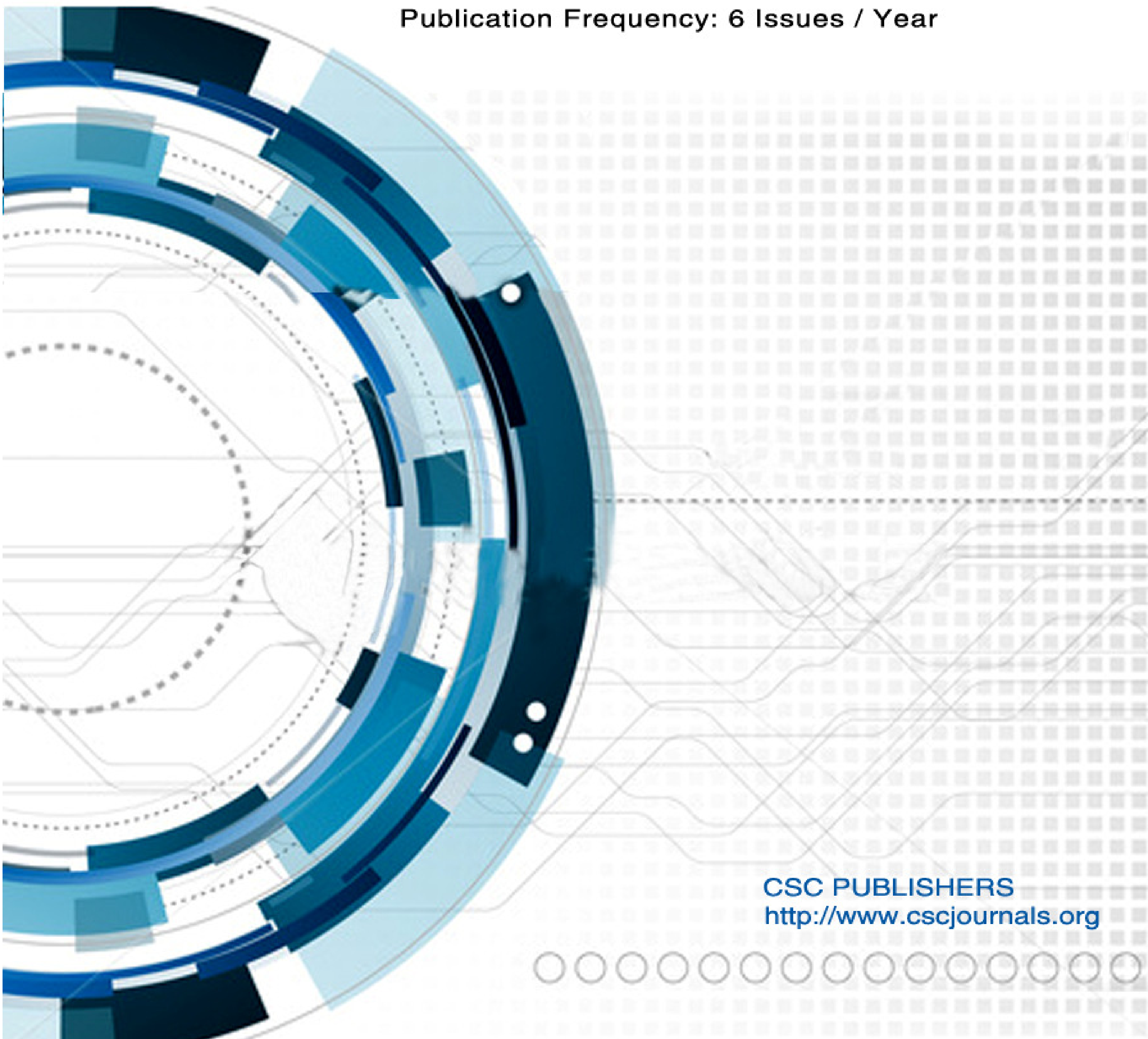
Editor-in-Chief
Dr. Kouroush Jenab

INTERNATIONAL JOURNAL OF ENGINEERING (IJE)

ISSN : 1985-2312

Volume 6 • Issue 2 • April 2012

Publication Frequency: 6 Issues / Year



CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ENGINEERING (IJE)

VOLUME 6, ISSUE 2, 2012

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-2312

International Journal of Engineering is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJE Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ENGINEERING (IJE)

Book: Volume 6, Issue 2, April 2012

Publishing Date: 16-04-2012

ISSN (Online): 1985-2312

IJE Journal is a part of CSC Publishers

http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2012

EDITORIAL PREFACE

This is the second issue of volume six of International Journal of Engineering (IJE). The Journal is published bi-monthly, with papers being peer reviewed to high international standards. The International Journal of Engineering is not limited to a specific aspect of engineering but it is devoted to the publication of high quality papers on all division of engineering in general. IJE intends to disseminate knowledge in the various disciplines of the engineering field from theoretical, practical and analytical research to physical implications and theoretical or quantitative discussion intended for academic and industrial progress. In order to position IJE as one of the good journal on engineering sciences, a group of highly valuable scholars are serving on the editorial board. The International Editorial Board ensures that significant developments in engineering from around the world are reflected in the Journal. Some important topics covers by journal are nuclear engineering, mechanical engineering, computer engineering, electrical engineering, civil & structural engineering etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with volume 6, 2012, IJE appears in more focused issues. Besides normal publications, IJE intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

The coverage of the journal includes all new theoretical and experimental findings in the fields of engineering which enhance the knowledge of scientist, industrials, researchers and all those persons who are coupled with engineering field. IJE objective is to publish articles that are not only technically proficient but also contains information and ideas of fresh interest for International readership. IJE aims to handle submissions courteously and promptly. IJE objectives are to promote and extend the use of all methods in the principal disciplines of Engineering.

IJE editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJE. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJE provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Engineering (IJE)

TABLE OF CONTENTS

Volume 6, Issue 2, April 2012

Pages

70 - 85 Rethinking Embedded System Design
Subhendu Das

86 - 95 Performance Testing and Comparision of A Turbine Ventilator, A Vent Column, and Their Combination under Thermal Buoyancy and Wind Effects
Nguyen Q. Y, Nguyen T. Bay, Ha Phuong

96 - 117 Methodology of Mathematical error-Based Tuning Sliding Mode Controller
Farzin Piltan, Bamdad Boroomand, Arman Jahed, Hossein Rezaie

Rethinking Embedded System Design

Subhendu Das *subhendu11das@gmail.com*
CCSI
West Hills, 91307, USA

Abstract

Embedded engineering is designed using objects of nature and it also interacts with nature. Therefore it is forced to obey the laws of nature. Nature does not make any assumptions. But all our mathematical and scientific theories do. Therefore these theories cannot be valid for embedded engineering applications. In this paper we present four new laws of nature that all embedded systems follow. These laws are (1) Boundedness (2) Finite time (3) Simultaneity and (4) Complexity. During the last fifty years embedded analog and digital engineering have evolved and changed significantly. However our mathematical and scientific theories remained in the original state. We select several theories commonly used in embedded engineering and show that none of them satisfy these laws. As a result, when we implement these theories in our embedded software, we are forced to add so many patches and kludges to make the engineering work, that our systems become very unreliable.

Keywords: Software, Embedded, Mathematics, Science, Engineering, Kalman, Education.

1. INTRODUCTIONS

In this paper our focus is on the problem of embedded system software and its failure to provide reliability and stability [1]. The embedded software has two categories of problems. The first problem is related to its own software design, implementation, and test issues. The second problem is related to the algorithm and theories that the software uses in its design. In this paper we are concerned with this second problem.

We believe that if we understand the root cause of the problem then we will be able to automatically find the solution. Unlike internet or PC type software, embedded system is part of nature. Therefore it must follow the laws of nature. We discuss four natural laws that all embedded systems follow: (a) boundedness law of all physical variables (b) finite time law of all activities, (c) simultaneity law of natural phenomena, and (d) the complexity law of nature. If we examine any embedded system software and hardware, we will find that all four laws are implemented very carefully. On the other hand if we examine the mathematical and scientific theories with similar care we will find none of them obey the above four laws. These inconsistencies are the root cause of the software unreliability issues of the second problem mentioned above.

It is not that we did not know about these laws. They existed in nature for eternity. They slowly started emerging as our engineering became more and more complex. We started integrating many subsystems together; and we configured them to communicate with each other, to create more realistic systems. Eventually we implemented real time multitasking software, which highlighted our focus on finite time repeating concepts and the simultaneity law of nature. The technology advanced so rapidly and implemented things in such details, thus increasing the complexity, that other branches of mathematics and science did not get opportunities to realize it. As a result we see a crisis in embedded system software and in our theories.

