show in Algorithm 5.1.

The first part in equation (5.12) is the momentum part of the particle. The inertia weight, w, represents the degree of the momentum of the particles. The use of variable w (inertia weight) was proposed in [34]. This parameter is in charge of dynamically adjusting the speed of the particles, so it is responsible for balancing between local and global search. A low value of inertia weight implies a local search, while a high value leads to a global search. Applying a high inertia weight at the start of the algorithm and making it decay to a

Algorithm 5.1 Particle Swarm Optimization (PSO) Algorithm

Input: c_1, c_2, n, t_{max}

Output: x_i, v_i, p_{best} g_{best}

1. *Initialization*: Initialize a population (array) of particles with random positions and velocities in the n-dimensional problem space using a uniform probability distribution function.
2. *Evaluation*: Evaluate the fitness value of each particle.
3. *Comparison* 1: Compare each particle's fitness with the particle's *pbest*. If the current value is better than *pbest*, then set the pbest value equal to the current value and the pbest location equal to the current location in *n*-dimensional space.
4. *Comparison* 2: Compare the fitness with the population's overall previous best. If the current value is better than *gbest*, then reset *gbest* to the current particle's array index and value.
5. *Updating*: Change the velocity and position of the particle according to eq. (5.12) and (5.13), respectively [34] [35]:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot ud() \cdot (p_i(t) - x_i(t)) + c_2 \cdot Ud() \cdot (p_g(t) - x_i(t)) \tag{5.12}$$

$$x_i(t+1) = x_i(t) + \Delta t \cdot v_i(t+1) \tag{5.13}$$

6. *Stop criterion*: Loop to step (*ii*) until a stop criterion is met, usually a sufficiently good fitness or a maximum number of iterations (generations).

end.

low value through the PSO execution makes the algorithm search globally at the beginning of the search, and search locally at the end of the execution. The following weighting function w is used in eq. (5.12):

$$w = w_{max} - \frac{w_{max} - w_{min}}{t_{max}} t. \tag{5.14}$$

Equation eq. (5.14) shows how the inertia weight is updated, considering tmax is the maximum iteration number, t is the current iteration number, and wmax and wmin are the initial and final weights, respectively.

The second part is the "cognitive" one, which represents the independent behaviour of the particle. In this approach $x_1 = [x_{i1}, x_{i2}, \ldots, x_{in}]^T$ stands for the position and $v_1 = [v_{i1}, v_{i2}, \ldots, v_{in}]^T$ for the velocity of the *i*-th particle; $p_1 = [p_{i1}, p_{i2}, \ldots, p_{in}]^T$ represents the best previous position of the *i*-th particle (the position giving the best fitness value); $t = 1, 2, \ldots, t_{max}$ indicates the iterations. Index g represents the index of the best particle among all the particles in the swarm. Variables $ud(\cdot)$ and $Ud(\cdot)$ are two random functions in the range [0, 1]. Equation (13) represents the position update, according to its previous position and its velocity, considering $\Delta t = 1$.

Positive constants c_1 and c_2 are called cognitive and social components, respectively. These are the acceleration constants responsible for varying the particle speed towards *pbest* and *gbest*. In this paper, the constriction coefficient method is used in PSO based on approach as shown in [10]. In doing so, the velocity equation is updated according to:

$$v_i(t+1) = K \cdot [v_i(t) + c_1 \cdot ud(\cdot) \cdot (p_i(t) - x_i(t)) + c_2 \cdot UD(\cdot) \cdot (p_g(t) - x_i(t))] \tag{5.15}$$

when using a constriction coefficient K,

$$K = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right.} \tag{5.16}$$

with $\varphi = c_1 + c_2$, $\varphi > 4$ and K is a function of c_1 and c_2. Usually, f is set to 4.1 ($c_1 = c_2 = 2.05$), and the constriction coefficient K is 0.729. Other possible choices for the constriction coefficients are available.

5.4 Thermal-Vacuum System for Space Qualification

A block diagram representing the physical operational activity for the thermal-vacuum system is depicted in Figure 5.7. The operation of the thermal-vacuum chamber is described next. A vacuum environment is accomplished by the use of two separate pumping systems, after what the temperature is modified. The first pumping system is a single, dual stage, rotary vane, mechanical pump that produces low pressure inside the chamber. Once the desired pressure is reached, a high vacuum is obtained by using a cryogenic vacuum pump with closed cycle helium compressor. The global system produces pressures around 1×10^{-7} torr to simulate the vacuum present in space. When a satellite is in a vacuum environment, the thermal cycle starts by modifying the temperature inside the shroud. The operation of the thermal shroud is achieved by means of a re-circulating, dense, and gaseous nitrogen (GN2) system. To maintain nearly constant heat transfer properties throughout the wide range of system operation, a constant density system is utilized. Cooling the circulating gas stream is accomplished by spraying liquid nitrogen (LN2) into the circuit while resistance type heaters mounted inside the piping network provide heat as required. The GN2 thermal system is accomplished by using a dual output, time proportioning, heat-cool, and Proportional-Integral-Derivative (PID) controller. The temperature controller sends out setpoints to the GN2 pressure PID controller to keep constant heat transfer characteristics. The system pressure is adjusted to the required level by modifying venting nitrogen gas through the venting control valve (VCV) or by switching the LN2 supply valve (SV1) as can be seen in Figure 5.7 and Figure 5.3.

Inside thermal-vacuum chamber there is no convective heat transfer, since pressure inside thermal-vacuum chamber is low. Convective heat transfer comes up only if vacuum is lost, i.e., internal pressure becomes significant. Furthermore, temperature gradient inside payload may be considered negligible if there is fast heat conduction inside payload. Heat transfer between payload and shroud in vacuum comes from radiation. Radiation heat transfer can be written as:

$$M_P * C_P * \frac{\partial T_p}{\partial t} = \sigma * \varepsilon * A * (T_{sh}^4 - T_p^4) + H_c(P_c) * (T_{sh} - T_p) \tag{5.17}$$

where $\partial T_p / \partial t$ is the payload transition rate, T_{pl} is the payload average temperature (absolute), T_{sh} is the shroud average temperature (absolute), M_{pl} is the payload mass, C_{pl} is the

