Interdisciplinary Description of Complex Systems 14(2), 223-235, 2016

FUZZY-GENETIC CONTROL OF QUADROTOR UNMANNED AERIAL VEHICLES

Attila Nemes*

Óbuda University, Doctoral School of Safety and Security Sciences
Budapest, Hungary

DOI: 10.7906/indecs.14.2.11
Regular article

Received: 16 February 2016.
Accepted: 8 March 2016.

ABSTRACT

This article presents a novel fuzzy identification method for dynamic modelling of quadrotor unmanned aerial vehicles. The method is based on a special parameterization of the antecedent part of fuzzy systems that results in fuzzy-partitions for antecedents. This antecedent parameter representation method of fuzzy rules ensures upholding of predefined linguistic value ordering and ensures that fuzzy-partitions remain intact throughout an unconstrained hybrid evolutionary and gradient descent based optimization process. In the equations of motion the first order derivative component is calculated based on Christoffel symbols, the derivatives of fuzzy systems are used for modelling the Coriolis effects, gyroscopic and centrifugal terms. The non-linear parameters are subjected to an initial global evolutionary optimization scheme and fine tuning with gradient descent based local search. Simulation results of the proposed new quadrotor dynamic model identification method are promising.

KEY WORDS

quadrotor UAV, fuzzy system, unconstrained optimization, genetic algorithms, gradient descent search

CLASSIFICATION

ACM: G.1.6.

JEL: Z19

PACS: 45.40.-f,87.19.lu

*Corresponding author, η: mr.attila.nemes@gmail.com; +36 30 63 77 830;
2612 Kosd, Mikes Kelemen u. 38, Hungary

INTRODUCTION

A wide area of robotics research is dedicated to aerial platforms. The quadrotor architecture has low dimensions, good manoeuvrability, simple mechanics and payload capability.

The study of kinematics and dynamics helps to understand the physics of the quadrotor and its behaviour. Together with modelling, the determination of the control algorithm structure is very important [1-6]. The quadrotor unmanned aerial vehicle (UAV) is controlled by angular speeds of four motors. Each motor produces a thrust and a torque, whose combination generates the main trust, the yaw torque, the pitch torque, and he roll torque acting on the quadrotor. Motors produce a force proportional to the square of the angular speed and the angular acceleration; the acceleration term is commonly neglected as the speed transients are short thus exerting no significant effects. Soft computing methods can be efficiently applied together with and also instead of conventional controllers.

Fuzzy modelling [7-12] can be conducted as black-box modelling where all the system knowledge is mere input-output data, however when expert knowledge is readily available, we should take advantage of it – fuzzy grey-box modelling is a rational choice. Identification of linear parameters is a well-studied area, with efficient matrix algebra and singular value decomposition based reliable tools. Non-linear parameters can also be simply traced to their local optimum with well-studied gradient descent methods, but we should always keep in mind that gradient descent methods are trapped by local optimum areas. Evolutionary algorithms are robust global optimum search engines, capable of multi-objective search as described in [13-16].

The article is organized as follows. In Section 1 the Introduction is given, in Section 2 the quadrotor dynamic model is presented. In Section 3 the Fuzzy-logic systems are illustrated. In Section 4 the multi-objective Genetic algorithms are illustrated. Section 5 presents the simulation setup and simulation results. Conclusions are given in Section 6.

QUADROTOR DYNAMIC MODEL

Motors of a quadrotor can only turn in a fixed direction, so the produced force is always positive. Motors are set up so that two opposites form a pair, which turns clockwise, while the other pair rotates counter-clockwise. This arrangement is chosen so that gyroscopic effects and aerodynamic torques are canceled in trimmed flight [17-20].

The main trust is the sum of individual trusts of each four motor. The pitch torque is a function of difference in forces produced on one pair of motors, while the roll torque is a function of difference in forces produced on other pair of motors. The yaw torque is sum off all four motor reaction torques due to shaft acceleration and blades drag. The motor torque is opposed by a general aerodynamically drag.

The complete dynamics of an aircraft, taking into account aero-elastic effects, flexibility of wings, internal dynamics of the engine, and the whole set of changing environmental variables is quite complex and somewhat unmanageable for the purpose of autonomous control engineering.

For a full dynamic model of a quadrotor system both (1) the center of mass position vector of (x, y, z) in fixed frame coordinates and (2) the orientation Euler angles: roll, pitch, yaw angles (Φ, θ, ψ) around body axes X, Y, Z are considered for the vector of generalized coordinates q. Using the Euler-Lagrange approach it can be shown how the translational forces $\mathbf{F}_\xi$ applied to the rotorcraft due to main trust can be full decoupled from the yaw, pitch and roll moments For a full dynamic model of a quadrotor system both (1) the center of mass position vector of (x, y, z) in fixed frame coordinates and (2) the orientation Euler angles: roll, pitch, yaw angles

(ϕ, θ, ψ) around body axes X, Y, Z are considered for the vector of generalized coordinates q. Using the Euler-Lagrange approach it can be shown how the translational forces $\mathbf{F}_\xi$ applied to the rotorcraft due to main trust can be full decoupled from the yaw, pitch and roll moments $\boldsymbol{\tau}$:

$$m\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} + mg\begin{bmatrix}0\\0\\1\end{bmatrix} = \boldsymbol{F}_\xi, \tag{1}$$

where m is the quadrotor mass and g is the gravitational constant.

$$\boldsymbol{J}\begin{bmatrix}\ddot{\boldsymbol{\phi}}\\ \ddot{\boldsymbol{\theta}}\\ \ddot{\boldsymbol{\psi}}\end{bmatrix} + C\left(\begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\theta}\\ \boldsymbol{\psi}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\theta}\\ \boldsymbol{\psi}\end{bmatrix}\right)\begin{bmatrix}\dot{\boldsymbol{\phi}}\\ \dot{\boldsymbol{\theta}}\\ \dot{\boldsymbol{\psi}}\end{bmatrix} = \boldsymbol{\tau}, \tag{2}$$

where $\boldsymbol{J}$ is a 3 × 3 matrix, called the inertia matrix and C is also a 3 × 3 matrix that refers to Coriolis, gyroscopic and centrifugal terms. Further on, for the scope of this article we shall address only equation (2) as the quadrotor dynamic model to be identified.