It must be recognized that our engineering requirements are significantly more complex [2] than what they were during the time of Newton. As an example, we may say that the first missile must make a hole in the building and the second missile must go through that hole. We have used this kind of technology in our recent battle fields of Iraq during 2003. This kind of precise requirements cannot be implemented using the theories that still inherit the characteristics of the simpler days of Newton. The problems are in the details; math and science do not go at that level of details, only engineering does. Note that this is not about approximations; this is on gross violations of the laws of nature. Our embedded technology has significantly advanced during the last fifty years, but our math and science did not keep pace with it.

Using standard examples from mathematics and science we show that they cannot work in embedded engineering because they violate the new laws presented. We use (a) all three Newton's laws, (b) Laplace and Fourier transforms, and (c) Kalman filtering to present the concepts of this paper.

The objective in this paper is only to discuss the problems, and highlight their root causes. The problem itself is very complex and big. It should be realized that the details must be avoided to present the comprehensive nature of the problem in a paper of this size. In reality all solutions are embedded in the detailed understanding of the problems. It is possible to create new architectures, based on Kalman's philosophy of using only measurements, and thus help to produce reliable embedded software. We present, however, an alternative approach little bit, near the end. The paper may appear like a philosophical presentation to people in the areas of mathematics, science, analysis, and simulation etc., but for all experienced hands-on engineers it will be quite realistic, obvious, and normal.

2: DEFINITIONS

Before we talk about math, science, engineering, their theories, assumptions, and compare them with new laws mentioned; it may be necessary to define the terminologies. Nature has only two kinds of things; some objects (living and non-living) and some actions. Actions are like forces of nature and have some energy associated with them. In some sense actions are characteristics of objects also. For example light energy is a characteristic of sun; similarly wind force is a characteristic of earth.

Definition of Laws of Nature

The laws of nature are the universal characteristics of the objects of nature. They are physical. They exist independent of human experiences and assumptions.

Everything that we see around us is engineering. The cars, airplanes, roads, buildings are all products of engineering. A product is a physical hardware that we can touch. Our modern engineering products are very sophisticated and satisfy complex requirements.

Definition of Engineering

It is a process that is required to create an useful product.

Thus engineering is not the textbooks on engineering subjects, like mechanical, electrical, etc. All products use natural components, and therefore they also obey natural laws. Thus we can define science in the following way:

Definition of Science

It is a collection of manmade theories that tries to explain the laws of nature.

Consider an example to clarify the distinction between science and engineering. If we place a magnetic needle under a wire, and pass current through the wire, then the magnet will be deflected. We call this an engineering experiment. It is a product that we can see, touch, and learn about it; and it does something useful also. The process used to demonstrate this needle movement is engineering. The science part says that the magnet has a field called magnetic field, the electricity creates a field called electric field (or may be a magnetic field); these two fields interact and create a force that deflects the magnet. The mathematics is a symbolic language. Its main purpose is to justify the scientific theories.

Definition of Mathematics
It is a symbolic language, used to describe expressions of natural language.
The theory is always a set of conclusions or a set of rules. But it also says that these rules or results will hold only under certain assumptions. These assumptions are thus a part of the theory.

Definition of Theory
A Theory is (a) a collection of assumptions and (b) a collection of conclusions that only hold under the assumptions.

Example of a Theory
Newton's First law: (a) In the absence of any interaction with something else (b) An object at rest will remain at rest (c) An object in motion will continue in motion at constant velocity, that is, in constant speed in a straight line

The item (a) in the above law is the assumption. The items (b) and (c) are the conclusions. The last two items will be valid only when the first item (a) is valid. A theory has two parts, if any one of the two parts fails then that theory will be invalid and we will say that the theory does not work in engineering or simply does not work.

Definition of Invalidity
A Theory is invalid if (a) Its assumptions cannot be tested or implemented or (b) Its conclusions cannot be verified by any experiment

3: THE LAWS OF NATURE
In this section we present the following new laws of nature: (a) boundedness law (b) finite time law (c) simultaneity law and (d) the complexity law. We describe them in details in the subsections below.

3.1 Boundedness Law
Let x denote any engineering variable, like voltage, current, water pressure, etc. In engineering x always has a lowest and a highest possible value. Or in other words they cannot take any arbitrary value from minus infinity to plus infinity. We call this feature of a variable as the boundedness law of nature. We will also call it the nonlinearity or saturation law of engineering. Using mathematical notations we can express this law in the following way:

$$L \leq x \leq U \tag{1}$$

Here L can be positive, zero, or any negative number but cannot be minus infinity. Similarly U can be negative, zero, or any positive number, but cannot be positive infinity. However U must be greater than L.

Figure 1 describes graphically the logic to implement (1). The horizontal axis is the input axis for the variable x. Along the horizontal axis, x can take any value from minus infinity to plus infinity. We show the boundedness of x in the vertical or output axis of the graph. The graph shows when x is between a and b, whatever way x changes, similar changes happen in the output axis. However, when x goes beyond b you can see that on the output axis it stays fixed at U. That is the output is limited to U. The same is true in the lower direction of x, and is limited by L. The box in Figure 1 represents a non-linear system.

Wherever there is an engineering variable in a system, an embedded engineer adds this box at that location of the system, to protect the components there, otherwise the system will fail or burnout. The box is also known as limiter. This box is implemented in analog hardware circuits in the form automatic gain control. It is also implemented in both digital hardware and embedded microprocessor based software. All hands-on experienced engineers will automatically place these boxes in their design. This is a very common embedded engineering practice.

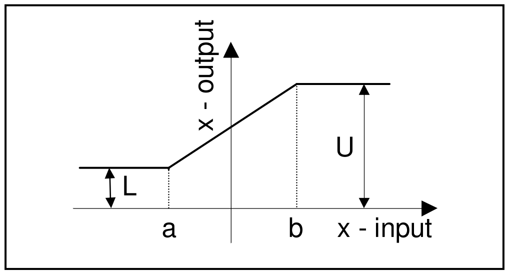


FIGURE 1: Saturation Non-Linearity

Since the box is a nonlinear box representing the natural law, and implemented by design, all engineering systems are nonlinear systems by nature. Or in other words there are no linear systems in engineering. Note that linear systems are not approximations to this kind of boundedness law. We explain later that local linearized solutions do not satisfy modern complex engineering requirements. They will violate the complexity law mentioned below.

The examples of such systems with boundedness law are abundant in engineering. All we have to look for is to see the schematic of any embedded hardware and the source code of real time embedded software or firmware. It is not that engineering has completely ignored it, theory has been developed, as an example see [3]. But it is still not practical for the complex problems of our time. However mathematics and science have largely ignored the boundedness law, as we show later with examples. Observe that this is not a philosophical issue, it is a real engineering problem, and creates conflict with existing theories that we use.

3.2 Finite Time Law

These days most of the complex engineering systems are controlled by one or more digital microprocessors and software. All activities that these systems perform are done in small interval of time duration, of the order of several micro or milliseconds. And such activities are repeated continuously [4].

Consider the example of a robotic arm, picking up an item from one place and dropping it in another place and repeating the process in, say, less than a second of time. Similarly a digital communication receiver system, like GPS, receives an electrical signal of microsecond duration, for example, representing the data, extracts the data from the signal, sends it to the output, and then goes back to repeat the process.

Our software runs under real time multitasking operating systems which are also nothing but finite state machines. A finite state machine is a collection of finite number of activities of finite durations, repeated asynchronously and/or synchronously based on the external as well as internal events. Every time a task returns, it finds a different environment. The previous tasks have operated on the system and created a new environment. Thus the same finite duration task or activity is always performed on different signal and under different environment.

Although our systems run continuously, like GPS transmitters and receivers, traffic light systems at street corners, but if you look at the internals you will always find that the building blocks are based on finite duration processes.

It is quite surprising that the embedded system evolved to perform things as continuous repetition of finite time activities. If we observe carefully we will see that the nature is also composed of finite time activities. Everything in nature has a birth process, maturity process, and death process. Each one of the processes is also a finite duration process. We also see that earth rotates over finite time around its axis and around the sun. Thus repeating finite time process is a law of nature. Yet, most of the mathematical theories that we commonly use, assume infinite time, as we show later with examples. We also show that if we replace the infinite time by finite number, however large, the theory completely fails. Thus again, these are not approximations of large numbers, these are collapse of theories, as explained later.

3.3 Simultaneity Law

A very important characteristic of nature is its simultaneity. Everything in nature occurs simultaneously and interactively. All humans are interacting constantly, simultaneously, and all over the world and for all time. So is true with all physical objects. We are never isolated. A company on precision weight measurement system [5] uses the moon's gravity effect, as it travels over earth, to precisely measure the weight of a mass on earth. Thus simultaneity is global and not just local. This company's products show how complex and sophisticated our modern requirements are. Before we even realize, everything in engineering will be simultaneously integrated together just like our natural world is. But our math and science are not yet ready for it. Most of the theories that we use now are more than hundred years old. Requirements, concepts, and philosophies of those simpler days are deeply embedded in those theories.