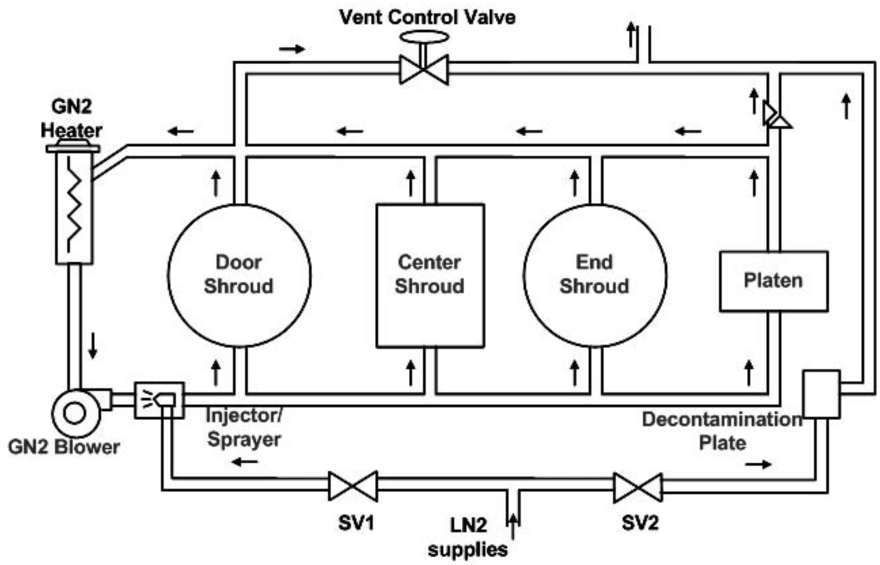

Figure 5.7. Block diagram for physical description of a thermal-vacuum chamber.

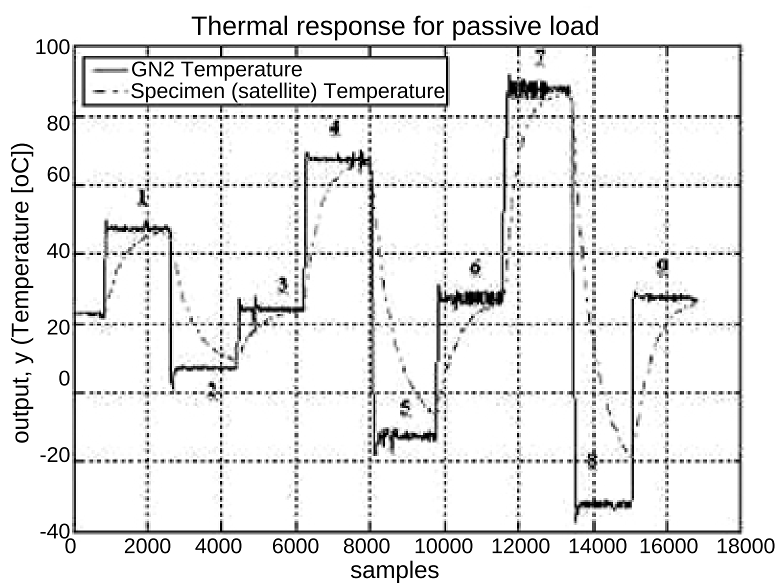

Figure 5.8. Thermal response: passive load.

payload heat capacity, σ is the Stefan-Boltzmann's natural constant, ε is the emissivity / absorptivity of a grey body, and A is the radiating area.

Because radiation is basically the source of heat transfer between the payload and shroud, thermal vacuum chambers are inherently nonlinear. This heat transfer depends nonlinearly on temperature, T^4 as presented in eq. (5.17). When this equation is linearized, it is possible to note that various thermal operational conditions correspond to the reference levels (set points) used during the space product qualification. The modification in dynamics occurs independently of linearization considering stationary behaviour or off-equilibrium nominal trajectory. The time-delay is concerned with the thermal optical characteristics of the specimen undergoing the test as well as its physical characteristics, best described by specific mass, specific heat, and thermal conductivity [19].

In a glimpse, this nonlinear behaviour may be confirmed by real-world industrial dynamical response corresponding to the thermal-vacuum system with passive load (Figure 5.8). Continuous and dashed lines represent, respectively, temperatures in the gas of the shroud and on the satellite. These experimental, measured data are employed to elicit the fuzzy model through PSO approach. A detailed analysis shows that the system presents local and global nonlinearities. This system has different conditions of operations according to diverse reference values. It means that the subsystems individually associate to piecewise input values are nonlinear and that the set of piecewise input references, i.e., the set of piecewise models for the global system are also nonlinear. Since there are nine references, in this example there should be necessary determine nine fuzzy sub-models. This is the case of representing (identifying) the global system through multiple models. This representation may also be understood as a gain-scheduling modelling approach. If linguistic terms and fuzzy logic are employed then this is the case of a fuzzy gain-scheduling model. A solution that fits the features of this system is, thus, to employ piecewise, gain-scheduling fuzzy modelling.

5.5 Identification of T-S Fuzzy System based on PSO Approach

Identification of dynamic systems can be performed with a series-parallel or parallel model. Series-parallel structure is the type of mathematical model adopted for thermal-vacuum system identification when using the hybrid piecewise, gain-scheduling PSO-TS modelling approach as shown in Figure 5.9.

Series parallel model was chosen due to its capability to make one step-ahead forecasting with guaranteed stability of the training procedure. The outputs of current system are used as inputs to the T-S fuzzy model. When a one-time ahead prediction is possible in this case, the T-S model is said to have external dynamics [40].

For dynamic systems, the mathematical model must incorporate time lags, that is, there must have some memory function in the T-S fuzzy model. In T-S fuzzy modelling and in other fuzzy and neural networks approaches, this is performed with delayed inputs and outputs that are employed as extra external inputs.

