Australian Journal of Basic and Applied Sciences, 8(4) Special 2014, Pages: 340-344

AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN:1991-8178

Journal home page: www.ajbasweb.com

Designing the 6-DOF Massive Parallel Arrays with Artificial Intelligence Control

[1]Felix Pasila, [2]Roche Alimin

[1]*Electrical Engineering Department, Petra Christian University,Siwalankerto 121-131, Surabaya 60236, Indonesia.*
[2]*Mechanical Engineering Department, Petra Christian University,Siwalankerto 121-131, Surabaya 60236, Indonesia.*

ARTICLE INFO

Article history:
Received 20 November 2013
Received in revised form 24 January 2014
Accepted 29 January 2014
Available online 5 April 2014

Key words:
Two-state force actuators, inverse static analysis (ISA); artificial intelligence; binary-discrete state manipulators (b-DSMs); 10-binary MPRs

ABSTRACT

Binary-Discrete State Manipulators (b-DSMs) are force regulated manipulators that undergo continuous motions despite being commanded through a finite number of states only. Designing a real-time control of such systems requires fast and efficient methods for solving their inverse static analysis (ISA), which is a challenging problem of this paper. In particular, an artificial intelligence method based on neuro-fuzzy method is proposed to investigate the on-line computation and the generalization error of ISA problem of a class of b-DSMs featuring two-state force actuators and six degree of freedom. The main advantages of a neuro-fuzzy system for b-DSMs are: it interprets IF-THEN rules from input-output relations (orientation, moment and binary state) and focuses on accuracy of the output network and offers efficient time consumption for on-line computation. The paper proposed two architectures which are based on the Neuro-Fuzzy Takagi-Sugeno (NFTS) inference scheme with Gaussian membership functions. They are NFTS and the Look-Up Table version of NFTS, which is called as NFLUT . Both structures are with multivariate input and multi-state outputs, such as orientations and moments as input networks and binary state of the b-DSMs as output networks. The learning procedure uses an accelerated LMA with optimal training parameters with at least half-million iterations with different 10 membership functions, employ 12% of the input-output correspondences from the known input-output dataset. For experimental database,the NF structure is tested using 1024 dataset. The optimized membership function (N) after two weeks searching time using Hill Climbing (HC) procedure is N = 17 for the 10-binary Massive Parallel Robots (MPRs). Regarding model performances for the ISA solution, the NFLUT features better generalization ability compared to the NFTS model but requires a rather larger computational time during on-line testing phas.

To Cite This Article: Felix Pasila, Roche Alimin, Designing the 6-DOF Massive Parallel Arrays with Artificial Intelligence Control. ***Aust. J. Basic & Appl. Sci.,*** *8(4): 340-344, 2014*

INTRODUCTION

This paper proposes an efficient way to control the massive binary-Discrete State Manipulator (b-DSMs) via the real-time Artificial Intelligence controller. The b-DSMs are a very special kind of mechanisms whose actuators can only be made switching between two states (extended or retract, +1 or -1). Moreover, the b-DSMs are a kind of manipulators, in an effort to consider sensor-less manipulators as well as to reduce the control procedure and complexity of computer interfacing. Currently b-DSMs can be classified into two different groups depending on whether their actuators act as discrete displacement generators or discrete force generators. Examples of b-DSM of the first type are the binary snake-like robots (SLRs), proposed by Chirikjian *et al* (1994, 1995, 1997, 2001) and Dubowsky *et al* (2001, 2002), which are kinematically constrained mechanisms employing a large number of bi-stable actuators whose configuration either fully contracted (inactive state) or fully extended (active state) without consideration of the arbitrary external forces acting on them. Examples of b-DSM of the second type are the binary Massively Parallel Robots (MPRs) (Waldron *et al.*, 2001a, 2001b), which are dynamically constrained robots employing a large number of on-off actuators that employ either a constant force (active state) or no force (inactive state).

Corresponding Author: Ahmad Joraimee Mohamad, Faculty of Electrical and Automation Engineering Technology, TATI University College.
E-mail: joraimee@tatiuc.edu.my

To achieve high position/force capabilities (both in terms of variation range and accuracy), the design of SLRs/MPRs practically requires a large number of actuators (at least 4-8 times larger than the number of degrees of freedom desired for the robot) that can be arranged in a hybrid series-parallel configuration (prevalently in-series for SLRs (Chirikjian *et al*., 1997, 2001; Dubowsky *et al*., 2001, 2002) whereas in-parallel for MPRs (Waldron *et al*., 2001a, 2001b).