The real time operating system (RTOS) also implements this simultaneity law in engineering. It is a multitasking system that interacts with interrupts from external and internal sources. Basically RTOS is a finite state machine, running on finite time intervals, and is designed to simultaneously accept the changes in the environment. Many of our embedded systems interact with external computers via serial interfaces. In many cases these interfaces bring user commands also. These interfaces are constantly monitored by several tasks to reconfigure the system according to the changes in the environment. Thus simultaneity is built into all embedded systems.

Clearly, RTOS is beyond the scope of mathematics and science, but it is an integral part of modern engineering. Anytime a task switches from one to another, it finds the environment completely changed. When the task was in sleep mode, the simultaneity law worked and changed that environment. But our present theories rely on the continuity of states from one task to another, but that does not hold under RTOS and all other laws discussed here. We show later that most of our theories do not have means to accommodate all the laws mentioned,

3.4 Complexity Law

All natural systems are immensely complex and indescribable. To illustrate, consider the Grand Canyon. If we ask the best author of the world to describe the Grand Canyon in written language; you will find that the description will be of no match with your experience and feeling when you personally see the Grand Canyon. This written document is a model of the Grand Canyon. Thus nature is beyond description by our language and therefore cannot be modeled by a symbolic language like mathematics. The Grand Canyon is a static example of complexity. The dynamic complexity of nature is even more severe than Grand Canyon. Here is one more example to convince the readers about the complexity law. Watch the 3D simulation of the human brain in operation from the discovery channel [6] to comprehend the nature. This is only a simulation; the real thing is actually lot more complex.

Nature has evolved over billions of years. As a result everything is very complex in nature. From a very small thing like an atom to a very large system like a galaxy are all very complex. We should recognize that not only the components are complex; the laws that govern them are also equally complex. Our engineering uses these complex components from nature and implements these laws of nature to make products that are supposed to satisfy very complex requirements. Thus embedded engineering is as complex as nature.

Imagine what is inside a microprocessor. It has billions of electronic components inside it. The processor has hundreds or thousands of 32-bit registers, each bit must be carefully programmed to make it work according to desired performances and requirements. These registers exchange and process information with nature via analog to digital and digital to analog converters at nanoseconds and microseconds speed. The numbers inside the microprocessor also change continuously because of variations in nature, like changes in temperature, drift of component parameters etc. The nature looks completely different at that level of speed and 32-bit details. Our mathematics, which was developed hundreds of years back, cannot comprehend such complexities.

Even today many embedded software do not use floating point processors. The integer processors require scaling of variables. Scaling is a nonlinear process to keep variables within bounds, essentially implementing the boundedness law. This statics scaling process can never work, in real time and under the circumstances governed by the laws mentioned. Texas Instrument, which manufactured such a processor finally decided to make a floating point version to accommodate requirements of embedded engineering. Matlab simulation software created dynamic scaling every time it scaled a variable in its simulation to give correct results. This approach, although correct, is clearly not feasible in real time engineering.

All the previous three laws, boundedness, finite time, and simultaneity, are all working together in nature and therefore also in embedded engineering. This togetherness adds another dimension to the complexity of nature. We call this complex nature as the global space time (GST) environment and it is tightly integrated with all embedded systems. We should also point out that simulation environment cannot be created to test out such a complex and simultaneous environment for a real time embedded engineering system.

4. VIOLATING THE LAWS

In embedded engineering software, we use many mathematical and scientific algorithms. None of these theories obey the new laws mentioned in the previous section. In the following subsections we take many well known theories that are often used in embedded systems and show their incompatibility with these laws of nature. All these theories make simplifying assumptions, but nature does and cannot make any assumptions. Note that we are not trying to find faults in these theories; they work perfectly according to their definitions and assumptions. All we are saying is that they were invented hundreds of years back and not valid any more for our modern engineering. Engineering has advanced significantly during the last fifty years but our theories did not make similar progress.

When we implement these theories they fail to work. But the software and test engineers know how to fix them. They are forced to add many adhoc patches and kludges to make the engineering work. That is why we say math and science do not work, but engineering does. However, as a result of these patches the embedded system remains very unreliable and unpredictable.

The two worlds, designers and testers, are isolated and do not know the tools and languages of each other and cannot communicate. We see an interesting parallel highlighted [7] by the Nobel Laureate in economics Wassily Leontief – “How long will researchers working in adjoining fields abstain from expressing serious concern about the splendid isolation in which economics now finds itself?” System engineers do not seem to recognize the inadequacy of our mathematical and scientific theories, which are in complete isolation with embedded engineering.

4.1 Linear Theories

The above boundedness law shows that all engineering systems are nonlinear by design requirements. Therefore all mathematical and scientific theories that are based on the assumption of linearity are no good for any engineering system. As an example, linear control system theories, based on linear Laplace transform theory cannot work. Their applications to engineering problems are theoretically incorrect. Violating the boundedness law is not an approximation; it is a violation of fundamentals. It will lead to wrong results, instability, and system crashes. According to our definition of invalidity, Kalman filter, Laplace transform, and linear control theory cannot work in embedded systems because they violate the assumption of boundedness.

Most of the engineering systems have multiple modes of operations, like transient mode, steady state mode, low voltage mode, high voltage mode etc. In many cases, abnormal situations happen and cannot be avoided. This boundedness law protects the systems from failures. In almost all cases there is no theory that considers such assumptions and possibilities in their theoretical proof, mostly because they are all linear theories and therefore ignore this boundedness law.

Because of the boundedness law, in a typical engineering system there can be more than ten such nonlinear boxes. In many large systems there can be hundreds of such boxes. In most cases we have to show that the system is working properly at the boundary points of the nonlinearity. In many applications it will be a requirement to go to the boundary points L and U of Figure 1, and maintain the operating conditions at these limits. For example, the motor must run at the maximum speed, the voltage used must be of full value, the angular position must reach the limiting position etc. to achieve the desired performances.

It will be rarely required to operate the system in the linear region of the saturation nonlinearity. Moreover, when there are more than one or two or ten such nonlinear boxes in a system, it will be almost impossible to keep all variables simultaneously in the linear region of the saturating boxes, and still maintain the optimal performance. This will happen because the nonlinear equations that define the operation of the system, such as (3) below, will not necessarily create equilibrium positions in the linear region at all operating conditions.

Besides the boundedness law, almost all systems are also nonlinear. These systems are modeled using nonlinear equations. For example expression (2) is a linear model but (3) is not.

$$\frac{dx}{dt} = ax \tag{2}$$

$$\frac{dx}{dt} = bx + cx^2 \tag{3}$$

In an isolated environment, as discussed later, (3) can be linearized to (2) and linear theories can be used as approximations. However under boundedness law both (2) and (3) will fail to work. Moreover, there are no isolated systems or environments in nature or engineering.

4.2 Laplace and Fourier Transforms

Under the operational environment of the finite time law any scientific or mathematical theory that is based on infinite time assumption will be inappropriate for embedded engineering systems. According to our definition they will be invalid, because they violate the finite time assumptions. Consider for example the very well known Laplace transform theory defined by (4):

$$F(s) = \int_0^\infty e^{-st} f(t) dt \tag{4}$$

In the expression (4) the variable t is usually considered as time. It shows that the time must be valid from zero to infinity. Thus the function F(s) on the left hand side, called infinite Laplace transform (ILT), will be valid only for infinite time system. Thus a major theory, Laplace Transform, of mathematics is invalid for embedded engineering applications. It is also well known that (4) is based on the assumption of linearity. And therefore it will also not be applicable for any engineering systems under the boundedness law.

Let us consider an example with finite time T to bring out the fact [8] that the Laplace transform is based on infinite time assumptions, that is, it cannot be used for finite duration signals. The finite duration step function *f(t)* is defined by

$$f(t) = \begin{cases} 1 & 0 \le t \le T \\ 0 & otherwise \end{cases} \tag{5}$$

Using the definition (4) we get the expression for the finite Laplace transform (FLT):

$$\mathcal{L}_T(1) = \int_0^T e^{-st}.1.dt$$

$$= \frac{1}{s} - \frac{1}{s}e^{-sT} \quad (6)$$

$$= \frac{1 - e^{-sT}}{s} \quad (7)$$

We can see from (6) that the FLT has the standard ILT term $\frac{1}{s}$ and another expression involving e^{-sT}. The second term is zero only when T is infinity. Thus if we use only the first part of (6), the ILT part, then we will implicitly assume infinite time situation for our finite time problems and the Laplace model will not be correct for real engineering problems.

Observe from expression (7) that the FLT does not have any poles at the origin, but the ILT has. At s=0 the expression (7) takes 0/0 form. Thus by using the first part of (6) for finite time engineering problems we artificially inject poles in the models. The entire Laplace Transform theory must be revised and rewritten for applications in finite time engineering [9]. For embedded engineering applications ILT is not a correct tool and its use will make the software unreliable.