Equation (2) can be analyzed as three resultant torques τ_i acting along the i^{th} axes respectively as $i \in (\phi, \theta, \psi)$, which using Christoffel symbols of the first kind can be defined as a function of the state vector of Euler angles $\boldsymbol{q} = (\phi, \theta, \psi)$, their velocities ($\dot{q} = dq/dt$) and accelerations ($\ddot{q} = d\dot{q}/dt$) as:

$$\sum_{j}\left(D_{ij}(q)\cdot\ddot{q}\right) + \sum_{j,k}\left(\dot{q}_j \cdot D_{ijk}(q)\cdot\dot{q}_k\right) = \tau_i, \quad i, j, k = 1, 2, 3. \tag{3}$$

The first component of equation (3) is shortly referred to as $J\ddot{q}$ the inertia matrix part, while the second as $C\dot{q}$ the Coriolis matrix term for which components are defined as:

$$J_{ik} = D_{ik}(q), \quad C_{ik} = \sum_{j=1}^{p}\dot{q}_j \cdot D_{ijk}(q), \quad i, j, k = 1, 2, 3. \tag{4}$$

Where D_{ik}, D_{ijk} are in general, highly non-linear scalar functions of the state vector q. They contain sin(˙) and cos(˙) functions of q, and their products and sums defined by the geometry of the system.

There are general relations that can be used for reducing the number of unknown elements of $\boldsymbol{J}$ and C, like: (1) $\boldsymbol{J}$ is symmetric and (2) D_{ijk} are Christoffel-symbols of D_{ij}, thus further properties are inherently defined as:

$$D_{ik} = D_{ki},\ D_{ijk} = \frac{1}{2}\left(\frac{\partial D_{ij}}{\partial q_k} + \frac{\partial D_{ik}}{\partial q_j} - \frac{\partial D_{jk}}{\partial q_i}\right),\ D_{ijk} = D_{ikj},\ D_{kij} = -D_{jik},\ D_{kjk} = 0, \quad \forall\, i,k \geq j. \tag{5}$$

It should be noted that direct measurement of any single component from equation (4) is not possible; the only measurable data, on the output of the system, is the resultant torque of equation (3).

Identification of all non-linear functions (4) under these terms is a considerable problem.

FUZZY –LOGIC SYSTEMS

Takagi-Sugeno-Kang (TSK) type Fuzzy-logic systems (FLSs) having n inputs and 1 output are defined as:

$$f(q) = \frac{\sum_{l=1}^{M}\omega_l(q)\cdot y_l(q)}{\sum_{l=1}^{M}\omega_l(q)}, \tag{6}$$

where M is the number of rules, q is the vector of n input variables, y_l is a scalar function of n input variables, defined by $(n + 1)\cdot c$ parameters as in equation (8). The antecedent, the premise part of a fuzzy rule is:

$$\omega_l(q) = \prod_{i=1}^{n} \mu_{F_{l(i)}}(q_i)\,, \tag{7}$$

where $\mu_{F_{l(i)}}(q_i)$ is the membership function (MF) of the i^{th} input variable in the l^{th} rule that defines the linguistic value $F_{l(i)}$. The linguistic form of the l^{th} rule from the previously described first order TSK FLS is defined in [13] as:

$$\text{IF } (q_1 \text{ is } F_{l(1)}) \text{ AND } (q_2 \text{ is } F_{l(2)}) \text{ AND } \ldots (q_n \text{ is } F_{l(n)}) \text{ THEN } y_l = \sum_{j=1}^{n} c_{l(j)} \cdot q_j + c_{l(0)}\,, \tag{8}$$

Zadeh-formed MFs are the z-, the s-, and π-functions (named after their shape) defined respectively as:

$$mfz(q,b_1,b_2) = \begin{cases} 1 & q \le b_1 \\ 1-2((q-b_1)/(b_2-b_1)) & b_1 < x \le \tfrac{1}{2}(b_2+b_1) \\ 2((q-b_1)/(b_2-b_1)) & \tfrac{1}{2}(b_2+b_1) < q \le b_2 \\ 0 & q > b_2 \end{cases}$$

$$mfs(q,b_1,b_2) = 1 - mfz(q,b_1,b_2) \tag{9}$$

$$mf\pi(q,b_1,b_2,b_3,b_4) = \begin{cases} mfs(q,b_1,b_2) & q \le b_2 \\ 1 & b_2 < q \le b_3 \\ mfz(q,b_3,b_4) & q > b_3 \end{cases}$$

where $b_1 \le b_2 \le b_3 \le b_4$ are parameters defining MFs. In case there is more than one value q such that the degree of membership of q is equal to one, the interval where $\mu_k(q, b) = 1$ (the interval $[b_2, b_3]$ for $mf\pi$ type μ_k) is called the plateau of the μ_k MF. When having for example 3 naturally ordered linguistic values $l \in \{a, b, c\}$ (*a* ↔ *low*, *b* ↔ *medium*, *c* ↔ *large*) constraints on parameters to preserve this ordering are:

$$\begin{aligned} &b_{a1} < b_{b1} < b_{c1}, \\ &b_{a2} \le b_{a3} < b_{b2} \le b_{b3} < b_{c2} \le b_{c3}, \\ &b_{a4} < b_{b4} < b_{c4}. \end{aligned} \tag{10}$$

A linguistic variable can be assigned K different linguistic values, each described by a MF $\mu_k(q,b)$ such that for every input x it holds that $\sum_{k=1}^{K} \mu_k(q,b) = 1$, the MFs are said to form a fuzzy-partition. Forming fuzzy-partitions by antecedent membership functions ensures that there can not be a numerical input within the defined input range that will not result in firing at least one rule consequent of the fuzzy model, which means that there is a defined output for all possible input states. Keeping specific properties of fuzzy-partitions imposes a set of hard constraints on membership function parameters as detailed in [15]. By imposing these restrictions on all linguistic variables of the FLS, and additionally assuming that the rule base is complete in the sense that it covers the whole input domain, it immediately follows that the TSK model structure of equation (6) simplifies to:

$$f(q) = \sum_{l=1}^{M} \omega_l(q) \cdot y_l(q)\,. \tag{11}$$

Automatic fine tuning FLS parameters that satisfies all of above listed constraints is a significant problem.

