International Journal of Hybrid Information Technology
Vol.8, No.1 (2015), pp.141-152
http://dx.doi.org/10.14257/ijhit.2015.8.1.12

Intelligent Time Variation Nonlinear Fuzzy-Flatness Technique

Atefeh Chahkoutahi, Mohammad Reza MoradiPour, Mohammad Gholami, Sirous Ashja and Mohammad Hossein Rahimi

Department of Electrical Engineering, Delvar Branch, Islamic Azad University, Delvar, Iran
atefehchahkootahi@gmail.com

Abstract

The focus of this paper is on the development and high precision robust control of an electro-mechanical robot manipulator that serves as a sensing and motion system for hybrid testing. The originality of the design is inspired from the Stewart Platform mechanism for a parallel axis configuration and a two-degree-of-freedom (2-DoF) moving platform. This design resulted in strongly non-linear and coupled dynamics as well as an inertial moving platform that attracted model-based control strategies. A novel non-linear control technique based on differential-geometric flatness was selected to meet the multiple simultaneous specification control of linearization, decoupling and asymptotic tracking. Pole placement was used to achieve a stable tracking, while the fuzzy-logic added intelligence to the control system through an automatic tuning of the pole placement coefficients. Simulation results demonstrated the validity of the fuzzy-flatness control with asymptotic and stable tracking at different frequency inputs. For the experimental implementation, the real-time constraint was further imposed and the actuators time-delay was compensated for using a forward prediction algorithm based on a fourth-order polynomial extrapolation. This compensation demonstrated a well synchronized control signal at different excitation conditions. Moreover, the non-linear flatness control was systematically assessed for the experimental validation and its implementation was made accessible for future validation and perspectives. This current research has contributed to the rapprochement of three important autonomous domains, namely: Parallel Manipulators, Hybrid Testing and Automatic Control. In addition, it has inspired many research perspectives for robust non-linear control and multi-frequency substructuring.

Keyword: *robust control, hybrid testing, fuzzy flatness, pole placement, nonlinear flatness control, parallel manipulator*

1. Introduction

Over the last two decades, there has been a rapid growth in research efforts aimed at the development of hybrid testing techniques for complex engineering structures. Amongst these techniques, Real-Time Dynamic Sub structuring (RTDS) is considered to be the state of the art hybrid testing technique, which has resulted from research advances in numerical modeling, experimental testing and real-time control [1]. RTDS has allowed for the accurate experimental validation and dynamic assessment of several applications across numerous fields of mechanical [2–4] and civil engineering [5–7]. This current research aims to introduce this technique to aerospace structures with the main motivations being the need to understand the aero elastic response of novel wingtip devices resulting from the coupled structural and aerodynamic forcing [8]. The application of this research is therefore multidisciplinary in essence and ranges from aerodynamics and structures, to their associated

ISSN: 1738-9968 IJHIT

numerical modeling techniques as well as robust control strategies for RTDS [9]. However, the focus of this thesis will be on developing advanced robust and non-linear control techniques and should be considered the first step in the establishment of a RTDS proof-of-concept implementation for aerospace applications.

With the continuous increase of jet fuel price and new aviation restrictions towards reducing emissions, new technologies have been emerging to enhance aircraft performances by reducing weight, fuel consumption and emission [10-11]. A good portion of fuel consumption reduction could be achieved by introducing novel wing-tip devices such as winglets, to mitigate the induced drag, or by introducing morphing wing skin, to reduce the form drag by improving flow laminarity [12-13].

Designed as small vertical airfoils at the wing-tip, winglets reduce the induced aerodynamic drag associated with the wing-tip vortices that develop as the airplane moves through the air. According to Aviation Partners, winglets are flying on over 5,000 aircraft, saving approximately 3 billion gallons of fuel, which translates into cutting CO2 emissions by more than 32.2 million tons world-wide [10]. This is in fact a true technological achievement that should be explored further, especially with the advances in green aviation technologies. As Aviation Partners claims: *a greener world is on the wing*, which implies that novel wing designs and especially wing-tip devices should be investigated to further improve the aerodynamic efficiency and hence contribute to a greener planet [10-11]. Within this context, an extensive cooperative research has been conducted over the last decade between aerodynamicists and aircraft design engineers to investigate different novel wing-tip devices in order to increase the aerodynamic efficiency and contribute positively to greener aviation programs. For instance, novel wing-tip concepts being investigated at the Office National d'´Etudes et de Recherches A´erospatiales (ONERA).