Assume that there is a T-S fuzzy model that produces an output, $\hat{y}(k+1)$, based on an input $u(k)$ and the noise contribution present in the modelled process, $n(k)$. The estimated T-S fuzzy model output based on PSO, $\hat{y}(k+1)$, used for computing the minimum square error when compared with the actual output, y(k) was computed by using *one-step ahead*

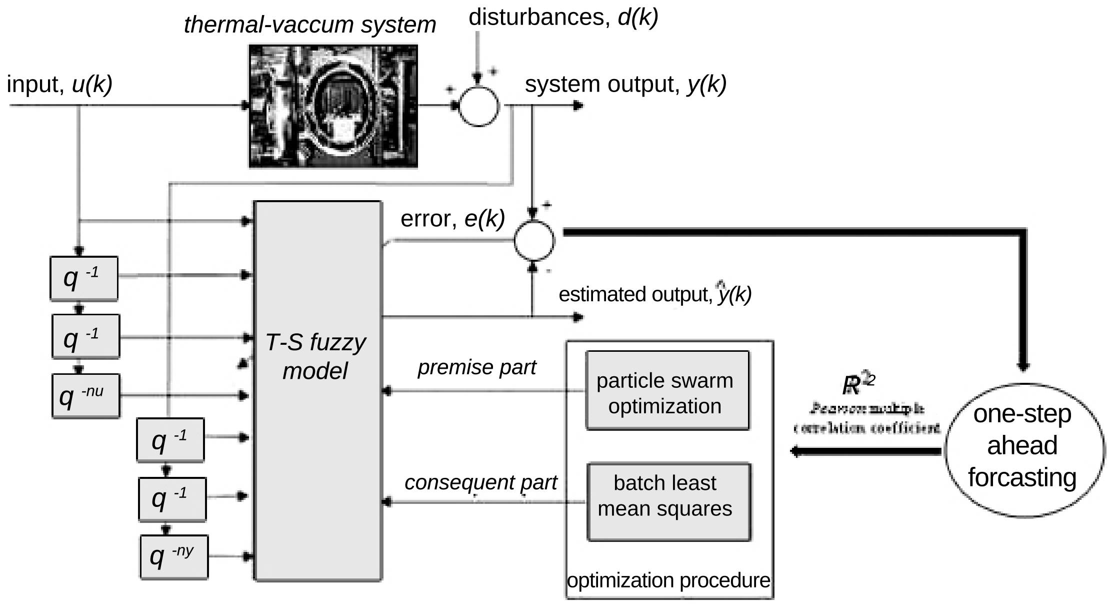

Figure 5.9. Series-parallel identification of thermal-vacuum system using T-S model using PSO and batch least mean squares methods for one-step ahead forecasting.

forecasting. Denote *ny*, *nu*, and *nn* as the time maximum lags of the model output, control input, and noise, respectively. Depending on the time-lagged inputs that are used for the T-S fuzzy model, different configurations of models can be used. In this work, a NARX (Nonlinear Auto Regressive with eXogenous inputs) model was adopted, given by

$$\hat{y}(k) = f_{TS}\left[u(k-1), u(k-2), \ldots, u(k-nu), y(k-1), y(k-2), \ldots, y(k-ny), \theta\right] \quad (5.18)$$

where the unknown nonlinear function f_{TS} is the T-S fuzzy model of the system and k is the time. This function is parameterized by the vector θ - representing the elements a_i, b_i, c_i, in eq. (5.3) and eq. (5.10) and A_i, and B_i, in eq. (5.10) - and depends on premise and consequent parts of T-S fuzzy model.

One of the most important tasks in building an efficient forecasting model based in T-S fuzzy model is the selection of the relevant input variables. The input selection problem can be stated as follows: among a large set of potential input candidates, choose those variables that highly affect the model output. Unfortunately, there is no systematic procedure, currently available, which can be followed in all circumstances [30]. In this work, input selection is heuristically performed. The inputs of T-S fuzzy system are process output and control input signals of reduced order with $ny = 2$, $nu = 1$, and $nn = 0$. In this work, the three vectors of input for the T-S fuzzy system are $[u(k-1); y(k-1); y(k-2)]$ and the model output is $\hat{y}(k)$.

Although, PSO allows to extract the number of rules and to determine the premise and consequent elements, here this method is applied to obtain membership functions and thus to determine the premise space partition. The knowledge and expertise of operators are applied in cooperative approach with experimental input-output data. In doing so, it uses a predefined number of rules and membership functions arbitrarily chosen for each input Figure 5.10.

Setting up this parameter as 3 production rules, PSO needs to deal with a vector of particles positions and velocity whose elements are 9 centres and 3 spreads of a Gaussian function, respectively, core and support of membership functions. In this case, the spread of Gaussian membership function adopted for each input of vectors $[u(k-1); y(k-1); y(k-2)]$ of T-S fuzzy model is the same.

The system identification by T-S fuzzy model is appropriate if a suitable performance index is available according to the necessities of users. Among a population of potential solution to a problem, every particle of PSO has a fitness value for expressing appropriate optimization result. The function representing this quality measure employs the position of all particles, x_i, which is calculated after each iteration.

The performance criterion (fitness function) chosen for evaluate the relationship between the real output and the estimate output during the optimization process was the *Pearson multiple correlation coefficient* index. This Coefficient represents the harmonic mean of R of training and validation phases of T-S fuzzy model conducted by $R^2_{harmonic}$ as given by:

$$R^2_{harmonic} = \frac{2}{\frac{1}{R^2_{training}+\varepsilon} + \frac{1}{R^2_{validation}+\varepsilon}} \tag{5.19}$$

where:

$$R^2_{training} = 1 - \frac{\sum_{k=1}^{0.5Na}[y(k)-\widehat{y}(k)]^2}{\sum_{k=1}^{0.5Na}[y(k)-\bar{y}]^2} \quad R^2_{validation} = 1 - \frac{\sum_{k=0.5Na+1}^{Na}[y(k)-\widehat{y}(k)]^2}{\sum_{k=0.5Na+1}^{Na}[y(k)-\bar{y}]^2} \tag{5.20}$$

are, respectively, the R^2-training (estimation) and R^2-validation phases of the model; e is the small tolerance value (10^{-16}), Na is the total number of samples evaluated, and $\bar{y}$ is the system real output. When $R(\cdot)^2$ is close to unit, $R(\cdot)^2$ = 1.0, a sufficient accurate model for the measured data of the system is found. A R^2 between 0.9 and 1.0 is suitable for applications in identification and model-based control [33].

The main parameters deeply related to the success of PSO for tuning the premise part of T-S fuzzy model are: (*i*) the number of particles (size of population), (*ii*) the initial position and velocity of particles, (*iii*) the cognitive and social components (c_1 and c_2), (*iv*) the form of inertia factor updating, and (*iv*) stopping criterion, $tmax$ (adopted $tmax = 100$ iterations). One of advantage of this technique is that the initial population of particles is randomly generated through a uniform probability distribution function. The sufficient number of particles for this application was setup as 10. The main parameters of PSO approach are shown in Table 5.1 and Table 5.2.