Similarly, data analysis that we have done using infinite time Fourier theory could be wrong also [4]. If you analyze this way then you will find that all mathematical theories will be invalid under all four laws. It has been shown that by switching from infinite time to finite time Nyquist Sampling theorem changes [10]. This new sampling theorem can change the design of embedded systems significantly, and may even give a new direction for embedded design.

4.3 Newton's First Law

The assumption behind the first law is – "In the absence of any interaction with something else". Is there a place on earth where there is no interaction with something else? The answer is no. Everything in nature is tied together by the simultaneity law. Embedded engineering is embracing that law more and more in its implementations. Therefore the first law will become invalid, by our definition, if we use in embedded engineering.

You cannot place a ball above the surface of earth and leave it alone there, because earth is interacting with the ball and it will make it fall. Thus in near earth all objects are interacting with earth's gravitational force. If you put the ball in deep space, then it will face the gravitational attractions of sun, earth, moon, and all other planets. Their resultant force will not be zero. Thus there is no place in the universe where there is no interaction with something else.

If you leave the ball in the deep space, it will immediately start moving in a curved path. The path will be curved because the objects in our universe are constantly moving. The total force on the ball will be changing all the time, both in magnitude and in direction; therefore it cannot go in a straight line. Thus we see that the conclusions of the law cannot be true. The conclusion is false because the assumption is wrong. Newton [1642-1727] discovered these laws almost three hundred years back. We should not expect that they will work now for our modern engineering.

The following statements can be found in the textbook [11, p. 8] about the Newton's first law: "We could hardly sustain that this principle [First law] is a strict experimental result. On the one hand it is not evident how to recognize whether a body is free of forces or not. Even if a unique body in the universe were thought, it is undoubted that its movement could not be rectilinear and uniform in every reference system".

Clearly, to make the law work in embedded engineering, we have to use kludges and patches in our engineering software, and the software will never be robust. Newton's first law violates the

Simultaneity law discussed before, which says everything is interconnected and working simultaneously at same time to make things happen. Newton's all three laws assume isolated environment, which is not feasible in modern embedded engineering. Thus all three Newton's laws are invalid for engineering.

4.4 Newton's Second Law

In terms of mathematical notations this simple law is expressed as:

$$f = m * a \tag{8}$$

Here f is the net force acting on an object of mass m. And the resultant acceleration of the object is a. Assuming that the mass is one, and replacing the acceleration by the second derivative of position x, we can rewrite (8) as in (9).

$$\frac{d^2x}{dt^2} = f \tag{9}$$

The first level of modifications that engineers have added to the right hand side (RHS) is the gravitational force g:

$$\frac{d^2x}{dt^2} = f + g(x) \tag{10}$$

In (10) we write g as a function of x, because gravity depends on the position. Note that the quantity x is a three dimensional vector in space, it has North, East, and Up (NEU) coordinates, near earth. The earth has been modeled as an ellipsoid, like the World Geodetic System 1984, (WGS84), and then extensive formulae has been developed by mathematicians, physicist, and engineers to define g(x) as a function of NEU coordinates. These expressions of g are quite complex and can be found in [12, Ch. 7].

Since the earth is rotating around its axis, there is always a force, called Coriolis force that acts on all objects near the earth space. This force has been shown to be dependent on the velocity of the object. Thus the expression (10) should be modified to (11) following [13, p. 76]. Where Ω is related to the earth's angular velocity, which is a constant. A derivation of the formula can be found at [14].

$$\frac{d^2x}{dt^2} = f - \Omega * \frac{dx}{dt} + g(x) \tag{11}$$

Thus we can see that Newton created a very simple equation (8). Such a simple equation violates the complexity law. Since Newton's original equation did not consider x and dx/dt in the RHS of expression (11) we can safely say that the Newton's formula cannot satisfy modern requirements. The second law makes the same assumption as that of the first law. It assumes an isolated environment [15, p. 114], which does not exist anywhere in the universe. It should be pointed out that even after significant modifications of (11) engineers were still not satisfied with the requirements. At present navigation engineers use the help of GPS (global positioning system) to augment inertial navigation system [16] based on (11). We must satisfy all four laws of embedded systems simultaneously with valid design concept without any flaws. At this time we do not have any theory to accomplish that.

It is not necessary to know the modified or better formula to express nature. For, there is no end of improvement in any formula. At the same time our engineering requirements are also becoming more and more sophisticated. We must know that nature is very complex because of GST, and such simple things, like (11), cannot represent nature. If we realize this concept to start with then we will automatically rethink engineering design.

Any assumptions in theory may cause many kinds of problem in embedded system. Most important concern is the stability or convergence of such algorithms. Most of such theories do not have any proof of convergence in the environment of RTOS and under the boundedness law. As a result the solution that we may get can be completely wrong. Note that the approximation (9) is valid for (11) only under isolated environment that neglects all the new laws presented. Expression (9) says that the object will travel in a straight line, whereas (11) says the object cannot fly in a straight line. Thus application of (9) may cause severe problems even in moderately complex systems.

Similarly observe that (6) has ILT part as its first term. In some sense it can be said that ILT is an approximation of the FLT. But we can see that this kind of approximation can lead to significant distortion of theory, concept, and philosophy. In the case of ILT this approximation moves us from analytic functions to a function of many poles.

4.5 Newton's Third Law

All Newton's laws make the assumption of isolated environment or absence of any interactions from other forces. In a sense they are valid only for point particles with nothing else in the neighborhood as pointed out in [17]. We should recognize that nature does not make any assumptions. Our real time embedded systems also cannot make any assumptions because they interact with nature and they have to work.

This third law can be found in [15, p.120] and has been explained in the following way. The forces always occur in pairs or that a single isolated force cannot exist. Any one of these two forces can be called the action force, and the other one then can be called the reaction force. The reaction force is equal in magnitude of the action force and of opposite in direction. Thus the sum of the two forces is always zero and can be written as in (12):

$$F_1 + F_2 = 0 \qquad (12)$$

Since we are not isolated, for every action there will always be more than one reaction $\{F_2, F_3, \ldots F_N\}$. However the summation of all reactions must still be equal to the original action that produced all the reactions. Therefore in real life we should have (13) instead:

▪

$$F_1 = -(F_2 + F_3 + \cdots + F_N) \qquad (13)$$

For example, if you throw a stone, the stone will react with the air particles and will generate a chain of reactions, as it moves in its path. Note that the stone also reacts with the gravitational force and changes its path accordingly. The gravitational force is always there and will react with everything and produce different reactions. The stone will eventually hit a place and create many reactions there. Practically, therefore we cannot have a single reaction. Thus action and reaction always occur in pair as in (12) is not realistic.

Design based on (13) or based on the acknowledgement of simultaneity will result in a new approach to embedded system design. At this time all our mathematical and scientific theories are based on isolated environment. Expression (13) shows that the summation of all forces is equal to zero. Thus the third law is essentially a law of conservation [18]. Thus all we need is to follow the law of conservation.

4.6 Kalman Philosophy

The following description of the evolution of Kalman Filter (KF) theory will demonstrate how our mathematical theories are inconsistent with nature and embedded systems. That is, this historical experience for over thirty years has shown that KF cannot work in embedded systems. Alternatively, this history in a sense validates the existence of the laws of nature mentioned in this paper and their use by embedded systems and also indicates the need for a different philosophy.

Many engineering systems can be described by their differential equation models, which can be generated using system theory [19]. These differential equations have several important components: (a) the equation itself, (b) the control variable, (c) the initial value constant and (d) the parameters. All of them must be known precisely for us to find the precise solution of the system. Kalman said that both control variable and the initial conditions are never known correctly, and he was right. Therefore he proposed a method, which is now known as KF. He proposed to take the measurements from the system and then use them to estimate the solution precisely. There are many books on the subject now, see for example [20].

Very soon engineers realized that the KF does not work. The assumptions behind this theory are not valid for engineering. Most important assumptions that KF violates are: (a) the theory is proven for linear systems only. Since all engineering systems are designed to satisfy the boundedness law, which forces them to become nonlinear, the KF cannot work. (b) Every time slice, the operating system switches the tasks and modifies the environment. The original environment does not exist anymore; when the KF takes over in the next time slice. The implementation of KF does not know and cannot know how the software and the system have changed the GST. (c) Just like control variable and initial value constant are not known for many reasons, the parameters of the equations are also not known for the same reasons.

KF uses the following mathematical equations:

$$\frac{dx}{dt} = A(t)x(t) + B(t)u(t) \tag{14}$$

$$x(0) = x_0 \tag{15}$$

$$y(t) = C(t)x(t) + v(t) \tag{16}$$

In the above equations x(t) describes all the system state variables, like pressure, flow rate etc. The variable u(t) is the control variable, x_0 is the initial value constant for x(t) at time zero. The symbol y(t) in (16) represents the measurement and the measurement error is represented by the symbol v(t).