This paper is organized as follows:

- In Section 2, main subject of real time dynamic subtracting is presented.
- Detail of parallel robot manipulator is presented in Section 3.
- In Section 4, control methodology is presented.
- Detatil of fuzzy-flatness control is presented in Section 5 and finally in Section 6, the conclusion is presented.

2. Real Time Dynamic Substracting

RTDS consists of creating a hybrid testing environment where only the critical part of the system (usually non-linear and difficult to model) is tested experimentally while the rest of the system is modeled numerically [6]. RTDS is often referred to by various terms including: experimental dynamic sub structuring, mechanical hardware-in-theloop testing, model-in-the-loop testing, real-time hardware-in-the-loop testing, hybrid dynamic testing, and assorted permutations of these terms. The structure of interest is split into two substructures and the interface between them is provided by a set of transfer systems as illustrated in Figure 1. These actuators, which are typically electric or hydraulic actuators [4], apply displacements to the nonlinear component via a control system which is designed so that the transfer system follows the appropriate output from the numerical model. Simultaneously, the force required to impose these displacements is measured and fed back into the numerical model to give a two-way coupling.

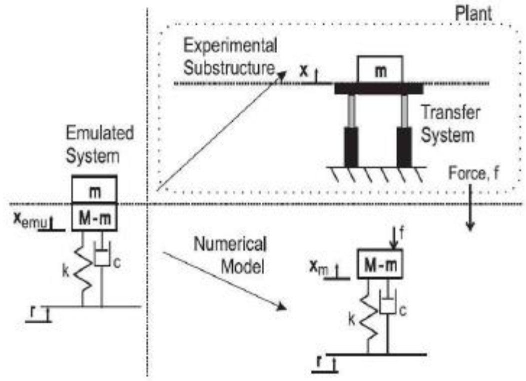

Figure 1. Schematic of Sub Structuring

The main objective of the substructuring test, is to emulate the dynamic behaviour of the original structure while the tested sub-structure(s) are not isolated from the rest of the system during the test [1]. Three main criteria are crucial to achieving this objective. Firstly, the loading equipment (or the transfer system) must be capable of imposing large loads and accurate displacements on the laboratory specimen, and the behavior of this loading system must be consistent and predictable over a wide range of frequencies [4]. Secondly, the computational solver used within the numerical model needs to be stable, accurate and computationally efficient. In certain applications, it must also be able to deal with non-linear multi-degree of freedom systems. Thirdly, the interaction between the two substructures must be reliably emulated by a set of communication devices (actuators and sensors), and the reciprocal boundary conditions must be imposed on the interface of each substructure. This implies that a quasi-instantaneous measurement of physical forces and displacements must be conducted whilst the two substructures have to be analyzed simultaneously and in real-time.

Figure 2 illustrates how the RTDS control problem is fundamentally different from a standard control one.

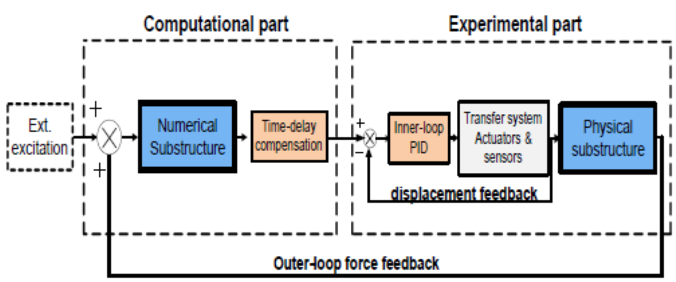

Figure 2. The Working Principle of RTDS Control in Block Diagram

3. Parallel Robot Manipulator

A manipulator is a robotic device consisting of a set of rigid links connected together by a set of joints. One fixed link is called the base, while another link whose motion is prescribed and used to interact with the environment is called the end-effector. Before the introduction of parallel and serial manipulators, some definitions are needed:

• Degree of Freedom (DoF): in mechanics, the DoF of a mechanical system is the number of independent parameters that define its configuration.
• Link: is the individual body which makes up a mechanism, and could be either static or dynamic (actuator).
• Joint: corresponds to the connection between two links providing the required physical constraints on the relative motion between these two links.
• Kinematic Chain: is the assembly of links that are connected by joints, and could be classified as a closed or an open kinematic chain depending on the links connection arrangement.
• End-effector: is the end of the manipulator which draws the motion workspace.

Robotic manipulators are highly non-linear and dynamically coupled multi-axis electromechanical systems. If the manipulator's links and joints are serially connected from the base to the end-effector, it is called a serial or open-chain manipulator as shown in Figure 3.

Figure 3. Examples of Serial Manipulators with Six DoF in a Kinematic Chain

However, if the end-effector is connected to the base by at least two kinematic chains, then it is called a parallel or closed-chain manipulator as shown in Figure 4. For instance, examples of cooperative closed-chain manipulators handling a common object are discussed in, where two planar rigid manipulators connected to three links through revolute joints, are cooperating in motion to handle a flexible object.

Figure 4. Typical Parallel Manipulators

Joints are usually actuated by motors (although passive joints can also be used), so that a whole manipulator can be controlled to perform a certain number of tasks. Parallel manipulators have received much attention from many researchers whose work covered diverse areas of kinematics, singularity analysis, calibration as well as dynamics and control.

Due to different design aspects of these concepts, specific advantages and aspects of performance can be expected. Table 1 summarizes the main features.

Table 1. Comparative Evaluation of Parallel and Serial Manipulators

Features	Serial Manipulator	Parallel Manipulator
Workspace	Large	Small
Stiffness	Low	High
Singularity Problems	Some	Abundant
Payload	Low	High
Inertial	Large	Small
Structure	Simple	Complex
Speed	Lower	Higher
Acceleration	Lower	Higher
Forward Kinematics	Easy	Difficult
Inverse Kinematics	Difficult	Usually Easy
Dynamics	Complicated	Very Complicated
Control	Simpler	Complicated
Design Complexity	Low	High
Cost	Lower	Higher

The dynamic formulation of robotic manipulator is:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau} \tag{1}$$

Where τ is actuation torque, M (q) is a symmetric and positive define inertia matrix, $N(q, \dot{q})$ is the vector of nonlinearity term. The nonlinear term of dynamic formulation is as follows [1-4]:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{B}(\boldsymbol{q})[\dot{\boldsymbol{q}}\,\dot{\boldsymbol{q}}] + \boldsymbol{C}(\boldsymbol{q})[\dot{\boldsymbol{q}}]^{2} + \boldsymbol{G}(\boldsymbol{q}) \tag{2}$$

Where B(q) is the matrix of coriolios torques, C(q) is the matrix of centrifugal torques, and G(q) is the vector of gravity force.

4. Control Strategies

Since the emergence of the first models of robot manipulators, the control problem of such systems has drawn the attention of numerous researchers. The reasons behind this are the countless challenges imposed by these systems, for instance:

- The highly non-linear dynamics of the manipulator and the actuators, which include inertia, friction, backlash, and in some cases mechanical flexibility.
- Cross-coupling between neighboring inputs and outputs of the system makes the control problem more challenging for decoupling.
- Time-varying system dynamic parameters due to changes in specimen mass (payload), testing configuration, speed of motion as well as component wear.

The following sections will give a brief review on different control techniques used to control robot manipulators. These techniques can be classified into model-based control (known as dynamic control strategies) and performance-based control (referred to as model-free or kinematic control strategies).