To illustrate the effectiveness of the multi model piecewise, gain-scheduling T-S fuzzy model several simulations were carried out. Distinct results were obtained during the optimization process according to the number of iterations (tmax) employed. Data description and input data used for T-S fuzzy modelling using PSO (for premise part of rules) and batch mean least squares method (for consequent part of rules) is presented in Table 5.3. This optimization process was carried out several times and the best fitness value when using 30 (or 50) iterations was of multiple model approach in terms of cost function precision, conform Table 5.4 and Table 5.5.

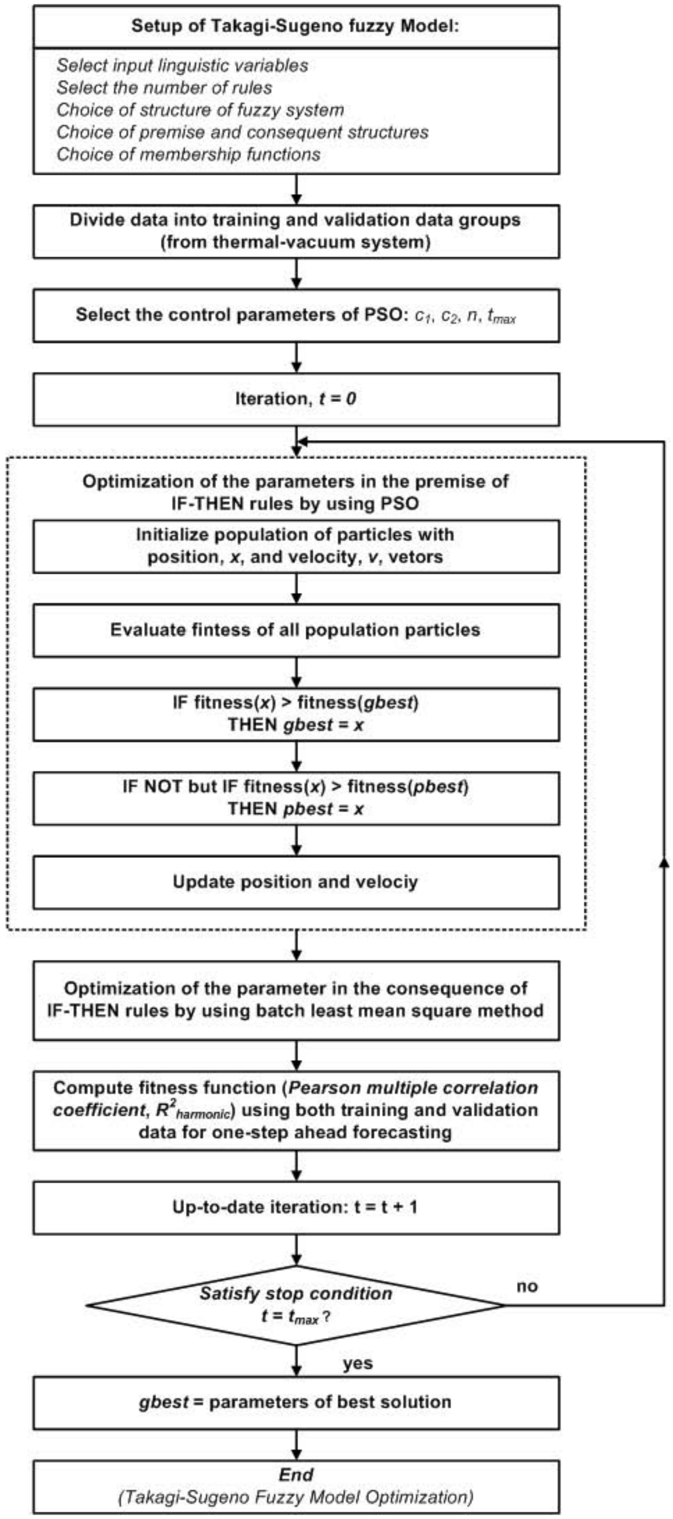

Figure 5.10. Flow chart of PSO algorithm.

Table 5.1. Parameters for PSO application in T-S fuzzy model.

Parameters	Selection
Number of particles	10
Number of iterations*, *tmax*	30 or 50 (see also Table 5.2 and Table 5.3)
Inertia weight setup	constriction factor, $K = 0.729$
Cognitive component	$c_1 = 2.05$
Social component	$c_1 = 2.05$

* *stopping criterion*

Table 5.2. Data description employed in both single and Piecewise, GS T-S fuzzy modelling.

Model	Region	Initial Samples	Final Samples	Steady-state output
	1	200	655	46.30
	2	656	1100	9.33
	3	1100	1545	23.32
	4	1546	2004	65.96
Piecewise	5	2005	2450	-6.62
Model*	6	2451	2895	25.84
	7	2896	3354	86.23
	8	3355	3764	-19.45
	9	3765	4202	25.53
Single Model**	1-9	1	4202	Multiples

* n sub-models according to each reference signal

** one model for all references

Table 5.3. Input data used in both single and Piecewise, GS T-S fuzzy modelling.

Model	Region	Initial Samples (50 % data)	Training Samples (50 % data)	Validation Samples (50 % data)
	1	200	228	227
	2	656	222	222
	3	1100	222	222
	4	1546	229	229
Piecewise Model*	5	2005	223	222
	6	2451	222	222
	7	2896	229	229
	8	3355	205	204
	9	3765	219	218
Single Model**	1-9	200	2001	2001

* n sub-models according to each reference signal

** one model for all references

Table 5.4. Input data for both singular and Piecewise, Gain-scheduling T-S fuzzy modelling by using PSO and mean least squares method (after tmax using PSO).

	Model of Region	$R^2_{harmonic}$	$R^2_{training}$	$R^2_{validation}$	t_{max}
	1	0.998660	0.999913	0.997409	30
	2	0.997624	0.999425	0.995829	30
	3	0.980940	0.994708	0.967547	30
	4	0.993825	0.999354	0.988356	30
Piecewise	5	**0.936217**	**0.994821**	**0.884133**	**30***
Model*	6	**0.963058**	**0.997760**	**0.930689**	**30***
	7	0.998747	0.999970	0.997526	30
	8	**0.947673**	**0.992144**	**0.907017**	**30**
	9	0.969982	0.983553	0.956780	30
	mean	**0.990913**	**0.996629**	**0.985326**	-
Single Model**	1-9	0.990613	0.996818	0.984484	30

** when using 50 iterations were obtained for

region 5 $\rightarrow R^2_{harmonic} = 0.980140$, $R^2_{traning} = 0.993389$, $R^2_{validation} = 0.967240$,

region 6 $\rightarrow R^2_{harmonic} = 0.998876$, $R^2_{traning} = 0.999912$, $R^2_{validation} = 0.997841$,

*** when using 100 iterations were obtained for

region 8 $\rightarrow R^2_{harmonic} = 0.999425$, $R^2_{traning} = 0.999439$, $R^2_{validation} = 0.999411$,

Table 5.5. Comparison of multiple models and single model.