The engineering experiences show that not only the control variable u(t) in (14), and the initial condition x_0 in (15) are unknown, the parameters A, B, and C are also unknown, and for the same reasons. There are other KF related matrices like P, Q, and R which are covariance matrices, and are not introduced here for simplicity. But they are also unknown because of similar reasons. In addition the GST effect of the operating system causes more uncertainty for these parameters. Thus the whole KF theory failed to produce results correctly and required lot of kludges in the implementation of software.

The engineers therefore decided to introduce adaptive KF (AKF) that will continuously estimate ABC and PQR matrices. There are many papers that have been published over long period of time on AKF theories. They have given detailed theoretical proofs on the convergence and optimality of the AKF. Some results can be found here [21], and [22]. The very fact that we have started using methods to estimate parameters is a recognition that the KF is not working, and the assumptions of KF are wrong. The AKF did not work also because of the boundedness law and the GST effect. The AKF uses a linear model.

The engineers then extended the theory to nonlinear equations, known as the extended Kalman filter (EKF). So far no one has produced any theoretical proof that EKF is valid. That is, no one has proved that the EKF filtering equations will converge; will provide optimal solution etc. If we use something that does not even have a theoretical foundation under its assumptions, then that theory cannot work at all in embedded engineering. The paper [23] writes that the EKF fails to account for fully nonlinear dynamic system. The paper [24] says for nonlinear noisy systems optimality has not been proven. But these are not enough; the KF must satisfy all four laws of embedded systems to qualify for its usability; and it does not.

4.7 Differential Equations

Now we show that the theory of differential equations (DE) cannot work in our technology. The linear differential equations like (14-16) cannot work in any embedded engineering systems, because all embedded systems are nonlinear; since by design systems implement the boundedness law. All digital systems are discrete in both time and in space. Therefore, there cannot exist the concept of limit, continuity, and derivatives, and so no nonlinear DE will be meaningful for embedded engineering.

When we approximate derivatives by making it discrete in time and space then we have to create complete theory for their existence, uniqueness, convergence, optimality etc. for all nonlinear discrete systems that include the boundedness law. There exist some theories, only in a limited sense, for continuous time systems [25], but not for discrete cases. We have expressed these concerns while discussing EKF case. If there is no theory then that idea should not be used in any applications. It will create more confusion, more problems, more patches, kludges, and provide poor performances.

Today our technology is processing communication systems even at 10 or 20 GHz rate using digital technology [26] and using parallel processing methods. As the technology evolves we may see even faster rate sampling of analog signals and use of digital techniques. Although it is possible to implement the digital ideas in [26] using the theory described in [4] and implementing by analog means as presented in [27], it is still not conceivable that there will be a need for DE in embedded systems in near future.

We should also realize that the derivative is actually a law of conservation in disguise [18]. The derivative says that a small change in one thing will produce a small change in another thing. According to the action-reaction law, these two changes must be same also. Therefore Δy, a small reaction in something, say y, divided by Δx, a similarly small action on something else, say x, must always be one. We can then write as in (17):

▪
$$\frac{\Delta y}{\Delta x} = 1 \qquad (17)$$

We are neglecting the sign for simplification, and without loss of generality. This ratio is called the derivative. When these delta values are very small, the ratio is denoted as dy/dx, and can be represented as in (18).

▪
$$\frac{dy}{dx} = 1 \qquad (18)$$

But in the theory of differential equations the derivatives are not always equal to one. There are many reasons for that. This happens mainly because x and y are normally not the same kind of variables or do not have the same units. If we convert both variables to the same physical unit, then the derivative will always be equal to one, because this was derived from the action reaction law (8), the third law of Newton. Thus in all of our modeling approach we should always ensure that (18) holds. Therefore by introducing differential equation we cannot get any new characteristics in our models. Best thing will be to follow the law of conservation moment by moment.

A theory to control continuous time dynamic systems using DE and with boundedness nonlinearity can be found in paper [28] and in [3]. The idea in the paper definitely solves an important class of embedded engineering problems. The paper replaces the nonlinear function of the boundedness law by a smooth tanh trigonometric function and then uses Hamilton Jacobi Bellman method to solve the control problem. However, a realistic implementation of the idea for a complex embedded project operating under multitasking operating system is yet to be demonstrated. This author does not feel confident that it can be implemented for discrete systems using a kludge free software.

4.8 Measurable Functions

Many theories in math, science, and engineering use the assumption of piece wise continuous (PWC) functions. These functions have discrete jumps in certain places over their domain. There are no piece wise continuous functions in any embedded engineering application. If you put an oscilloscope probe on any pin of a digital microprocessor you will find a continuous signal. Thus even in digital electrical engineering there are no PWC functions.

Measurable functions are another step extension of PWC in the abstract direction. Most common example found in textbooks is the following:

▪

$$f(x) = \begin{cases} 1 & x \text{ is rational} \\ 0 & x \text{ is irrational} \end{cases} \quad \forall\, x \in [0,1] \tag{19}$$

If PWC is nonexistent, then how can (19) be useful in engineering? This kind of functions has been used as foundations of measure theory to extend the theory of integration, theory of differential equations, stochastic process [29] etc. The concept of measure theory provides a foundation of probability theory also.

We have seen that the KF, which is based on probability theory, does not work. No matter how sophisticated theory we make using measureable functions, at the core of it we must find the probability of every event of the process. These events are defined using the set theoretic concept of sigma algebra [30]. It is not possible to find probability of events in a sigma-algebra for a real time engineering systems. No product will be reliable if it is based on such chance events, with assumptions. As mentioned before, nature and therefore engineering do not make any assumptions. Moreover probability is a linear measure also and engineering is not. We must extend mathematics downwards for its feasibility in engineering, and not upwards in abstract imaginary fields.

5: PROPOSED DESIGN

We have taken several standard and commonly used theories to show that none of them obey the new laws of nature identified in this paper. If we pick any theory from our existing database of theories we will be able to show that none of them will satisfy the new laws. Thus we will make our software unreliable; engineering will remain vulnerable, when we use such theories.

As mentioned before the goal of this paper was to point out the root cause of the reliability problem of embedded systems, however we want to venture little bit into the idea of an approach for the alternative. The suggested approach in this section is quite dramatic, and yet very natural and taken from human activities. It is very difficult to think radical in our community and abandon all existing mathematical methods and go for a new approach. However, we believe that this approach will become a common practice in days ahead, as the sensor technology becomes cheaper, smaller, and better.

If we know that the existing theories cannot work, because they violate all the laws, then why use any kind of theory to begin with. Best approach in engineering will be to use as much measurements as possible, and sample as fast as possible [10], and use them to make decisions at the next sample time. It will be always possible to implement such a design in all engineering problems; we just have to rethink it using a global view of our requirements and technology. Elimination of all theories that violate the laws, will significantly improve the quality of our embedded software. A 10 GHz high speed digital communication system has been proposed [27] to implement such a scheme without using any conventional adaptive algorithms.

The lessons from Kalman's idea are very profound, and that is why we call it a philosophy in this paper. Kalman actually wanted to present his philosophy and provide its theoretical foundations with the help of a linear model. It says use the measurements to estimate what you want. We humans do not use math and science when we walk; we take measurements constantly using all our sensors, like eyes, ears etc., to make our ways around correctly. We have the technology and do not need to use inconsistent theories of math and science in our engineering products. After all our KF experiences show that we cannot find ABC and PQR parameters. This measurement only approach may even reduce the cost of development and maintenance of the product. Engineering will become simpler also, and will be very robust.

Our technology may not be able to provide high quality sensors today at low prices. Under these circumstances we should take advantage of the simultaneity and complexity laws by using multiple sensors at multiple locations and processing them simultaneously using simple algebra. It is also very important to rethink the requirements and isolate it carefully from underlying design assumptions, which often include mathematics and science as its foundation. In the design process, while working at the details level, we must repeatedly come back to our high level thought processes to look at the requirements, in order to avoid such math and science theories.

In summary, we use theories only because we do not have enough measurements, the measuring devices are not small, and are expensive. To compensate for the lack of these measurement data, we use unproven theories and patch work to extrapolate and interpolate data. If we rethink engineering then we will invest and produce better, smaller, and low cost sensors to make large set of simultaneous and interactive measurements thus eliminating the theories completely.

6. CONCLUSIONS

We have shown inconsistencies in many of our mathematical and scientific theories with the natural laws of embedded systems. If we take any theory from math and science, and then look at it using the embedded laws, we will be able to show its invalidity. Because of these problems, these theories cannot be used in embedded engineering. If we use them then we will be forced to add many patches and kludges in our embedded real time software to make engineering work and that will make our software very unpredictable, difficult to maintain, test and debug. It is possible to rethink the entire engineering architecture and design problem using the Kalman philosophy of taking measurements, and using very simple algebraic theories.