In [15] a method is introduced that simplifies parameter optimisation of equation (11) while preserving all required constraints. Fuzzy-partitions can be simply formed from Zadeh-typed MFs by making equal the last two parameters of each preceding MF to the first two parameters of the succeeding MF. This way a fuzzy partition of K MFs is defined by $2(K-1)+1$ parameters. Let our input space be normalised ($x_{min} = 0$ and $x_{max} = 1$). If we do not want to allow any plateaux, parameter b_2 must be equal to b_3 in equation (9) this way the number of parameters is further reduced to $K-2$. When we take into consideration all constraints of equation (10), we end up with a series of strictly ordered parameters:

$$b_1 < b_2 < \ldots < b_{K-1}. \tag{12}$$

Let us add two more constraints:

$$0 < b_1 \text{ and } b_{K-1} < 1. \tag{13}$$

Let us define the first MF to be:

$$mfz(x, 0, b_1). \tag{14}$$

Let the K-th, the last MF concluding the fuzzy partition be:

$$mfs(x, b_{K-2}, 1). \tag{15}$$

Let us define intermediate kth MFs to be:

$$mf\pi(x, b_{k-1}, b_k, b_k, b_{k+1}) \tag{16}$$

for $k = 1, \ldots, K-2$, where $b_0 = 0$ and $b_{K-1} = 1$. This way the ordered series of $K-2$ parameters b_i, together with constants 1 and 0, are the minimal number of parameters to define a fuzzy-partition of Zadeh-formed MFs. This minimal number of non-linear parameters is a very important issue for optimisation as over parameterised systems are hard to optimise. The only problem is that when tuning the non-linear parameters of a FLS having an n dimensional input space, we must comply with $\sum_{i=1}^{n} K_i$ pieces of hard constraints. Although there are a number of constrained optimisation methods it is obvious that an unconstrained optimisation method would be more efficient. Let us consider $K-1$ pieces of rational, positive or zero parameters as proposed in [12]:

$$a_\kappa \in R_0^+, \ \ \kappa = 1, \ldots, K-1. \tag{17}$$

When we define b_k as:

$$b_k = \sum_{j=1}^{k} a_j \Big/ \sum_{\kappa=1}^{K-1} a_\kappa \, . \tag{18}$$

for every $k = 1, \ldots, K-2$; all the constraints of equation (12) and equation (13) are automatically fulfilled for every a_κ from equation (18) without any further restrictions on any a_κ, other than $0 \le a_\kappa$. An ANFIS like optimisation, defined in [16] or any other efficient unconstrained nonlinear numerical method can be applied to minimise equation (11) error along the a_κ parameters. For calculating all linear parameters a linear least square (LS) method can be applied to $c_{l(j)}$ parameters of the consequent part. To avoid traps of local optimal solutions for a_κ, a preliminary global search should be applied.

MULTI-OBJECTIVE GENETIC ALGORITHMS

A genetic algorithm (GA) is constructed on bases of imitating natural biological processes and Darwinian evolution [21-24]. GAs are widely used as powerful global search and optimization tools [25]. Real life optimization problems often have multiple objectives. To establish ranking of chromosomes for Gas the comparison of two objective vectors is required. Often a simple weighted sum is used, but its drawbacks are widely known. Pareto based comparison [19] is the bases of a few popular methods like Non-dominated Sorting GA (NSGA) [22] and Multi-Objective GA (MOGA) [23, 25]. A general multi-objective optimization problem consists of n number of scalar minimization objectives where every scalar objective function $f_i(x)$ is to be minimized simultaneously, where x is an n-dimensional vector of parameters. As maximization can be easily transformed to minimization, the generality of the previous statement stands. A vector x_1 Pareto-dominates x_2, when no scalar component of x_2 is less than the appropriate component of x_1, and at least one component of x_1 is strictly smaller than the appropriate component of x_2. Since no metrics can be assigned to Pareto-dominance, there have been two different attempts to define a GA ranking method, which can be used for Pareto-dominance vector comparison: (1) "Block-type" ranking is defined in [23] as: *Rank* is equal to 1 + (*number of individuals that dominate the* i^{th} *individual*); (2) "Slice-type" ranking is defined in [5] as: *Rank* is equal to 1 + (*number of turns when the non-dominated individuals are eliminated, needed for the* i^{th} *individual to become non-dominated*).

Quantity-dominance, as proposed in [15] is defined as: vector $a = [a_i]$. Quantity-dominates vector $b = [b_i]$ if a has more such a_i components, which are better than the corresponding b_i component of vector b, and a has less such a_j components, which are worse than the corresponding b_j. A metrics can be defines as: the measurement of the extent of Quantity-dominance is the difference between the number of better and the number of worse components. For a measurement based ranking method the *Rank* of the i^{th} objective vector can be simply defined as *the sum of Quantity-dominance measurements for every individual measured from the* i^{th} *vector*. This ranking method can be readily applied with Quantity-comparison. The proposed vector comparison method provides more information when comparing two vectors than the classic Pareto-based comparison, thus the GA is faster, more efficient in its search. The MMNGA algorithm is computationally less expensive, and more efficient compared to the classical methods, and its results analyzed on a number of GA hard problems are at least equally good [16].