	Model of region	$R^2_{harmonic}$ (multiple model)	$R^2_{harmonic}$ (single model)	$R^2_{training}$ (multiple model)	$R^2_{training}$ (single model)	$R^2_{validation}$ (multiple model)	$R^2_{validation}$ (single model)
	1	0.998660	4.44×10^{-16}	0.999913	0.998432	0.997409	0.000000
	2	0.997624	0.923552	0.999425	0.999449	0.995829	0.858369
	3	0.980940	0.972558	0.994708	0.999267	0.967547	0.947240
	4	0.993825	0.000000	0.999354	0.996003	0.988356	0.000000
Piecewise Model*	5	0.980140	0.956528	0.993389	0.998001	0.967240	0.918365
	6	0.998876	0.637106	0.999912	0.999443	0.997841	0.467588
	7	0.998747	0.000000	0.999970	0.998072	0.997526	0.000000
	8	0.999425	0.961618	0.999439	0.998058	0.999411	0.927744
	9	0.969982	0.000000	0.983553	0.994789	0.956780	0.000000
	mean	**0.990913**	**0.494595**	**0.996629**	**0.997946**	**0.985326**	**0.457700**

Both single T-S fuzzy model and multiple, piecewise fuzzy model fitted the training data with $R^2_{training}$ = 0.983553. Although single PSO-TS fuzzy model achieved a good approximation for experimental data, it has not generalised well to new data for all regions when compared to the proposed multi-model approach.

These models were obtained through PSO by using different sampling rates of data. Continuous and dashed lines represent measured and simulated outputs in Figures 5.11 to 5.16. Experimental results had shown that the hybrid T-S fuzzy system and PSO approaches presented successful results due precision in predicting nonlinear dynamics.

5.6 Summary

Identification of nonlinear systems is a difficult task. Models derived form first principles are usually difficult and/or costly to develop for processes that are not well understood or very complex. Fuzzy identification is an effective tool for the approximation of uncertain nonlinear systems on the basis of measured data. The basic structure of a fuzzy model consists of a rule base, a database and a reasoning mechanism. For this purpose, T-S fuzzy model are widely investigated. T-S fuzzy models use if-then rules to describe the process through a set of locally valid relationships. In this case, the problem of nonlinear system identification is reduced to identification of sub-systems defined over fuzzy input sub-spaces driven by the reference. In order to obtain an optimal approximation the T-S fuzzy model was employed in a cooperative approach with an efficient optimization procedure for premise and consequent part of IF-THEN rules.

Different approaches for eliciting T-S fuzzy models from data have been proposed. In this work is presented a hyperspace search mechanism based on swarm intelligence known as Particle Swarm Optimization (PSO) to find out the premise part of rules of a T-S fuzzy model for a thermal-vacuum system. For T-S fuzzy-like models, parameter optimization techniques of premise part can be chosen independently form each other. The batch mean least squares method is then employed to identify the parameters for consequent part of

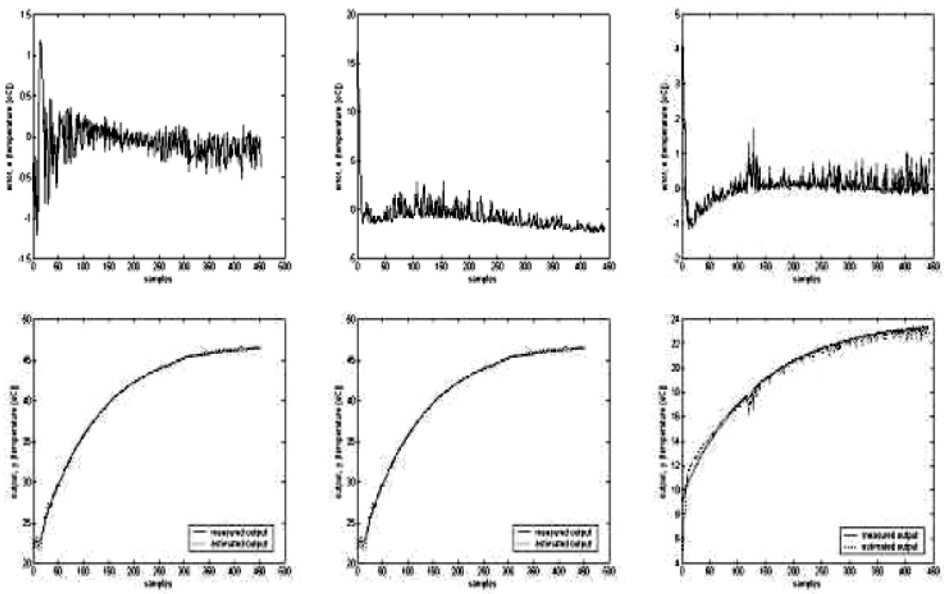

Figure 5.11. Region 1, 2, 3: resulting error (a) and thermal response (b) with piecewise, gain-scheduling model.