7. REFERENCES

[1] M. Guido. and B. Andrich. "Metric suite for directing the failue mode analysis of embedded software systems", ICEIS, 2005

[2] H. Kopetz. "The complexity challenge in embedded system design", Research report 55, Austria, 2007

[3] S. E. Lyshevski. *Control system theory with engineering applications.* Boston, USA: Birkhauser, 2001, Chapter 4.

[4] S. Das, N. Mohanty, and A. Singh. "Function modulation – the theory for green modem", pp. 121-143, International journal on advances in networks and services, vol. 2, no. 2&3, 2009

[5] R. Boynton."Precise measurement of mass", Soc. allied weight engineers, Arlington, Tx, 2001.

[6] Discovery,(2007), Neurons, How they work, Discovery channel video, http://www.youtube.com/watch?v=c5cab4hgmoE

[7] C. Halls. et al. "The need to reintegrate the natural science into economics", available: http://dieoff.org/page228.pdf , 2001

[8] L. Debnath. *Integral transforms and their applications*, CRC press, 1995

[9] S. Das. "Finite time engineering", WCECS, San Francisco, October 19-21, 2011

[10] S. Das. Mohanty, N. & Singh, A.: "Is the Nyquist rate enough?", ICDT, IEEE Computer society, available: IEEE Xplore digital library, 2008

[11] R. Ferraro. *Einstein's Space-time*, Springer, 2007

[12] A. B. Chatfield. *Fundamentals of high accuracy inertial navigation.* Virginia, USA: Progress in Astronautics and Aeronautics, 1997, Volume 174.

[13] R. M. Roger. *Applied mathematics in integrated navigation systems*, second edition, Virginia, USA: AIAA, 2003, p. 76.

[14] R. Christensen. and N. Fogh. *Inertial navigation systems*, masters project, department of electronic systems, Aalborg University, Denmark, available: http://www.control.aau.dk/uav/reports/08gr1030a/08gr1030a_student_report.pdf , 2008.

[15] R. A. Serway. and J. W. Jewett. *Physics for scientists and engineers with modern physics*, Sixth edition, USA: Thomson, 2004.

[16] M. Nylund. A. Brown. and J. J. Clark. "Kinematic GPS-Inertial navigation on a tactical fighter", Proc. ION GPS, Oregon, USA, 2003

[17] M. J. Pinheiro. "On Newton's third law and its symmetry breaking effects", Phys. Scr. 84 055004, 2011

[18] S. Das. "Laws of conservation", Intellectbase academic conference, Las Vegas, Dec 15-17, 2011

[19] D. Rowel. "Linear graph modeling: state equation formulation", MIT, Department of mechanical engineering, advanced system dynamics and control, available: http://web.mit.edu/2.151/www/Handouts/EqFormulation.pdf , 2004

[20] M.S. Grewal. and P.A. Andrews. *Kalman filtering: theory and practice*, NY: John Wiley, 2001.

[21] M. Oussalah. and D. J. Schutter. "Adaptive kalman filter for noise identification", 25th international conference in noise and vibration engineering, Belgium, 2000.

[22] K. Xiong. et al. "Adaptive robust extended kalman filter", From the book – Recent advances and applications, I-Tech, Vienna, Austria, 2009

[23] S. Gillijns. et al. "What is the ensemble Kalman filter and how does it work?", Proc. Am. Cont. Conf., June, MN, USA, 2006

[24] M. Boutayeb. et al. "Convergence analysis of the extended kalman filter as an observer for non-linear discrete-time systems", Proc. 34th. Conf. Dec. & Con., New Orleans, LA, December 1995

[25] H. K. Khalil. *Nonlinear systems*, third edition, New Jersey, USA: Prentice hall, 2002.

[26] Broadcom, US Patent, "Electronic dispersion compensation utilizing interleaved architecture and channel identification for assisting timing recovery". US Patent number – 7830987, Nov 9, 2010

[27] S. Das. "A new analog design for the EDC family", Vitesse, California, USA, internal proprietary and private document, 2011

[28] S. Lyshevski. "Constrained optimization and control of nonlinear systems: new results in optimal control", 35 th conference on decision and control, Kobe, Japan, 1996

[29] W. Wong. And B. Hajek. *Stochastic process in engineering systems.* NY, USA: Springer-Verlag, 1985.

[30] P. R. Halmos. *Measure Theory*. NY, USA: Springer-Verlag, 1974.

Performance Testing and Comparison of A Turbine Ventilator, A Vent Column, and Their Combination Under Thermal Buoyancy and Wind Effects

Nguyen Q. Y *quocyn@gmail.com*
Department of Civil Engineering
HCMC University of Technology
HoChiMinh City, Vietnam

Nguyen T. Bay *nguyentbay@gmail.com*
Department of Civil Engineering
HCMC University of Technology
HoChiMinh City, Vietnam

Ha Phuong *haphuong852001@yahoo.com*
Department of Civil Engineering
HCMC University of Technology
HoChiMinh City, Vietnam

Abstract

Ventilation performance of a curved-blade turbine ventilator, a straight column covered with a flat hat, and a device of their combination of the same material and throat diameter of 21cm were tested on a room model of 3.0m long, 1.5m wide, and 3.0m high under simulated external wind and/or internal heat source. The wind speed was from 0m/s to 3.6m/s. The heat flux was up to 3KW. Air speed through each device was measured and plotted as functions of both the wind speed and the heat flux. The results show that when buoyancy effects were dominant, i.e. internal heat source under low wind speed, the column performed best, followed by the combined device and lastly the turbine. When wind effects were dominant, the combined device worked best, followed by the turbine which was close to the column. Performance of the column was seen to suffer from the external wind while that of the turbine and the combined device benefited from it. Performance of the combined device was found to be better than that of the turbine due to stack effects gained by an increased throat height compared to the turbine's. This observation suggests a simple modification to boost performance of current commercial low-throat turbine ventilators.

Keywords: Turbine Ventilator, Vent Column, Wind Effects, Buoyancy Effects, Ventilation Performance.

1. INTRODUCTION

Saving energy and utilizing natural, green energy resources are one of the key concerns of the world presently. In buildings, energy can be saved by employing natural methods for ventilation to replace electric fans or air conditioners. Natural ventilation strategies or devices are based on two main effects (Khan et al 2008a, Linden 1999, Awbi 2003): pressure induced by external wind and stack effects induced by temperature difference.

Among natural ventilation devices, wind-driven turbine ventilator is the most common one. The device works both as a turbine driven by the external wind and as a fan withdrawing air from buildings' interior (Khan 2008a, Khan 2008b, Lai 2003). Though this device has been used for long time, it still attracts research interests, particularly to compare its performance to other devices' in enhancing ventilation (Lai 2003, Revel & Huynh 2004, Khan 2008b). Lai (2003) tested turbines with different sizes, and ventilation performance of a turbine with blades fixed or removed (thus forming a simple hole) under external wind. He found that the stationary turbine with fixed blades worked identically to when the blades were removed, but poorly compared to when it was free to rotate. Revel & Huynh (2004) and Khan et al (2008b) compared flow rate through turbines and simple straight columns covered with plates of flat or conical shapes under wind effects. They reported the column with the flat cover gave the ventilation flow rate almost equivalent to that of the curved blade turbine of the same throat size.

To enhance performance of turbine ventilators, Lai (2006) tested a prototype of a turbine using combined wind and photovoltaic energy by installing a fan inside the turbine. His results showed that the fan helped to increase ventilation rate under low wind speed.

In this study, ventilation performance of a turbine ventilator is tested and compared with a simple vent column and another device, which is made from combination of the turbine and the column, under both wind and buoyancy effects.

Our work is motivated from two facts. Firstly, as reported by Revel & Huynh (2004) and Khan et al (2008), a simpler, hence cheaper, straight column can match performance of a more complicated, hence costlier, turbine ventilator under wind effects. Secondly, in Vietnam, it is advertised as well as believed by the majority of people that turbine ventilators can help withdraw hot air from inside of buildings or factories even without external wind. A question was then raised to us about actual performance of the turbine, particularly when it is compared to the simple straight vent column under effects of external wind and large temperature difference between inside and outside buildings. To our knowledge, such tests have not yet been reported in the literature.

In addition, it is also interesting to combine the turbine and the column to benefit from the strength of each device: rotating effects of the turbine blade under external wind and stack effects of the column under large outside-inside temperature difference. Therefore, the third device was formed by adding the column to the turbine.

2. EXPERIMENT

Experimental setup is described in Figure 1.

The experiments were to simulate practical situations where the turbine was placed on top of a building or a room for ventilation. External wind flowed only over the turbine rather than over the whole building. With heat sources inside the building, e.g. cookers, computers in dwellings or working machines in factories, there was large temperature difference between inside the buildings and ambient air.