In case of multi-rotors the roll and pitch are equal to:

$$\phi = \operatorname{asin} \frac{\dot{x}\sin\psi - \dot{y}\cos\psi}{\ddot{x}^2 + \ddot{y}^2 + (\ddot{z}+g)^2}, \quad \theta = \operatorname{atan} \frac{\dot{x}\cos\psi - \dot{y}\sin\psi}{\ddot{z}+g}. \tag{19}$$

From equations (3) and (19) it is obvious that controll torques for multi rotors are direct functions of up to the forth time derivatives of state variables (x,y,z) and ψ. To have realistic, feasible torques along a trajectory, which are efficiently controllable without chattering, we need smooth torque changes. Having $\boldsymbol{\tau} = \boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$ and $\boldsymbol{q} = \boldsymbol{q}(\psi, \dot{\boldsymbol{r}}, \ddot{\boldsymbol{r}})$ for smooth torque changes, we need smooth changes of the so called displacement crackle $\boldsymbol{c}(t) = d^5\boldsymbol{r}/dt^5$, the fifth time derivative of displacement $\boldsymbol{r}(x, y, z)$. Proposal of this article is to use a smooth displacement crackle function, which can be defined with a continuous displacement pop function $\boldsymbol{p}(t) = d^6\boldsymbol{r}/dt^6$ as:

$$\boldsymbol{p}(t) = G\frac{\sin(\omega_d \cdot t + \sigma_d)}{\omega_d}, \quad c(t) = \int \boldsymbol{p}(t)\boldsymbol{dt} = G[1 - \cos(\omega_d \cdot t + \sigma_d)]. \tag{20}$$

where ω_d is the natural dampened frequency from equation (3), σ_d the phase delay is kept is zero, and the gain G is selected for each trajectory and system such that the required boundaries for displacement, velocity and acceleration limits are met. The integration constant for $\boldsymbol{c}(t)$ is to be selected as equal to G to achieve the required properties for the cracle function; for all further integrations to calculate trajectory snap, jerk, acceleration, velocity and displacement by intergrating $\boldsymbol{c}(t)$ we are to use integration constants equal to zero. The resulting general trajectory plot is as presented in Figure 1.

We can eficiently identify D_{ij} inertia matrix components of the dynamic model in equation (4) as FLSs defined by equations (11) to (18), where the FLS general input variable q will be substituted for the appropriate state variables of (ϕ, θ, ψ). When the D_{ij} inertia matrix components are constructed in this way, forming the D_{ijk} components as Christoffel symbols is to be expressed by partial derivatives of equation (11) :

$$\frac{\delta f(q)}{\delta q_i} = \sum_{l=1}^{M} \left(\frac{\delta \omega_l(q)}{\delta q_i} \cdot y_l(q) + \omega_l(q) \cdot \frac{\delta y_l(q)}{\delta q_i} \right). \tag{21}$$

The unknown inertia matrix components of equation (2) to be identified are:

$$D_{13}(\theta), \quad D_{22}(\phi), \quad D_{23}(\phi, \theta), \quad D_{33}(\phi, \theta). \tag{22}$$

Based on quadrotor system structure and inertia matrix symmetry the remaining inertia components are known to be:

$$D_{11} = I_X, \quad D_{12} = 0, \quad D_{21} = D_{12}, \quad D_{31} = D_{13}, \quad D_{32} = D_{23}. \tag{23}$$

Based on equation (5) the following Coriolis term matrix components can be calculated by equations (22):

$$\begin{aligned} &D_{122} = -\frac{1}{2}\frac{\delta D_{22}}{\delta \phi}, \quad D_{123} = \frac{1}{2}\left(\frac{\delta D_{13}}{\delta \theta} - \frac{\delta D_{23}}{\delta \phi}\right), \quad D_{322} = \frac{\delta D_{23}}{\delta \theta}, \\ &D_{133} = -\frac{1}{2}\frac{\delta D_{33}}{\delta \phi}, \quad D_{223} = -\frac{1}{2}\frac{\delta D_{33}}{\delta \theta}, \quad D_{312} = \frac{1}{2}\left(\frac{\delta D_{23}}{\delta \phi} + \frac{\delta D_{13}}{\delta \theta}\right). \end{aligned} \tag{24}$$

Remaining D_{ijk} components are trivial identities defined by equation (5).