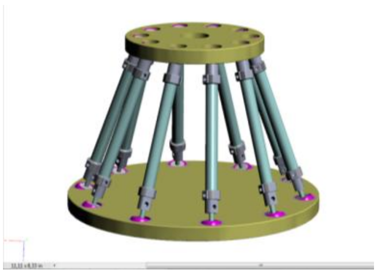
The paper considers b-DSM type MPRs model, where the inverse static analyses (ISA) of MPRs are usually very difficult problems whose solution practically requires quite complicated processes. In the past, significant research efforts have been devoted to address these inverse problems, in particular by resorting to: exhaustive brute-force search approaches (Waldron *et al*., 2001a, 2001b); methods of classical differential geometry and variation of calculus (Chirikjian, 1995, 1997); combinatorial heuristics algorithms (Dubowsky *et al*., 2001, 2002); genetic algorithms (Dubowsky *et al*., 2002); probability theory (Chirikjian *et al*., 2001); high-gain Hopfield networks and Boltzmann machines (Waldron *et al*., 2001a, 2001b). Even though most of the proposed solution schemes are formally very elegant and quite effective in reducing problem complexity from exponential time to polynomial time, the resulting algorithms still involve too many calculations for real-time manipulator control.

2. Artificial Intelligence control based on Neuro-Fuzzy Method:

In this paper, we investigate the potentialities of using artificial intelligence algorithm based on neuro-fuzzy (NF) method for the real-time solution of the ISA problem that feature six degree of freedom actuated by a number of in-parallel-placed two-state force generators. The NF is a hybrid intelligent system which combines the human-like reasoning style of fuzzy systems with the learning capability of neural networks. The main advantages of a NF system are: it interprets IF-THEN rules from input-output relations and focuses on accuracy of the output network; and it has an efficient time consumption for on-line computation. In the field of artificial intelligence, NF refers to combinations of artificial neural networks and fuzzy logic. This idea was proposed first by J. S. R. Jang (1993) and later was improved Palit et.al. (2002; 2005). NF is a hybrid intelligent system, which combines the human-like reasoning style of fuzzy systems with the learning ability of neural networks. In the following section, we proposed two NF models which are based on the Neuro-Fuzzy Takagi-Sugeno (NFTS) inference scheme with Gaussian membership functions. They are NFTS and the Look-Up Table version of NFTS, which is called as NFLUT. Concerning the ISA problem, both proposed models can be applied as solutions because they provide a strong connection between input values $\boldsymbol{X}$ with their output variables ternary number $\boldsymbol{u} = (u_1,\ldots, u_{10})$.

3. B-DSMs Mechanism:

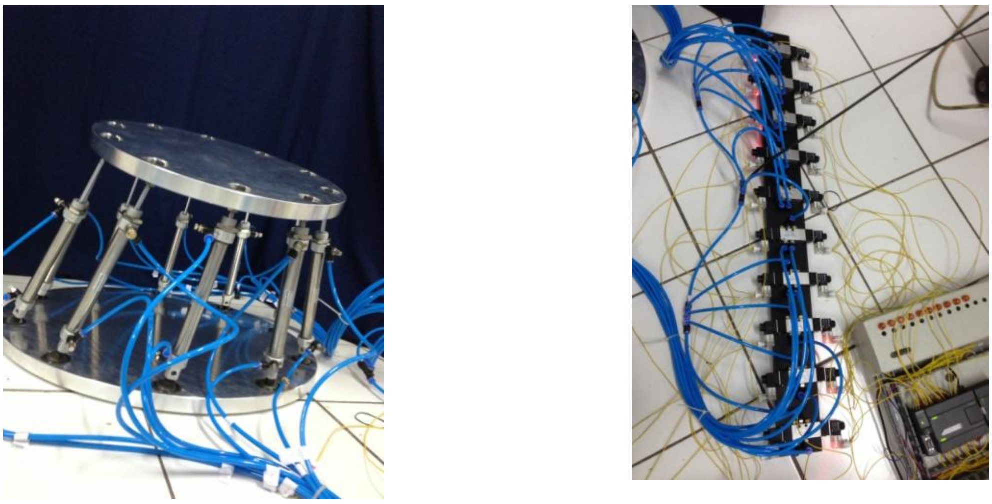
In this Section, we discuss the b-DSMs mechanism that is considered in this paper, as depicted in Figs. 1 and 2. It features 10 identical Crank and Slotted-Lever (CSL) respectively with 10-SPS 3D mechanism. The terms S, P and S are for spherical, prismatic and spherical joint respectively, sharing the same moving platform at their moving joint. The moving platform is hinged at the based platform at point O, the m links with variable length Ai-Bi, where i =1, 2, …, m; here m = 10, are hinged at the common based platform points Ai and at moving platform points Bi respectively, symmetrically located with respect to the XYZ axis along the both platform with radius r.