To simulate a ventilated room, a room model was built with the size of (length x width x height) 3.0m x 1.5m x 3.0m, which was similar (in height and length) to those of a small real room. The model was made of wood for good heat insulation properties. Air entered the model through an inlet of 0.8m x 0.6m and exited through the tested devices on the top of the model.

External wind was simulated using a 50W electrical fan with the diameter of 40cm. Elevation of the fan, positioned on top of the room to make flow covering the tested devices but not the whole room, was adjusted to match that of each tested device. Metal screens were placed between the fan and the tested devices to reduce the rotating velocity component of the wind created by the fan and to increase its uniformity. Wind speed was adjusted by two means: adding electrical resistances connected serially to the fan and/or putting more screens.

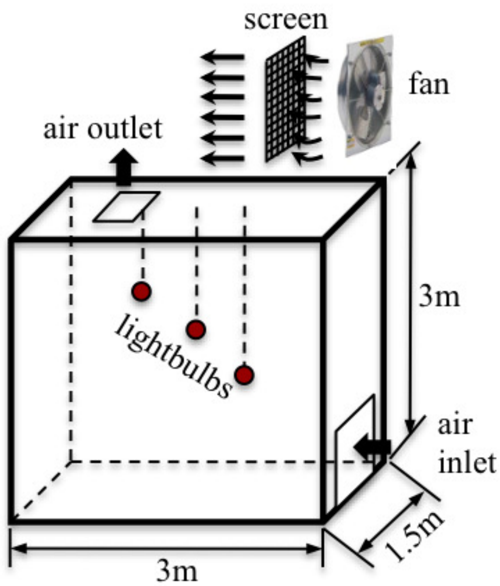


FIGURE 1: Sketch of the experimental setup.

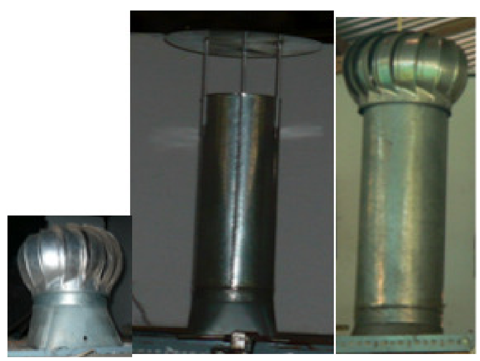


FIGURE 2: Three devices under test: (from left to right) turbine ventilator, straight column, and the combined turbine-column device.

For simulating internal heat sources, electric light bulbs of 200W each and distributed in the center plane of the room model were used. The heat flux was controlled by the number of lightbulbs turned on in each experiment.

With these apparatuses, the setup was capable of producing maximum wind speed of 3.6m/s, which was about the average wind speed in Hochiminh City, and maximum heat flux of 3kW, which yielded maximum inside-outside temperature difference of about 15^{o}C. Higher wind speed would require faster and bigger fans; hence more complicated systems to remove the wind's rotational velocity as well as to maintain its uniformity (for example, using a honeycomb). Higher heat flux would reduce the heat insulation ability of the model walls when the walls became hot quickly and large portion of heat was transferred to ambient environment.

Tested devices consisted of a turbine ventilator, a vent column, and a device of their combination, as shown in Figure 2. All of the three devices were made of the same metal material. The turbine had 16 curved blades with the throat diameter of 21cm and the height from the base to the throat of 18cm. The vent column was made by removing the blades of the turbine and connecting a circular tube to the throat to form the total height of 68cm. This height was over the minimum required value of 0.5m to prevent back draught (Awbi, 2003). The column was covered with a plane cap with the diameter of 42cm, or two times of the column's, and the distance from the top of the column is 15cm, or three quarters of the column diameter. The dimensions of the cap were selected after Khan (2008b). The combined device was made by replacing the cap of the column by the turbine blades. Consequently, all three devices had the same throat diameter of 21cm.

Speed and temperature of the air ventilated from the room model and through each device were measured under three conditions: external wind alone, internal heat source alone, and external wind coupled with internal heat source. Location of the measurement point was the same for three devices, namely at center of the throat of them. The external-wind speed was measured at a distance of 20cm in front of the devices. The speed of the air ventilated from the room model, the speed of the external-wind, and the temperature of the air were measured by a Kanomax A041 hotwire anemometer with the speed resolution of 1cm/s and the temperature resolution of 0.1^{o}C. Ambient air temperature was also read on an alcohol thermometer with a resolution of 0.25^{o}C. Testing range of the external wind speed was from 0m/s to 3.6m/s, and of the internal heat flux was 1KW, 2KW, and 3KW (equivalent to 5, 10 and 15 light bulbs turned on). Ambient temperature varied from 26^{o}C to 34^{o}C while internal temperature was from 26^{o}C to 50^{o}C.

3. RESULTS AND DISCUSSIONS

Figure 3 shows the air speed through the turbine's throat V_t for five test sets: external wind alone and the turbine was fixed intentionally (0KW-fixed); external wind alone and the turbine was free to rotate (0KW); wind coupled with 1KW, 2KW, and 3KW internal heat flux (1KW, 2KW, 3KW, respectively). V_t is seen to increase proportionally with both the wind speed and heat flux. In the test case 0KW, the free turbine did not rotate when the wind speed is 0m/s and 0.5m/s, and was identical to the fixed one. When the wind speed increased, the air speed through the free turbine was approximately twice of that of the fixed one. Therefore, hindered turbine ventilators, which are very common for the ones serving long time without maintenance, has very poor ventilation performance. This observation was also reported by Lai (2003). His results showed that a turbine with blades kept stationary could only produced ventilation flowrate of about one third of that when the blades were free to rotate.

It is noted that in the case of no wind and no heat (0m/s, 0KW) there still exists air speed through the turbine. This is believed to be due to the temperature difference caused by solar heat absorbed by the metal roof of the laboratory. Since all tests were done in nearly the same ambient conditions, this natural temperature difference affected all three tested devices similarly and thus should have negligible influence on the comparison of their performance.

With heat added (1KW, 2KW, and 3KW), but without external wind (0m/s), the turbine was still able to rotate with the air flow through it induced by the internal heat source. If the turbine was held stationary deliberately in these cases, the air speed through its throat decreased, as indicated by data point named "fixed" in Figure 3. From this observation, it can be interpreted that the freely rotating turbine, though received energy from the air flow to rotate, consumed less energy than the energy lost through the fixed turbine.

Khan (2008b) reported that turbines obey the fan law: the flow rate is proportional to the rotation speed. In our tests, as the wind speed increased and more heat was added, the turbine spun faster. Accordingly, the air speed should increase with both the wind speed and the heat flux, as seen in Figure 3.

In the case of combined wind-buoyancy effects, the empirical formula proposed by Walker and Wilson (Awbi 2003) allowed estimating the combined air speed:

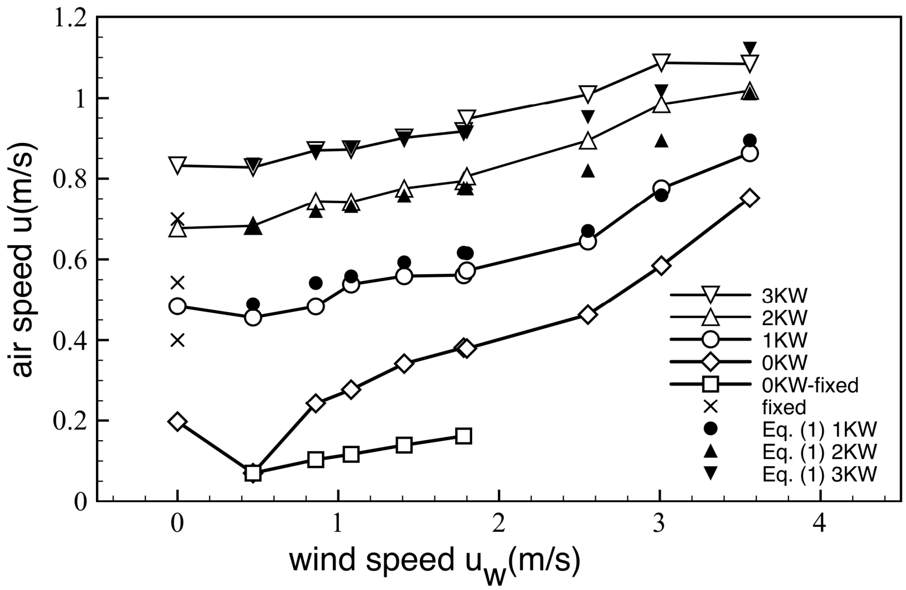


FIGURE 3: Air speed through the turbine ventilator.

$$u = \sqrt{u_w^2 + u_b^2} \tag{1}$$

where u_w, u_b, and u are the wind induced air speed, buoyancy-induced air speed, and the air speed under combined effects, respectively.