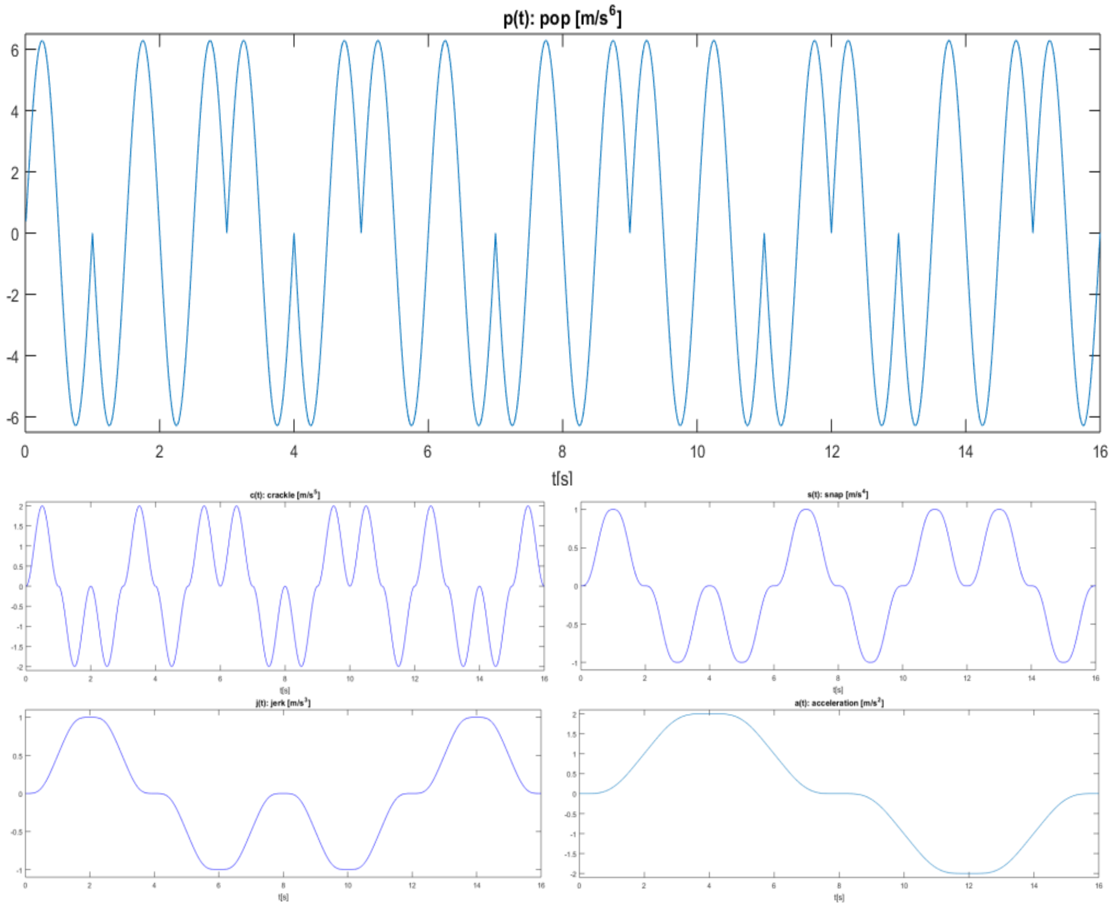

Figure 1. Trajectory pop $\boldsymbol{p}(t)$, crackle $\boldsymbol{c}(t)$, snap $\boldsymbol{s}(t)$, jerk $\boldsymbol{j}(t)$, acceleration $\boldsymbol{a}(t)$, velocity $\boldsymbol{v}(t)$ and displacement $\boldsymbol{r}(t)$.

SIMULATION SETUP AND RESULTS

The proposed method is tested for a quadrotor system simulation from [1] with following parameters: gravity constant g = 9,81 m/s^2, mass m = 6 kg, trust factor k = 121,5 e^{-6}, drag factor b = 2,7e^{-6}, body inertia along axes X, I_X = 0,6 kg·m^2, body inertia along axes Y, I_Y = 0,6 kg·m^2, body inertia along axes Z, I_z = 0,6 kg·m^2, simulation time t = 3 s. The training data set is collected from a simulation along a trajectory with jounce for (x,y,z) and ψ defined so that position changes simultaneously along a unit cube main diagonal (0 0, 0)-(1, 1, 1), while performing a full circle rotation in jaw motion 0-2π.The simulated resultant torque training data is as presented in Figure 2. The calculated roll, pitch and yaw motions are presented in Figs. 3-5.

Non-linear a_K parameters of the system are identified in a manner that first the input space is normalised to the unit hyper-cube. A set of non-linear parameters consists of six times four a_K integer parameters for defining six fuzzy-partitions of five MFs each, where each partition consists of one z-type MF, three π-type MFs and one s-type MF as in (9)-(18). These six fuzzy-partitions serve as antecedents for the four fuzzy-systems like in equation (11) and (21), used for identifying D_{ij}, with ij = (13, 22, 23, 33) as defined in equations (22)-(24) and (5). Two unknown linear parameters D_{11} and D_{12} of the quadrotor model as in equation (23), together with 170 linear parameters of the four TSK FLSs (2 FLSs with 5 MFs on one input, each rule with 2 c parameters, plus 2 FLSs with 5 MFs on both of the 2 inputs, each rule with $3 \cdot c$ parameters) of equations (22) and equations (24) are determined by the SVD-based LS method as used in [15]. Concluded from equation (17) six fuzzy-partitions (antecedent part of 2 FLSs with 1 input, plus 2 FLSs with 2 inputs are covered by 6 independent fuzzy-partitions) are represented by a vector of six times four a_K parameters, which are optimized by a multi-objective hybrid genetic algorithm as detailed in [16]. Each chromosome evaluation is extended to include an additional round of nonlinear LSQ optimization of a_K parameters. Chromosomes are updated before applying further GA operators, so the GA does not waste time on local optimization; only global search capabilities of the GA are utilized. Three objectives are set for minimization: (1) the root mean square of the torque identification error, (2) the maximum absolute error for any given training data input-output pair, and (3) the condition number of the linear system of equations used for LS calculation of linear parameters. The GA is set to work on a population of 125, divided into 5 subpopulations with migration rate 0,2 taking place after each 5 completed generations. Crossover rate, generation gap and insertion rate is 0,8, selection pressure is 1,5. In each generation 4 % of individuals are subject to mutation, when 1 % of the binary genotype is mutated. Individuals, chromosomes are comprised of 24 Gray-coded integers, each consist of 16 bits. The initial population is set up in a completely random manner.Matrix of the linear equation is pre-processed from equation (3), where FLSs like equation (11) and their partial derivatives like equation (21) are inserted as described in equations (22)-(24). Unknown linear parameters are D_{11}, D_{12} and the 170 c parameters of fuzzy-rule consequents.

Evaluation of each individual is done as follows: (1) Convert the coded a_K values from the chromosome to b_k by equation (18). (2) Evaluate all MFs, which will comprise six fuzzy-partitions from each of six b_k quadruplets by equations (14)-(16). Also evaluate derivatives of equations (14)-(16). (3) The pre-processed matrix of the linear equation is evaluated with the MFs. (4) Linear components of equations (11) and (21) are calculated by SVD decomposition as described in [20]. (5) Next the a_K parameters are fine-tuned by the Matlab “lsqnonlin” function, (6) MFs are re-calculated for the optimised a_K parameters and all linear parameters are re-calculated. (7) Resulting optimised a_K parameters are re-assigned into the chromosome of the evaluated chromosome.