(a)

(b)

Fig. 1: First design(a) and implementation(b) of b-DSMs with 10 actuators



(a) (b)

Fig. 2: Implementation b-DSMs (a); with valves condition and programmable logic controller (b); personal computer (matlab, not shown in the Fig.)

Practical implementation of the mechanisms, as depicted in Fig. 1, could be obtained by employing ten double-effect pneumatic cylinders with directional control valves in place of both slider and slotted-lever links. Fig. 2 shows the b-DSMs mechanism connected with 10 valves 5/2. The Valves are connected also with Siemens S7-200 and Personal Computer (PC). In the PC, Matlab program run the neuro-fuzzy methods by the given input of b-DSMs and produce the state outputs of the actuators. These outputs will be delivered to the PLC via Serial Communication.

4. Neuro-Fuzzy Architecture:

This Section presents the architecture of the considered model, like shown in Fig. 3. The architecture is called as feedforward Neuro-Fuzzy type Takagi-Sugeno multi-input multi output. It uses Gaussian membership function in the fuzzyfication phase.

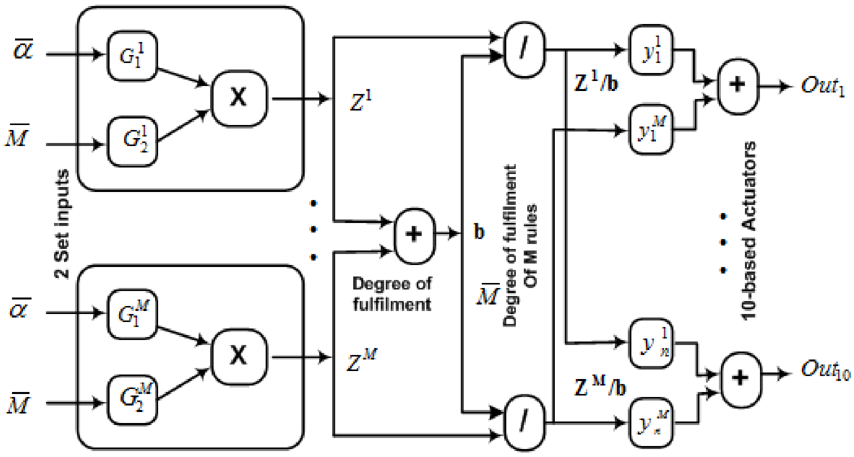


Fig. 3: Takagi-Sugeno-type MIMO (with input real and output BINARY) feedforward Neuro-Fuzzy network, no. of set input = 2, no. output = 10, No. membership function 17, training method: LMA

In particular, introducing the Gaussian membership functions to both NF methods G_j^n (j = 1, 2; n = *1*, …, *10*), as a fuzzyfication procedure for input pairs $\boldsymbol{X^D} = (\alpha^D, M^D)$, where α^D and M^D are the input set of the orientations (α_x, α_y, α_z) and the moments (M_x, M_y, M_z) of the moving platform with respect to the x-y-z Euler coordinates.

$$G_j^n(X_j) = \exp\left[-\left((X_j - c_j^n)/\sigma_j^n\right)^2\right] \tag{1}$$

with characteristic means c_j^n and variance σ_j^n together with the corresponding fuzzy rules R^n can be written as:

$$R^n : IF\ X_1 is\ G_1^n\ AND\ X_{21} is\ G_2^n$$
$$THEN\ y_i^n = w_{0i}^n + w_{1i}^n X_1 + w_{2i}^n X_2 \qquad (2)$$

with w_{0i}^n, w_{1i}^n and w_{2i}^n (for i = 1, …, *10*, and n = 1, …, *N*, *N* is the number of optimized rules for the model, here N = 17) being the Takagi-Sugeno weights (Takagi 1985), the common part of the considered Neuro-Fuzzy model calculates the continuous variables.