The model-free control strategy, also known as the kinematic control strategy, is based on the assumption that the joints of the manipulators are all independent and the system can be decoupled into a group of single-axis control systems. Therefore, the kinematic control method always results in a group of individual controllers, each for an active joint of the manipulator. Proportional-Integral-Derivative (PID) control, adaptive control, sliding mode control, fuzzy logic control, for tracking objectives for robust control using neural network control and hybrid neuro-fuzzy all belong to kinematic controllers. With the independent joint assumption, no a priori knowledge of parallel manipulator dynamics is needed in the kinematic controller design, so the complex computation of its dynamics can be avoided and the controller design can be greatly simplified. This is suitable for real-time control applications when powerful processors that can execute complex algorithms rapidly are not accessible. However, since joints coupling is neglected, control performance degrades as operating speed increases and a manipulator controlled in this way is only appropriate for relatively slow motion. The fast motion requirement results in even higher dynamic coupling between the various robot joints, which cannot be compensated for by a standard robot controller such as PID, and hence model-based control becomes the alternative.

Parallel manipulators exhibit inherently non-linear coupled dynamics and model-based control techniques were proven to significantly improve their performance. The literature on model-based control for robot manipulators is vast; an overview covering many different approaches is given in. For instance, based on approximated linear dynamic models, dynamic controllers were proposed for parallel manipulators with improved tracking performance. Although this type of controller is easy to implement by adopting the approximated linear model of parallel manipulators, the effect of the controller is limited to a small region of the configuration space for the inaccurate compensation of the non-linear dynamics. These works have placed solid foundation for further research on non-linear dynamic control strategies. For instance, the computed torque control (CTC), also known as feedback linearization, belongs to model-based techniques which use a model of the manipulator dynamics to estimate the actuator forces that will result in the desired trajectory. The basic concept of CTC is to line arise a non-linear system, and then to apply linear control theory. Since this type of controller takes into account the non-linear and coupled nature of the manipulator, the potential performance of this type of controllers should be improved compared to model-free control techniques. However, CTC method for robotic manipulators suffers from two difficulties. First, it requires exact knowledge of the manipulator dynamics, which is almost impossible in practical situations. Second, due to the fact that the CTC formulation is dependent on its dynamic model, it is not robust to structured and unstructured uncertainties, which may result in performance deterioration [11]. Therefore, the designer has to resort to an additional robust technique, which is usually model-free in order to achieve the required high performance and robustness. Hence, the two kinds of control strategies (model-based and model-free) are not absolutely opposite and are often combined to fulfil the global control task. For instance, the fusion of adaptive and computed torque control or adaptive and robust control where the robust technique is dependent on the dynamic model of the plant. The formulation of control is:

$$\boldsymbol{e(t) = \theta_a(t) - \theta_d(t)} \quad \textbf{(3)}$$

$$U_{PID} = K_{p_a} e + K_{V_a} \dot{e} + K_I \sum e \qquad (4)$$

5. Fuzzy Logic Control

Fuzzy-logic aims to provide an approximate but effective means of describing the behavior of systems that are not easy to describe precisely, and which are complex or ill-defined [8]. It is based on the assumption that, in contrast to Boolean logic, a statement can be partially true (or false) [9]. For example, the expression where the fuzzy value (near) applied to the fuzzy variable (distance), in addition to being imprecise, is subject to interpretation. The essence of fuzzy control is to build a model of human expert who is capable of controlling the plant without thinking in terms of its mathematical model. As opposed to conventional control approaches where the focus is on constructing a controller described by differential equations, in fuzzy control the focus is on gaining an intuitive understanding (heuristic data) of how to best control the process [11], and then load this data into the control system. The central idea of *fuzzy sets* is that elements can have partial membership in a given set. In contrast to a classical set, a fuzzy set, as the name implies, is a set without a crisp boundary [12]. In this respect, fuzzy sets are functions that map a value to a number between zero and one, indicating its actual degree of membership. A degree of zero means that the value is not in the set, and a degree of one means that the value is completely representative of the set [10]. They were introduced by Prof. Lotfi A. Zadeh in 1965 as an extension of the classical notion of ”set” and as a mathematical way to represent and deal with vagueness in everyday life. The fuzziness does not come from the randomness of the constituent members of the set, but from the uncertainties and imprecise nature of abstract thoughts and concepts. A Fuzzy-Logic System (FLS) is a non-linear mapping from the input to the output space, where the input is first fuzzified. The fuzzy sets computed by the fuzzy inference as the output of each rule are then composed and defuzzified. Fuzzification helps in evaluating the rules, but the final output of a fuzzy system has to always be a crisp number. (*i.e.* conversion from a fuzzy set to a crisp number). Fuzzy membership functions on the other hand, are defined in terms of numerical values of an underlying crisp attribute. For example: *Short*, *Medium* and *Tall* in terms of the fuzzy variable: *height*. In other words, determining how much each discrete input value belongs to each input fuzzy set using the corresponding membership function. Fuzzification is the process of translating crisp input values into fuzzy linguistic values (fuzzy sets) through the use of membership functions. Generally, fuzzy membership functions are defined in terms of numerical values of an underlying crisp attribute such as short, medium and tall in terms of the fuzzy variable ‘Height’. They are subsequently processed by the inference engine that retrieves knowledge in the form of fuzzy rules contained in the knowledge-base. Fuzzy knowledge-bases is implemented as a set of IF-THEN rules as follows:

• IF (condition1 AND/OR condition2) THEN (consequence).

In fuzzy logic terminology, the statement following the IF is known as the *premise*, *antecedent*, or *condition*. The corresponding statement following THEN is known as the *conclusion* or *consequent*, and the actual calculation of the consequent using the premises calculated from the fuzzified inputs is reserved for the inference engine. In designing fuzzy systems, one should decide whether the number of rules is sufficient and if there are specific interactions between the rules. These problems were discussed in detail in the works. The inference engine is the heart of a FLC and acts as the bridge between the fuzzification and defuzzification stages. It aims at translating the designers desired control rules from a linguistic representation to a numeric computation, and can be divided into three elements:

aggregation, composition, and accumulation. There are several types of FIS, which may be limited to two FIS, the most currently used, those of Takagi-Sugeno or Mamdani type. The performance of any fuzzy logic controller (FLC) is greatly dependent on its inference rules and can be drastically affected by the choice of membership functions. Thus, methods for tuning the fuzzy logic controllers are needed. Some applications considered neural networks and genetic algorithms to solve the problem of tuning a fuzzy-logic controller. In general, the system to be controlled using a FLC requires a crisp or discrete input, rather than a membership function that is produced by the inference engine. Defuzzification is the process of converting the fuzzy output set resulting from the inference process into a discrete number suitable for input to the plant. There are many different methods of defuzzification described in the literature, with varying levels of complexity. Two fundamental methods are known as the Mean of Maxima (MoM) method and the Mamdani's Center of Gravity (CoG) method.

The discipline of intelligent control techniques is one of the most active and fruitful areas for research on advanced control systems and a number of successful applications have been reported among various fields. Fuzzy control for instance is gaining widespread acceptance in a large variety of fields, from engineering to commercial and from forecasting to artificial intelligence. The reason behind such an increasing interest resides in its flexibility and good performance in many applications where other methods either tend to fail, or become cumbersome. The foundations of Fuzzy-Logic were established by Lotfi A. Zadeh in his seminal paper on fuzzy sets, while its application in control system was pioneered by

Mamdani. This latter has successfully carried out a pilot study on a model steam engine using fuzzy systems demonstrating that they may profitably and easily be used by control engineers. Since then, substantial research into fuzzy control systems has been performed. The first practical application by Mamdani and Assilian in 1975 paved the way for fuzzy control and although this alternate paradigm of control came up against much criticism it managed to capture the interest of many researchers. As such fuzzy control is now recognized as a standard, established method for controlling nonlinear systems, especially those that are complex or ill-defined and whose mathematical model is unknown or time-varying and requires human experience. However, fuzzy-logic cannot handle all types of uncertainties and is usually combined with an additional robustifying approach such as ANNs, H-infinity ($H\infty$) and Sliding Mode Control (SMC), which is a variation of variable structure control (VSS).