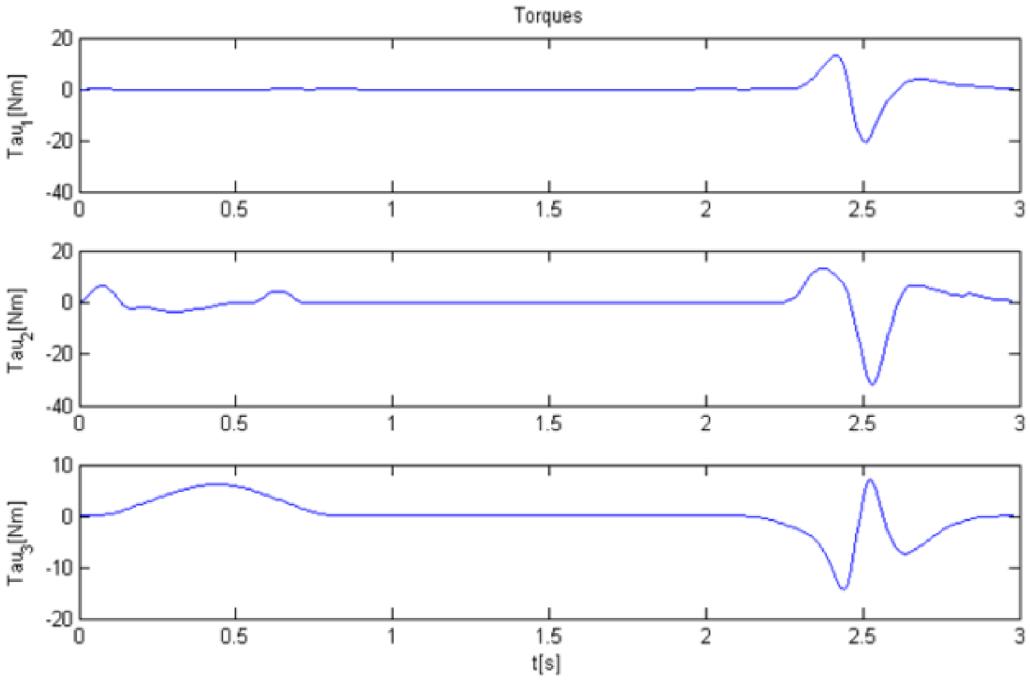

Figure 2. Torque training data set for otput.

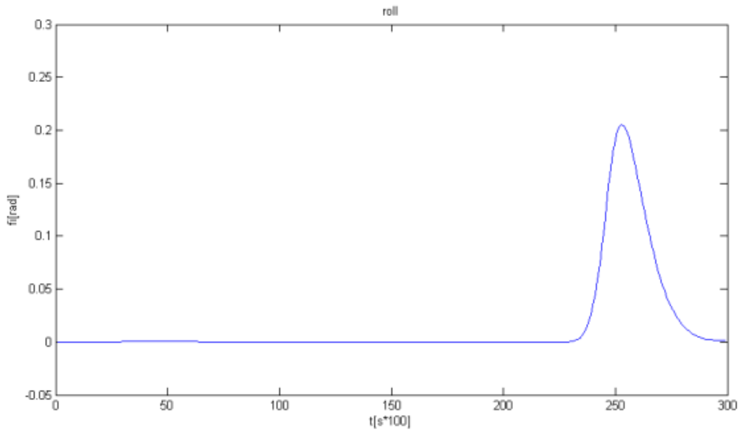

Figure 3. Roll training data for input.

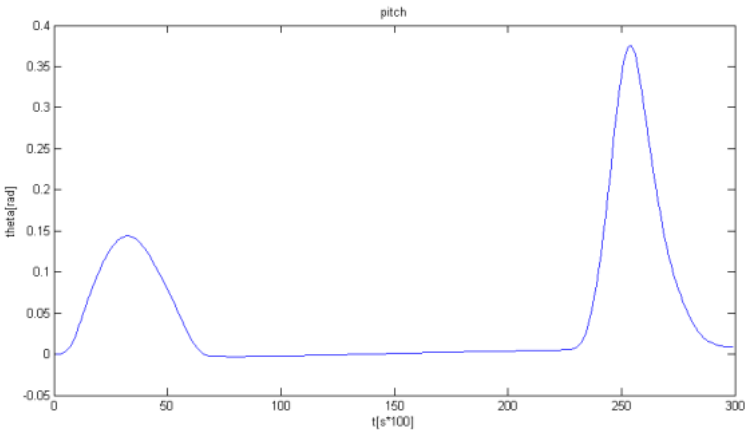

Figure 4. Pitch training data for input.

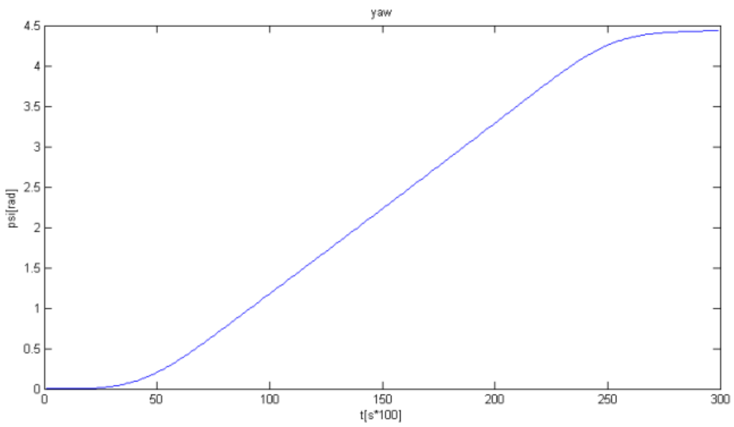

Figure 5. Yaw training data for input.

For the multi-objective rank assignment described in [16], the objective vector is created from: (i) the mean square of the identified torque error, (ii) the maximum absolute torque identification error and (iii) the condition number of the matrix of the linear equation. Stochastic universal sampling is used for selecting the next generation without explicit elitism. To speed up the GA processing, a database of evaluated chromosomes and their objective vectors is created, so only unique new individuals are evaluated in each generation.
Convergence is achieved in some 50 generation evaluations, when the mean square error is in order of $5e^{-7}$, the maximum torque error is smaller than 0,005 Nm. For non-dominated chromosomes the condition number of the used matrix of linear equation is in order of $1 \cdot e^{+38}$.
One typical non-dominated chromosome and the corresponding torque identification error are presented on figures 6 to 10. The numerical value of this chromosome is: [61029 8550 10175 18348 6668 22470 11993 57404 608 18024 25310 39946 26698 53573 39807 47476 1909 46 52007 47288 3712 920 50956 5174], which defines fuzzy-partition MF parameters as: b_i for J_{13}: [0,6221, 0,7093, 0,8130]. b_i for J_{22}: [0,0677, 0,2957, 0,4174]. b_i for J_{23}: [0,0072, 0,2221, 0,5238; 0,1593, 0,4791, 0,7167]. b_i for J_{33}: [0,0189, 0,0193, 0,5330; 0,0611, 0,0762, 0,9148].