$$\bar{u}_i = \sum_{n=1}^{N} y_i^n \left[\prod_{j=1}^{2} G_j^n(X_j) \Big/ \sum_{n=1}^{N} \prod_{j=1}^{2} G_j^n(X_j) \right] \qquad (3)$$

From (3), the two different models, hereafter briefly referred to as NFTS and NFLUT, are derived by alternatively estimating the actuator activation states u_i through one of the following threshold operations:

$$u_i = round(\bar{u}_i) \text{ or } u_i = RLUT(\bar{u}_i) \qquad (4)$$

where *RLUT* indicates a properly Reduced Look-Up Table involving $\bar{u}_i$ as only input of the table. Additionally, the NFLUT requires the generation of the RLUT, which is here constructed by storing the most significant $\boldsymbol{u}$-$\boldsymbol{\bar{u}}$ correspondences that occurred during training with the known dataset $\mathfrak{R}$. Prior to their use, NFTS and NFLUT models require the tuning of the parameters c_j^n, σ_j^n, w_{0i}^n, w_{ji}^n (for j = 1, 2; i = 1,2; n = 1, …, *10*;). Here, the number of parameters for the considered MPRs is 564 parameters. The values of these parameters are found by an optimized learning procedure. The learning procedure employs 12% of the ***X*-*u*** correspondences known from $\mathfrak{R}$ for the 10-binary MPRs respectively, that generated from Eq. (5) below.

$$M(\alpha, u_i) = F \sum_{i=1}^{10} u_i \left[k \cdot [A(\alpha) - O] \times [A(\alpha) - B_i] / \| A(\alpha) - B_i \| \right] \qquad (5)$$

Moreover, the fuzzy logic system, once represented as the equivalent Multi-Input Multi-Output feed forward network, can generally be trained using any suitable training algorithm, such as standard Backpropagation Algorithm (BPA) that is generally used for training of the NN (Palit 2002). Because of its slow speed of convergence, BPA needs to be further improved. Alternatively, a second order training algorithm, such as the Levenberg-Marquardt Algorithm (LMA), can also be used. It is noted that LMA is actually a second order training algorithm that is based on the modification of Newton's method and uses Jacobian matrix in order to approximate the second-order partial derivatives (called as Hessian Matrix). In particular, the learning procedure in this paper is performed via the accelerated Levenberg-Marquardt Algorithm (LMA). Detailed application and equations are explained in (Palit 2005).

RESULT AND DISCUSSION

In order to find the best initial parameters, i.e.: c_j^n, σ_j^n, w_{0i}^n, w_{ji}^n - and to be updated in the training algorithm, we proposed randomized Hill Climbing (HC) procedure (Pasila 2013) in order to find the optimized number of rules *N* for each *10*-binary models. This procedure is a local search algorithm that tries to find the best local minimum from the large number of iteration procedures by permitting the best training parameters that minimize the error model (e_m) and neglecting the others. The optimized *N* membership function after two weeks searching time as the results of HC procedure are: *N*= 17 for the 10-binary MPRs. Regarding model performances for the ISA solution for 10-binary: NFTS shows $\mathbf{t_p}$ = 3.1e3s, $\mathbf{t_c}$=1.6e-3s, $\mathbf{e_g}$ = 5.15N, and **FGE** = 9.36%; while the NFLUT shows $\mathbf{t_p}$ = 6.3e3s, $\mathbf{t_c}$ = 3.1e-2s, and $\mathbf{e_g}$ = 2.64N; **FGE** = 4.47% ; where $\mathbf{t_p}$ is the time for preparing the model, including learning procedure (in second), $\mathbf{t_c}$ is the time for computing online (s), eg is a generalization error(in N), and **FGE** is full scale generalization error (in %). For the testing purposes, both methods use 1024 data testing. The comparison of testing performance between NFTS and NFLUT methods can be seen on Fig. 4.