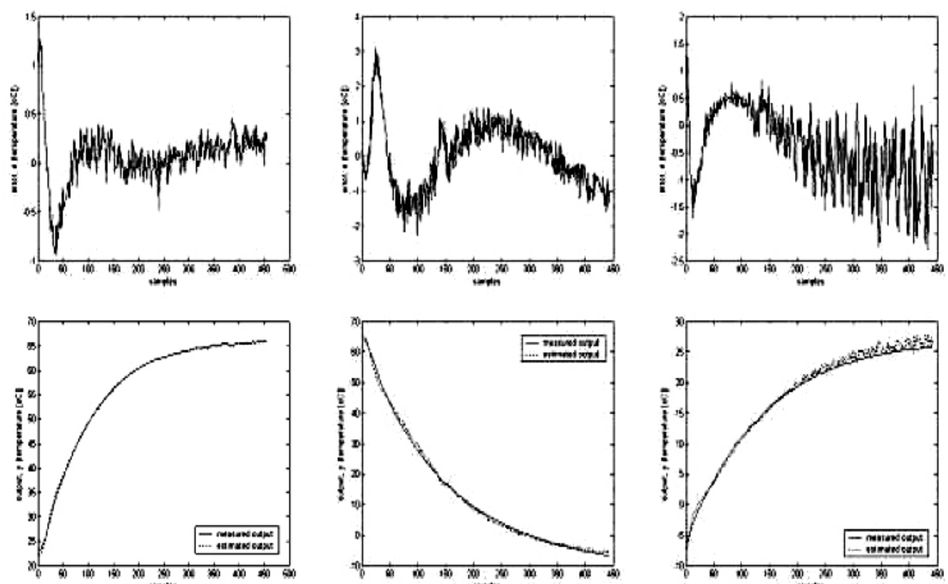

Figure 5.12. Region 4, 5, 6: resulting error (a) and thermal response (b) with piecewise, gain-scheduling model.

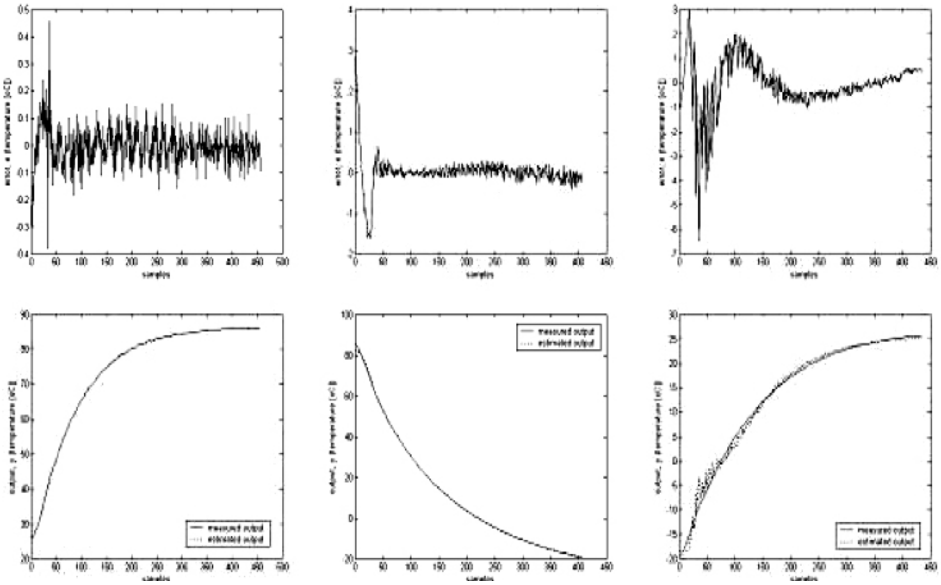

Figure 5.13. Region 7, 8, 9: resulting error (a) and thermal response (b) with piecewise, gain-scheduling model.

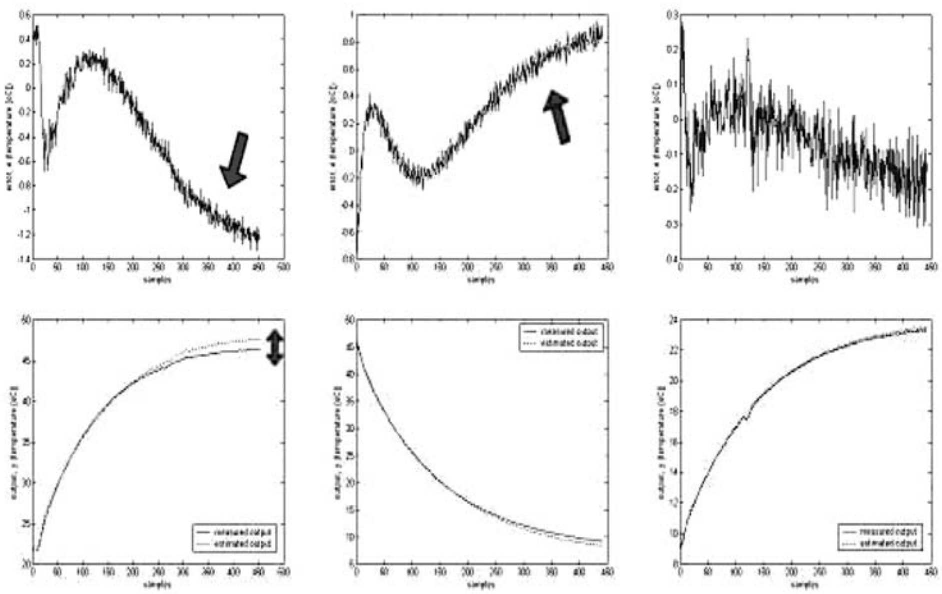

Figure 5.14. Region 1, 2, 3: resulting error (a) and thermal response (b) with general single fuzzy model.

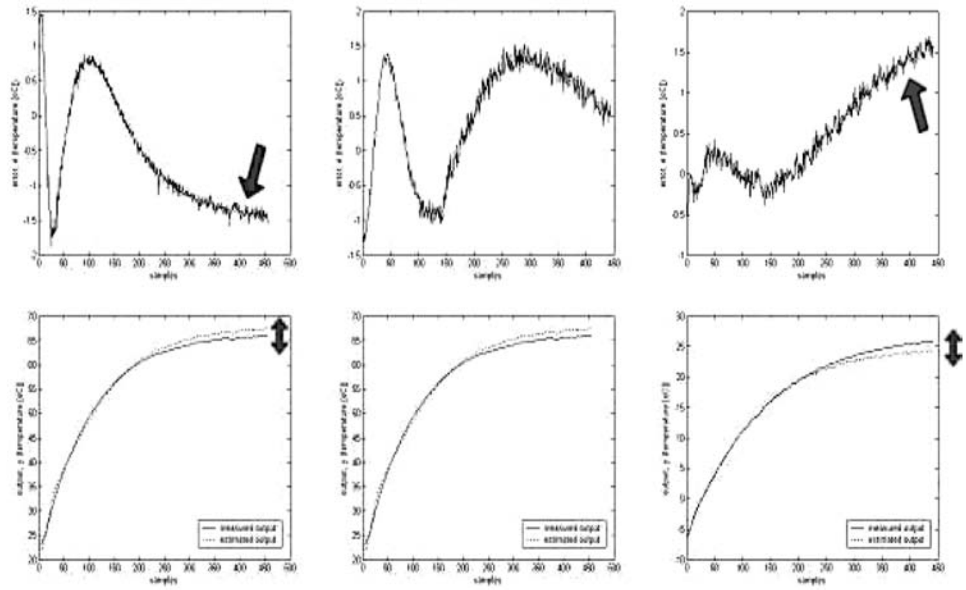

Figure 5.15. Region 4, 5, 6: resulting error (a) and thermal response (b) with general single fuzzy model.