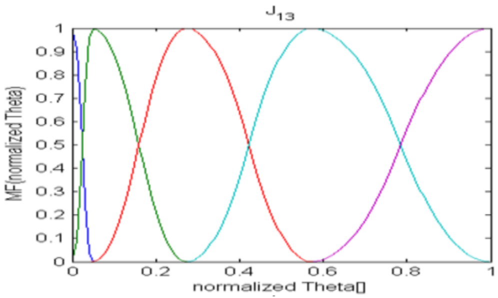

Figure 6. Fuzzy-partition for J_{13} antecedents.

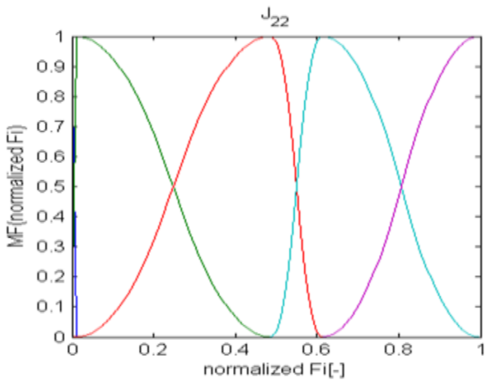

Figure 7. Fuzzy-partition for J_{22} antecedents.

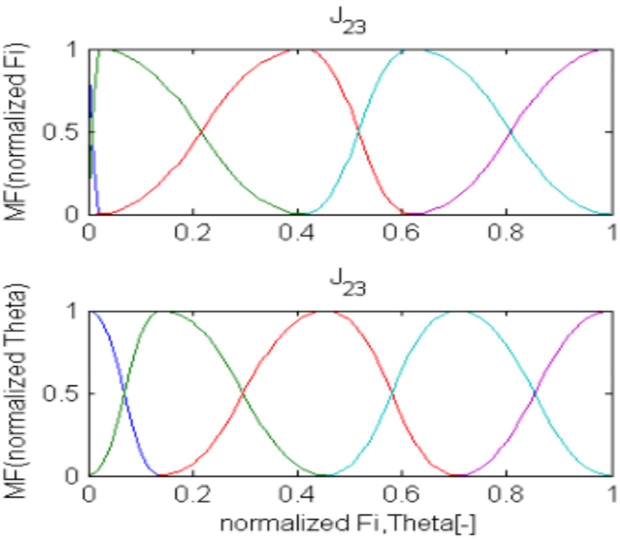

Figure 8. Fuzzy-partition for J_{23} antecedents.

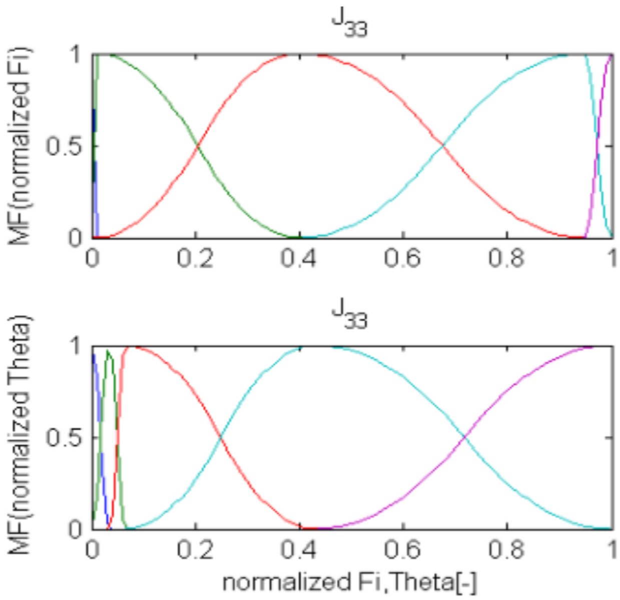

Figure 9. Fuzzy-partition for J_{33} antecedents.

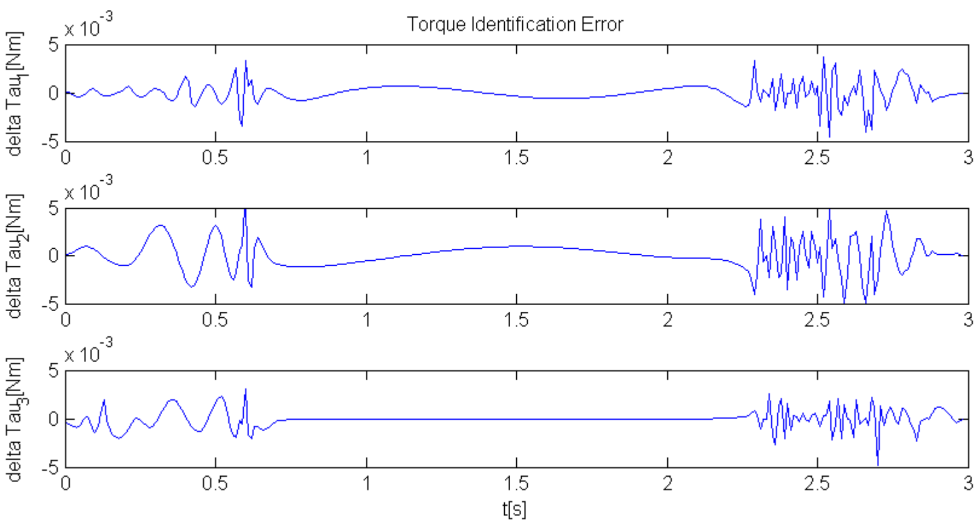

Figure 10. Torque identification error.