In standard control problems, the aim of the control engineer is distinguished from the start by one of the main control specifications known as; stabilization, tracking, disturbance rejection, or various combinations of the three. For the current research context, there are far more specifications than the standard three mentioned above, and they all have to be met simultaneously to satisfy the RTDS control requirements. The current research work will go beyond the time-delay compensation and will establish a proof of concept implementation of non-linear model-based control for hybrid testing. We will focus on controlling the bare rig in real-time without considering the sub-structuring force feedback. This shall consist of cancelling the coupled non-linear dynamics of the inertial moving platform as well as compensating for the actuators time-delay. This control objective is complex, consisting of multiple control specifications that have to be met simultaneously. To the author's knowledge, this is the first research contribution to hybrid testing, where all the control specifications have to be considered simultaneously resulting in a multiple simultaneous specification control (MSSC).

Based on fuzzy logic methodology

$$\boldsymbol{f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^{T}\, \zeta(x)} \tag{5}$$

where $\boldsymbol{\theta^T}$ is adjustable parameter and $\boldsymbol{\zeta(x)}$ is defined by

$$\boldsymbol{\zeta(x) = \frac{\sum_i \mu(x_i) x_i}{\sum_i \mu(x_i)}} \tag{6}$$

Where $\boldsymbol{\mu(x_i)}$ is membership function. $\boldsymbol{\tau_{fuzzy}}$ is defined as follows;

$$\boldsymbol{\tau_{fuzzy} = \sum_{l=1}^{M} \theta^{T}\, \zeta(x)} \tag{7}$$

The goal of the fuzzy controller can be summarized as to "determine the joint parameters of the robot that satisfy a given position of the end effector to be reached". The algorithm seeks to modify the joint angle values based on the target position considering the current state of the system. Instead of using absolute values for the position, the algorithm only considers the required displacement: how far does the end effector have to move to reach the position. The state of the robot is also characterized by the rate of movement of the end effector for a unit change in joint angle, which is called the end effector velocity, or the partial derivatives of the forward kinematics equations in velocity. Hence, instead of solving the inverse kinematics, this approach only needs the forward kinematics representation and its first derivative, which is straightforward to compute. The angle change required on a joint is a function of the end effector displacement needed and the rate of change of the joint (in distance per unit angle). Table 2 is used to show a rule base or fuzzy associative memory (FAM). In this table, input 1 represents the position displacement of the end effector that needs to be obtained while input 2 represents the velocity of the end effector with respect to the current joint or the end effector displacement for a positive unit change in joint angle. Both of these values are in their fuzzy state and are classified into 7 different values: negative large (NL), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM) and positive large (PL). The output follows the same logic: by matching the two inputs, one of 7 different fuzzy values is found.

Table 2. Fuzzy Association Bank

	Input 1=effector displacement required							
Input 2=Velocity of effector for positive joint change		**NL**	**NM**	**NS**	**Z**	**PS**	**PM**	**PL**
	NL	PL	PM	PS	Z	NS	NM	NL
	NM	PL	PM	PS	Z	NS	NM	NL
	NS	PL	PM	PS	Z	NS	NM	NL
	Z	Z	Z	Z	Z	Z	Z	Z
	PS	NL	NM	NS	Z	PS	PM	PL
	PM	NL	NM	NS	Z	PS	PM	PL
	PL	NL	NM	NS	Z	PS	PM	PL

The change in joint angle will be large if the displacement required is large. The same deductions can be made for small displacements. End effector velocity, however, only affects the sign of the angle change. For example, if the end effector velocity is negative for a positive change in joint angle (input 2), then the corresponding angle adjustment must be negative (output) to reach a positive position displacement (input 1). The process is applied successively for every joint. If after the first iteration the target position has not been reached, the entire procedure is repeated with the updated robot configuration until the target position is reached. Figure 5 illustrates the steps used.