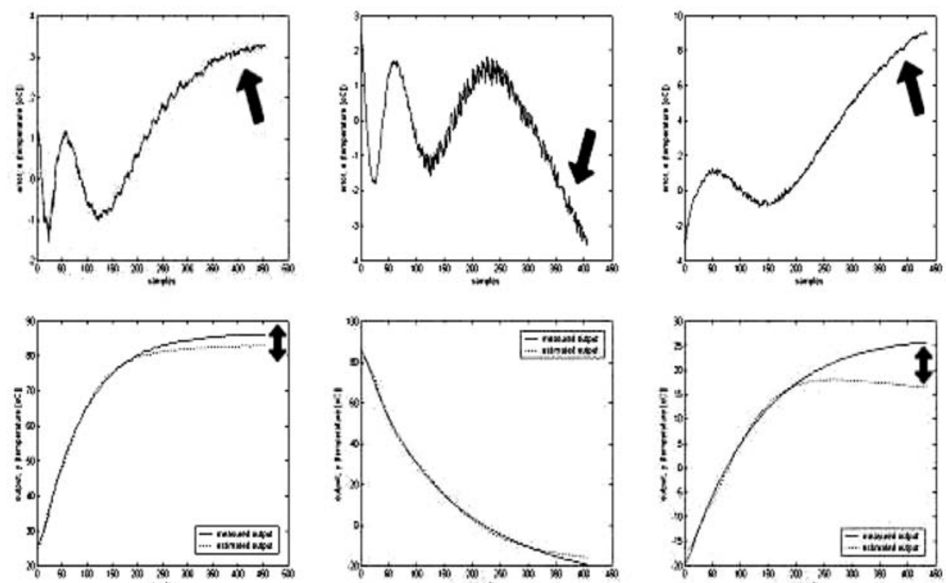

Figure 5.16. Region 7, 8, 9: resulting error (a) and thermal response (b) with general single model.

rules (this problem is linear in the parameters). PSO is a population-based evolutionary algorithm that presents advantages, such as, initial population of particles is randomly generated through uniform distribution, there are no operators such as crossover and mutation in genetic algorithm, simple rules describe complex behaviour, and there are simple code and low computational cost.

Results of piecewise gain-scheduling T-S fuzzy model indicate that the PSO algorithm is a suitable method for tuning T-S fuzzy model for this class of problem. In this work, input of reduced order was tested. The elicited fuzzy model with only three membership functions determining the premise space partition demonstrated its effectiveness in emulating the time response for the thermal-vacuum system. The resulting models exhibit a number of desirable characteristics such as accurate and robust capability for one step-ahead forecasting.

According to the suitable results obtained it must be of interest to explore the influence of other parameters in obtaining the model. For example, the number of membership functions may be increased, and since there is large amount of data other step ahead forecasting modes may be exploited. As well, results has shown that the proposed piecewise, gain-scheduling fuzzy T-S model seem adequate to be applied to control synthesis by employing Parallel Distributed Controller (PDC) or Fuzzy Reference Gain-Scheduling (FRGS) controller. The idea behind PDC to generate fuzzy controllers is to design compensator for each rule of the fuzzy model. Since the method employed in this paper supplies multiple models, each sub-model is assumed to be an appropriate model for each fuzzy controller synthesis generating a piecewise, gain-scheduling T-S fuzzy controller. FRGS approach, in turn, is a fuzzy controller synthesis in which the parameters of the controller (support and core) change according to exogenous signals, such as, multiple goals, changes in the environment, or diverse context. Since the identification process supplied fuzzy sub-models whose membership functions change their shape and distribution in the universe of discourse according to the reference, each sub-model would generate a sub-controller in which their membership functions fit their support and core to accommodate on-line and real-time changes in the dynamics of fuzzy models.

References

[1] Araujo, J.E., Sandri, S.A, Macau, E.E.N., Fuzzy Gain Scheduling Control Systems, *Proc. 19th International Meeting of the North America Fuzzy Information Processing Society*, pp. 461-464, Atlanta, 2000.

[2] Araujo, J.E., Sandri, S.A, Macau, E.E.N., A New Class of Adaptive Fuzzy Control System applied in Industrial Thermal Vacuum Process, *Proc 8th IEEE International Conference on Emerging Technologies and Factory Automation*, v.1, pp. 426-431, France, 2001

[3] Araujo, J.E., Kienitz, K.H., Adaptive Reference-Driven Decision-Making Process, *Proc FUZZ-IEEE International Conference on Fuzzy Systems*, v.1, pp.452-457, St. Louis, 2003.

[4] Babuska, R., Verbruggen, H.B., An overview of Fuzzy Modelling for Control, *Control Eng. Practice*, v.4, n.11, pp.1593-1606, 1996.

[5] Barada, S., Singh, H., Generating Optimal Adaptive Fuzzy-Neural Models of Dynamical Systems with Applications to Control, *IEEE Trans. on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, v. 28, n.3, 1998.

[6] Bertolissi, E., Birattari, M., Bontempi, G., Duchteau, A. and H. Bersini, Data-driven techniques for direct adaptive control: the lazy and the fuzzy approaches, *Fuzzy Sets and Systems*, vol. 128, pp. 3-14, 2002.

[7] Boukhris, A., Mourot, G., Ragot, J. Non-linear dynamic system identification: a multi-model approach, *International Journal of Control*, vol. 72, no. 7/8, pp. 591-604.

[8] Brandsttter, B. and U. Baumgartner, Particle swarm optimization - mass-spring systems analogon, *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 997-1000, 2002.

[9] Chen, L. and K. S. Narendra, Nonlinear adaptive control using neural networks and multiple models, *Automatica*, vol. 31, pp. 1245-1255, 2001.