CONCLUSIONS

Simulation results of the proposed new quadrotor dynamic model identification method are promising. The quality of identification with the relative torque error being uniformly <0.1% is excellent, suitable for taking part in a model based control algorithm. The typical condition number for used linear parameter evaluations is very high for the used training data setup, so a more advanced trajectory has to be planned with sufficient inertia excitation along the complete input domain. Also the FLS structure is to be made flexible, in terms that the GA should be able to turn off unnecessary MFs and thus reduce the number of unnecessary rules and linear parameters.

REFERENCES

[1] Lozano, R.: *Unmanned Aerial Vehicles*.
ISTE Ltd, London, 2010,

[2] Stengel, R.: *Flight Dynamics*.
Princeton University Press, Cloth, 2004,

[3] Nemes, A.: *Genetic Algorithm-Based Adaptive Fuzzy Logic Systems for Dynamic Modeling of Quadrotors*.
Proceedings of the 3rd International Conference MechEdu. Subotica, pp.96-103, 2015,

[4] Rodic, A. and Mester, G.: *Sensor-based Navigation and Integrated Control of Ambient Intelligent Wheeled Robots with Tire-Ground Interaction Uncertainties*.
Acta Polytechnica Hungarica, Journal of Applied Sciences **10**(3), 113-133, 2013,

[5] Nemes, A.: *Dynamic Modelling of Robot Manipulators by Zadeh-type Fuzzy Partitions*.
4th International Symposium of Hungarian Researchers on Computational Intelligence, Budapest, 2003,

[6] Rodic, A. and Mester, G.: *Virtual WRSN – Modeling and Simulation of Wireless Robot-Sensor Networked Systems*.
Proceedings of the 8th IEEE International Symposium on Intelligent Systems and Informatics, SISY 2010, pp.115-120, Subotica, 2010,

[7] Mester, G.: *Modeling of the Control Strategies of Wheeled Mobile Robots*.
Proceedings of The Kandó Conference 2006, pp.1-3, Budapest, 2006,

[8] Mester, G.: *Introduction to Control of Mobile Robots*.
Proceedings of the YUINFO’2006, pp.1-4, Kopaonik, 2006,

[9] Mester, G.: *Distance Learning in Robotics*.
Proceedings of The Third International Conference on Informatics, Educational Technology and New Media in Education, pp.239-245, Sombor, 2006,

[10] Mester, G.: *Intelligent Mobile Robot Controller Design*.
Proceedings of the Intelligent Engineering Systems, INES 2006, pp.282-286, London, 2006,

[11] Mester, G.: *Improving the Mobile Robot Control in Unknown Environments*.
Proceedings of the YUINFO’2007, pp.1-5, Kopaonik, 2007,

[12] Mester, G.: *Adaptive Force and Position Control of Rigid-Link Flexible-Joint Scara Robots*.
Proceedings of the International Conference on Industrial Electronics, Control and Instrumentation, 20th Annual Conference of the IEEE Industrial Electronics Society IECON'94, Vol. 3, pp.1639-1644, Bologna, 1994,

[13] Nemes A.: *New Genetic Algorithms for Multi-objective Optimisation*.
1st International Symposium of Hungarian Researchers on Computational Intelligence, Budapest, 2000,

[14] Nemes, A.: *Function Identification by Unconstrained Tuning of Zadeh-type Fuzzy Partitions*.
2nd International Symposium of Hungarian Researchers on Computational Intelligence, Budapest, 2001,

[15] Nemes, A.: *Synopsis of Soft Computing Techniques Used in Quadrotor UAV Modelling and Control*.
Interdisciplinary Description of Complex Systems **13**(1), 15-25, 2015,
http://dx.doi.org/10.7906/indecs.13.1.3,

[16] Nemes, A.: *System Identification Based on Multi-Objective Optimisation and Unconstrained Tuning of Zadeh-type Fuzzy Partitions*.
Proceedings IEEE SISY, 2003,

[17] Stepanić, J.; Kasać, J. and Lesicar, J.: *What is Taken for Granted about Quadrotors: Remarks about drive and communication*.
Proceedings of the 3rd International Workshop on Advanced Computational Intelligence and Intelligent Informatics, IWACIII 2013, pp.1-4, Shanghai, 2013,

[18] Mester, G.: *Intelligent Mobile Robot Motion Control in Unstructured Environments.*
Acta Polytechnica Hungarica, Journal of Applied Sciences **7**(4), 153-165, 2010,

[19] Mester, G. and Rodic, A.: *Autonomous Locomotion of Humanoid Robots in Presence of Mobile and Immobile Obstacles.*
Studies in Computational Intelligence, Towards Intelligent Engineering and Information Technology, Part III Robotics, Volume 243/2009, pp. 279-293, Springer, 2009,

[20] Rubóczki, E.S. and Rajnai, Z.: *Moving towards Cloud Security*.
Interdisciplinary Description of Complex Systems **13**(1), 9-14, 2015,
http:dx.doi.org/10.7906/indecs.13.1.2,

[21] Srinivas, N. and Deb, K.: *Multiobjective Optimisation Using Nondominated Sorting in Genetic Algorithms*.
Evolutionary Computation **2**(3), 221-248, 1994,

[22] Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*.
Addison Wesley Publishing Company, 1989,

[23] Fonseca, C.M. and Fleming, P.J.: *Multiobjective Optimisation and Multiple Constraint Handling with Evolutionary Algorithms I: A Unified Formulation*.
Technical Report 564, University of Sheffield, Sheffield, 1995,

[24] Hellendron, H. and Driankov, D.: *Fuzzy Model Identification*.
Selected Approaches. Springer, 1997,

[25] Mester, G.; Pletl, S.; Nemes, A. and Mester, T.: *Structure Optimization of Fuzzy Control Systems by Multi-Population Genetic Algorithm*.
Proceedings of the 6th European Congress on Intelligent Techniques and Soft Computing, EUFIT'98, Vol. 1, pp.450-456, Aachen, 1998.