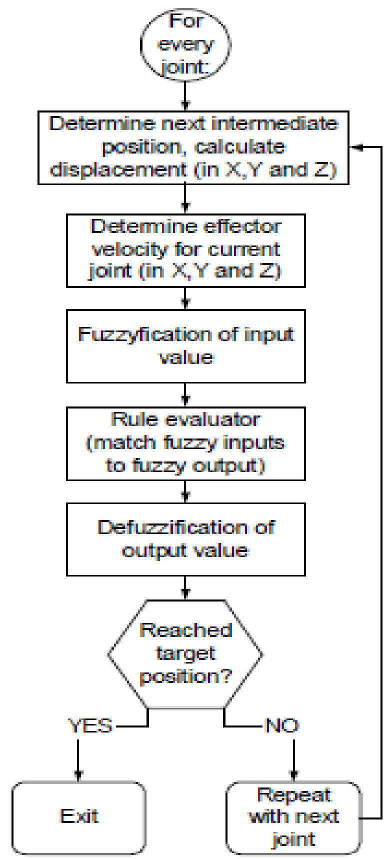

Figure 5. Fuzzy Block Diagram (without Collision Detection)

As the first input represents the displacement required for the end effector; it is encoded as *dx* and *dy* in Cartesian coordinates. In 3D workspace, a third displacement along the Z axis is added and computed similarly. For the second input, determining the end effector velocity is usually a simple process as the forward kinematics is a combination of sines and cosines of angles. The flatness implementation in simulation provided a perfect tracking for the reference trajectories with a negligible tracking error in both θ and x as illustrated in Figure 6 and Figure 7.

The static mass effect is demonstrated by the oscillatory motion around the datum line of • ± 50 Newton. The transient response is characterized by the initial response up to 1, 2 second. It is dominated by the actuators impulse to move the platform dynamics. Adjusting the constant input and output gains of fuzzy controller makes the tracking error of the manipulator system uniformly ultimately bounded. The control input to the actuators to achieve this tracking is illustrated in Figure 6.