[10] Clerc, M. and J. Kennedy, The particle swarm - explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58-73, 2002.

[11] Coelho, L. S. and R. A. Krohling, Predictive controller tuning using modified particle swarm optimisation based on Cauchy and Gaussian distributions, *Soft Computing: Methodologies and Applications*, Springer Engineering series in Advances in Soft Computing, pp. 287-298, 2005.

[12] Dickerson, J.A., Kosko, B., Fuzzy Funciton Approximation with Ellipsioidal Rules, *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, v. 26, n.4, 1996.

[13] Eberhart, R. C. and J. Kennedy, A new optimizer using particle swarm theory, *Proceedings of International Symposium on Micro Machine and Human Science*, Japan, pp. 39-43, 1995.

[14] Garner, J. T., *Satellite control - a comprehensive approach*, New edition, John Wiley & Sons Ltd. and Praxis Publishing Ltd., Chinchester, 1996.

[15] Gilmore, D. G., *Satellite thermal control handbook*, The Aerospace Corporation Press, California, 1994.

[16] Goldberg, D. E. *Genetic Algorithms in search, optimization, and machine learning*, Addison Wesley: Reading, MA, 1989.

[17] Haber, R. and H. Unbehauen, Structure identification of nonlinear dynamic systems - a survey on input/output approaches, *Automatica*, vol. 26, no. 4, pp. 651-677, 1990.

[18] High Vacuum Systems Inc. (HVSI), *"Operations and Maintenance Manual - 250 dm3 Thermal Vacuum System with Thermally Conditioned Platen"*, October, 1997.

[19] Jen, C.-L. and L. Tilwick, On-line, self-learning, predictive toll for determining payload thermal response, *Proceedings of Space Simulation*, lugar???, pp. 193-200, 2000.

[20] Johansen, T. A ., Identification of non- linear systems using empirical data and prior knowledge - an optimization approach, *Automatica*, vol. 32, pp. 337-356, 1996.

[21] Kennedy, J., Eberhart, R.C. and Y. Shi, *Swarm intelligence*, Morgan Kaufmann Pub. San Francisco, 2001.

[22] Krohling, R. A., Hoffmann, F. and L. S. Coelho, "Co-evolutionary particle swarm optimization for min-max problems using Gaussian distribution", *Proceedings of the Congress on Evolutionary Computation*, Portland, OR, USA, pp. 959-964, 2004.

[23] Leith, D.J., Leithead, W.E., "Gain-Scheduled Controller Design: an analytic framework directly incorporating non-eqilibrium plant dynamics", *Int. J. Control*, v.70, n.2, pp. 249-269, 1998.

[24] Leith, D.J., Leithead, W.E., "Gain-Scheduled and nonlinear systems: Dynamic Analysis by Velocity-Based linearization families", *Int. J. Control*, v.70, n.2, pp. 289-317, 1998.

[25] Leith, D.J., Leithead, W.E., "Survey of Gain and Design", *Int. J. Control*, v.70, n.2, pp. 289-317, 1998.

[26] Liu, Y. and K. Passino, Swam intelligence: literature overview, Ohio University, USA, *Technical Report* March 30, 2000.

[27] Ljung, L. *System identification: theory for the use*. Prentice-Hall: New York, 2nd edition, 1999.

[28] Mao, K. Z., Billings, S. A. and Q. M. Zhu, A regularized rational model estimation algorithm, *International Journal of Systems Sicence*, vol. 30, no. 5, pp. 455-465, 1999.

[29] Marinke, R., Araujo, E., Coelho, L. S. and I. Matiko, Particle swarm optimization (PSO) applied to fuzzy modelling in a thermal-vacuum system, *International Conference on Hybrid Intelligent Systems*, Rio de Janeiro, Brazil, (pre-print), 2005.

[30] Mastorocostas, P. A., Theocharis, J. B. and A. G. Bakirtzis, Fuzzy modeling for short term load forecasting using the orthogonal least squares method, *IEEE Transactions on Power Systems*, vol. 14, no. 1, pp. 29-36, 1999.

[31] Rugh, W.J., "Analytical Framework for Gain Scheduling", *IEEE Trans. on Control Systems Magazine*, v.11 n.1, pp. 79-84, January, 1991.

[32] Rugh, W.J., Shamma, J.S., "Research on gain scheduling", *Automatica*, v.36, n.10, pp. 1401-1425, Oct, 2000.

[33] Schaible, B., Xie, H. and Y. C. Lee, Fuzzy logic models for ranking process effects, *IEEE Transactions on Fuzzy Systems*, vol. 5, no. 4, pp. 545-556, 1997.

[34] Shi, Y. and R. C. Eberhart, Parameter selection in particle swarm optimizer, *Proceedings of Annual Conference on Evolutionary Programming*, V. W. Porto, N. Saravan, D. Waagen, and A. E. Eiben (editors), Berlin: Springer-Verlag, pp. 591-601, 1998.

[35] Shi, Y. and R.C. Eberhart, Fuzzy adaptive particle swarm optimization, *Proceedings of the Congress on Evolutionary Computation*, Honolulu, Hawaii, v.1, pp. 101-106, 2002.

[36] Sugeno, M. and G. T. Kang, Structure identification of fuzzy model, *Fuzzy Sets and Systems*, vol. 28, pp. 15-33, 1988.

[37] Takagi, T. and M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no.1, pp. 116-132, 1985.

[38] Tan, S., Hang, C. -C. and J.-S. Chai, Gain-scheduling: from conventional to neuro-fuzzy, *Automatica*, vol. 33, no. 3, pp. 411-419, 1997.

[39] Thomson, M. and S. P. Scholing, The practical application of a nonlinear identification methodology, *Control Engineering Practice*, vol. 4, no. 3, pp. 295-306, 1996.

[40] Van Gorp, J., Nonlinear identification with neural networks and fuzzy logic, *Thesis*, Vrije Universiteit Brussel, Faculteit Toegepaste Wetenschappen, Brussel, Belgium, 2000.

[41] Wang, L.-X. and J. M. Mendel, Generating fuzzy rules by learning from examples, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 6, 1992.

[42] Zhao, J., Wertz, V. and R. Gorez, Fuzzy gain scheduling controllers based on fuzzy models, *Proceedings of IEEE International Conference on Fuzzy Systems*, New Orleans, USA, vol. 3, pp. 1670-1676, 1996.