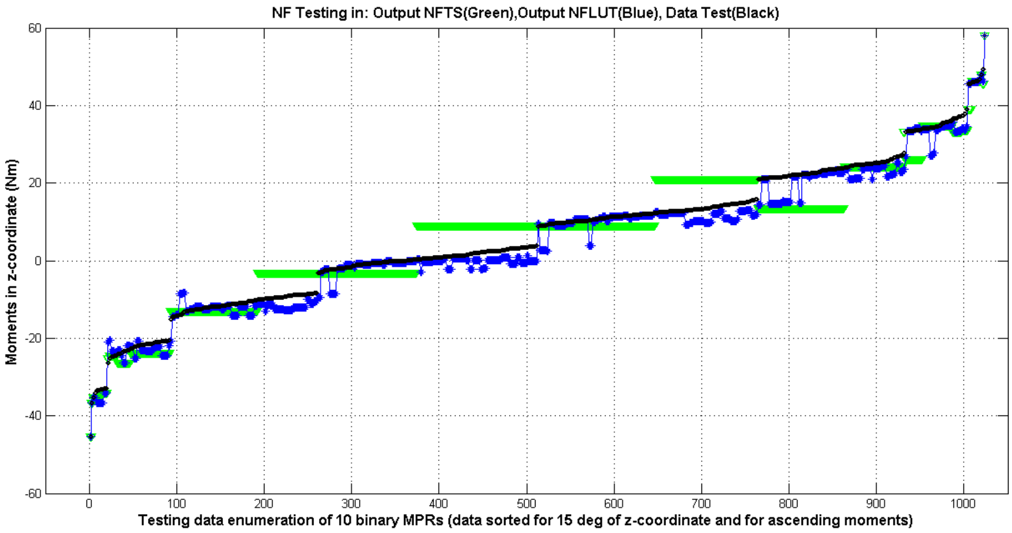


Fig. 4: Testing performance of 10-binary MPRs with NFTS and NFLUT methods

Conclusion:

As conclusion, this paper presented: 1) a 6-DOF massively parallel robots (MPRs) with 10 binary state force actuators; 2) Two models of Neuro-Fuzzy method type Takagi-Sugeno with Levenberg Marquardt Algorithm for the solution of inverse static analysis of the considered MPRs. They are NFTS and NFLUT. Thanks to the partitioned and spatially distributed actuator architecture, the considered discrete manipulator features rather sufficient and accurate torque generation capabilities, compared to the standard manipulator array mechanism. The results show that NFLUT features better generalization ability compared to the NFTS but requires a rather bigger computational time during the on-line phase.

AKNOWLEDGEMENT

The author would like to thank Prof. Vincenzo Parenti Castelli, DIEM, University of Bologna for letting him the chance to conduct the initiate research during 2009-2012 and Petra Christian University for supporting him with the extended research fund under "Hibah Penelitian" with No. 12/Pen-LPPM/UKP/2012.

REFERENCES

Chirikjian, G.S., 1994. A binary paradigm for robotic manipulators. Proc. IEEE Int. Conf. On Robotics and Automation, San Diego, CA, pp: 3063-3069.

Chirikjian, G.S., 1995. Kinematic synthesis of mechanisms and robotic manipulators with binary actuators. ASME J. Mech. Design., 117: 573-580.

Chirikjian, G.S., 1997. Inverse kinematics of binary manipulators using a continuum model. J. Intel. Robotic Syst., 19: 5-22.

Jang, J.S.R., 1993. ANFIS: Adaptive network Based Fuzzy Inference System. IEEE Trans. On SMC., 23(3): 665-685

Palit, A.K., D. Popovic, 2002. Nonlinear combination of forecasts using ANN, FL and NF approaches. FUZZ-IEEE, 2: 566-571.

Palit, A.K., D. Popovic, 2005. Computational Intelligence in Time Series Forecasting, Theory and Engineering Applications. Springer.

Pasila, F., 2013. Inverse static analysis of massive parallel arrays of three-state actuators via artificial intelligence. PhD Dissertation, University of Bologna.

Suthakorn, J. and G.S. Chirikjian, 2001. A new inverse kinematics algorithm for binary manipulators with many actuators. Adv. Robot., 15(2): 225–244.

Uphoff, E. and G.S. Chirikjian, 1995. Efficient workspace generation for binary manipulators with many actuators. J. Robotic Syst., 12(6): 383-400.

Wang, J. and G.S. Chirikjian, 2004. Workspace Generation of Hyper-redundant Manipulators as a Diffusion process on SE(N). IEE Trans. On Robotics and Automation, 20(3): 399-408.

Yang, P.H., 2001b. Design and Control of Bundles of Binary Actuators for Manipulator Actuation. PhD Dissertation, The Ohio State University.

Yang, P.H., K.J. Waldron, 2001a. Massively Parallel Actuation. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Como, Italy.