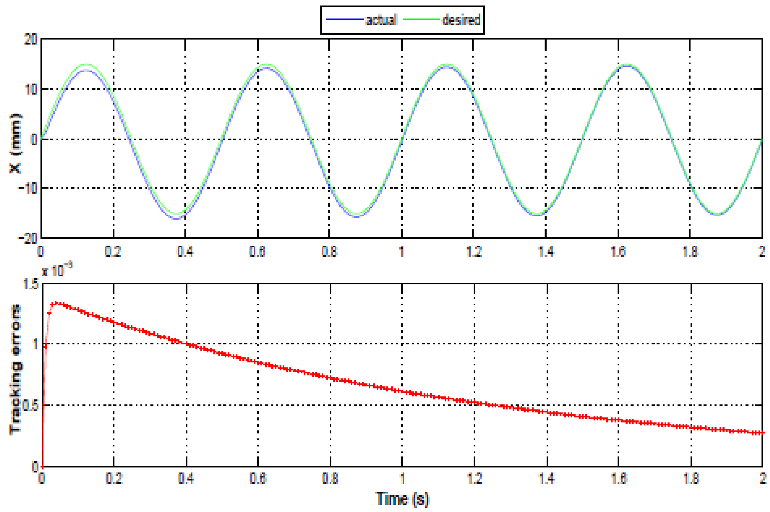

Figure 6. Fuzzy-flatness Tracking Control for Position

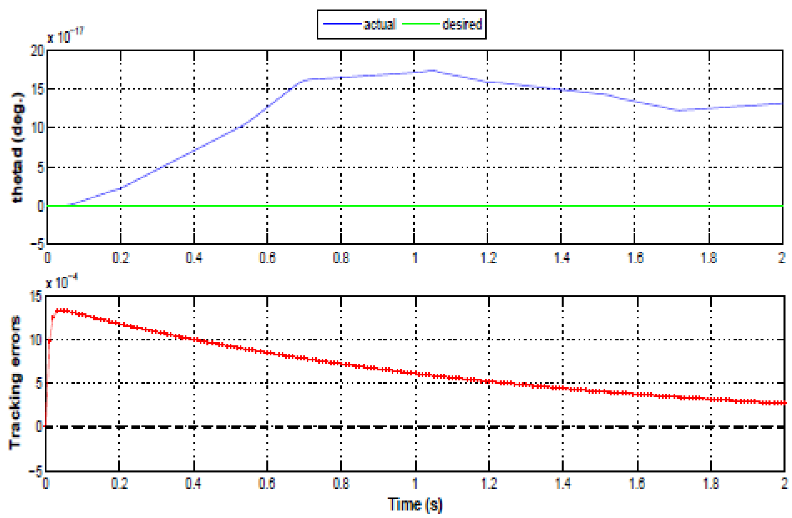

Figure 7. Fuzzy-flatness Tracking Control for the Decoupled Pitch Mode

6. Conclusion

The dynamic equations for the 2-DoF moving platform were derived from first principles using the Newton-Euler approach, resulting in a multi-variable non-linear dynamic model. The control objective was discussed within the framework of RTDS and the methodology with which it is operated was illustrated through a multiple simultaneous specification control (MSSC). A review of control strategies for robot manipulators is presented with examples from the literature to justify the selection of a suitable control strategy that could meet the MSSC objective. Differential-Geometric Flatness (DGF) was chosen to be the most suitable control strategy to provide a convenient framework for both trajectory planning and asymptotic tracking, meeting a number of performance specifications such as dynamic linearization and decoupling.

References

[1] T. R. Kurfess, "Robotics and automation handbook: CRC", (**2005**).
[2] J. J. E. Slotine and W. Li, "Applied nonlinear control", Prentice hall Englewood Cliffs, NJ, vol. 461, (**1991**).
[3] B. Siciliano and O. Khatib, "Springer handbook of robotics: Springer-Verlag New York Inc., (**2008**).
[4] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator", Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 27, no. 2, (**2002**), pp. 332-343.
[5] F. Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari and M. Mirazaei, "PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses", International Journal of Robotics and Automation, vol. 3, no. 3, (**2012**), pp. 106-150.
[6] F. Piltan, M. H. Yarmahmoudi, M. Shamsodini, E. Mazlomian and A. Hosainpour, "PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses", International Journal of Robotics and Automation, vol. 3, no. 3, (**2012**), pp. 167-191.
[7] M. W. Spong and M. Vidyasagar, "Robot dynamics and control: Wiley-India", (**2009**).
[8] A. Vivas and V. Mosquera, "Predictive functional control of a PUMA robot," Conference Proceedings, (**2005**).
[9] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," IEEE conference proceeding, (**2008**), pp. 212-217.
[10] N. G. Mozafari, F. Piltan, M. Shamsodini, A. Yazdanpanah and A. Roshanzamir, "On Line Tuning Premise and Consequence FIS Based on Lyaponuv Theory with Application to Continuum Robot", IJISA, vol. 6, no. 3, (**2014**), pp. 96-110. DOI: 10.5815/ijisa.2014.03.10.
[11] M. Bazregar, F. Piltan, A. R. Nabaee and M. M. Ebrahimi, "Design Modified Fuzzy PD Gravity Controller with Application to Continuum Robot", IJITCS, vol. 6, no. 3, (**2014**), pp. 82-94. DOI: 10.5815/ijitcs.2014.03.10
[12] A. Khajeh, F. Piltan, M. R. Rashidian, A. Salehi and E. Pouladi, "Design New Intelligent PID like Fuzzy Backstepping Controller", IJMECS, vol. 6, no. 2, (**2014**), pp. 15-26. DOI: 10.5815/ijmecs.2014.02.03
[13] A. Salehi, F. Piltan, M. Mirshekaran, M. Kazeminasab and Z. Esmaeili, "Comparative Study between Two Important Nonlinear Methodologies for Continuum Robot Manipulator Control", IJITCS, vol. 6, no. 4, (**2014**), pp. 66-80. DOI: 10.5815/ijitcs.2014.04.08