Eq. (1) was applied to estimate the air speed u for test cases of 1KW, 2KW, and 3KW from non-zero wind speed using u_w at 0KW and u_b at 1KW, 2KW, and 3KW but zero-wind speed, respectively. The results are also plotted in Figure 3, and show that Eq. (1) matches the measured data quite well, with maximum discrepancy about 10%.

To check reliability of the data, two test sets were conducted at wind speed of 1.8m/s which was obtained by two different combinations of the electrical resistances connected serially to the fan and the number of metal screens. Two separate data points of each test case (0KW, 1KW, 2KW and 3KW) at the wind speed of 1.8m/s in Figure 3 represent the results from this two test sets. Good reproductivity can be seen.

Figure 4 shows the air speed V_c through the column under combinations of the external wind and the internal heat source. As the heat flux increased, similar to the turbine, air speed increased accordingly. However, as the external wind speed increased, with and without the heat source, V_c seemed to vary in two regions. When wind speed was below about 1.3m/s, V_c was nearly constant. When the wind speed increased above 1.3m/s, in the test case of 0KW, V_c increased proportionally; in the test case of 1KW, V_c was nearly constant; but in the test case of 2KW and 3KW, V_c decreased.

From these observations, it can be seen that the external wind with speed below 1.3m/s had negligible effects on the air flow. Above this speed, with the wind alone, the airflow seems to benefit from negative pressure induced by the external wind on the top of column, hence V_c increased accordingly. With strong internal heat sources (test cases 2KW and 3KW), the air flow may be obstructed by the external wind with possible flow separation on the top of the column, hence effective area of the air flow was reduced and V_c decreased accordingly. With weak internal heat source (test case 1KW), air speed due to the heat alone was weak. Therefore, with an external wind, the air flow through the device was proportionally less obstructed by the wind flow separation, since the energy loss of the air flow was proportional to the square of its speed. In addition, in this test case, the nearly constant V_c at upper range of the wind speed can be explained from the balance between the reduction of the airflow area mentioned above and the negative pressure induced by the external wind.

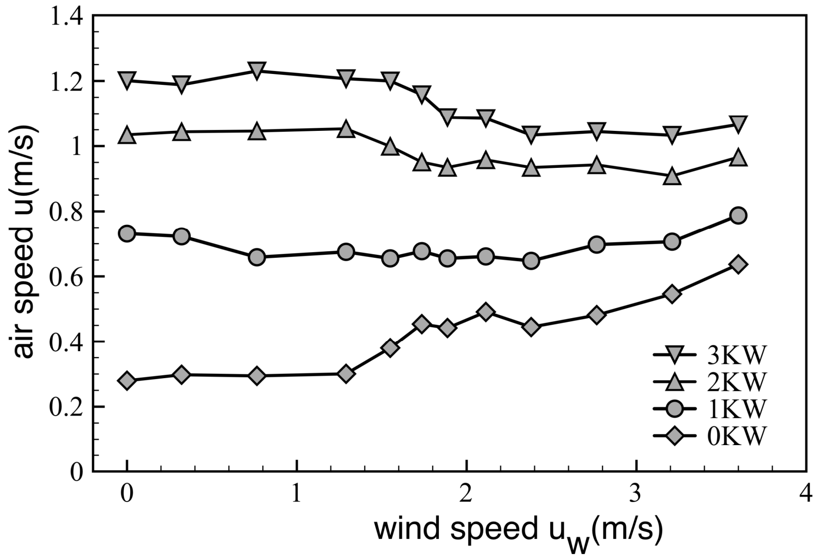


FIGURE 4: Air speed through the vent column.

As seen, for the column, wind effects did not assist buoyancy to enhance the air flow through the device. This is in contrast to Eq. (1). Consequently, Eq. (1) is not applicable to predict the combined wind-buoyancy effects on air flow through the column.

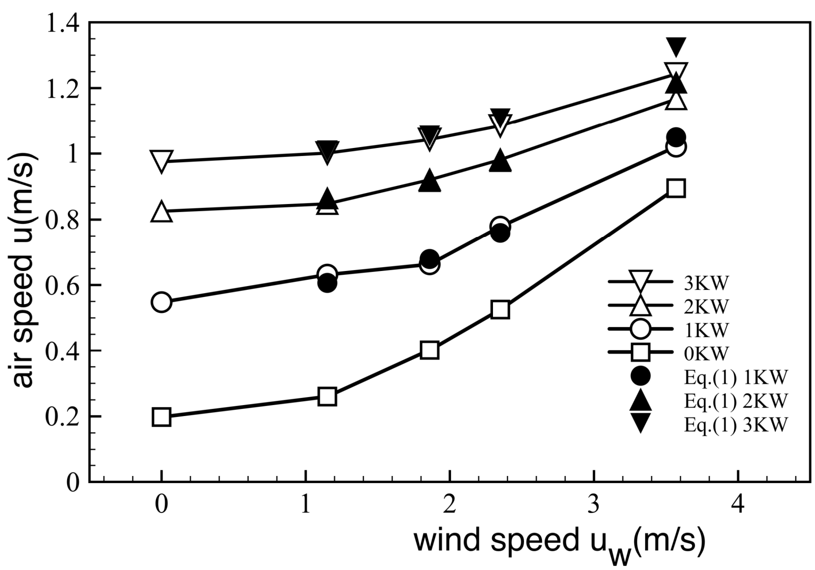


FIGURE 5: Air speed through the combined device.

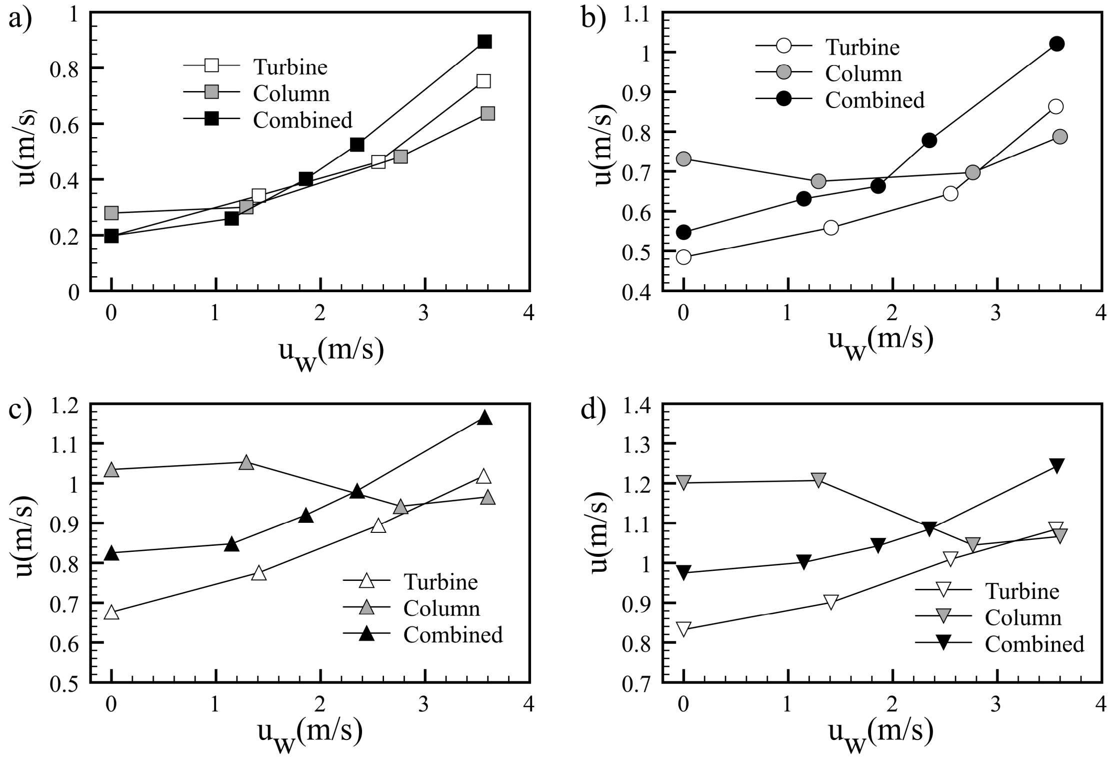


FIGURE 6: Air speed through three devices under different wind speed and heat flux: a) 0KW, b) 1KW, c) 2KW, d) 3KW

Figure 5 shows variation of the air speed V_{tc} through the turbine-column combined device. The variation is seen to be quite identical to that of the turbine; namely the air speed V_{tc} increased as both the wind speed and the heat flux increased. Eq. (1) was also applicable; the predicted data are also plotted in Figure 5. Agreement between the measured and the predicted data is seen to be even better than the case of the turbine.

To compare the performance of the three devices, their results are plotted together in Figure 6. With the wind alone (Figure 6a), performance of the turbine and the column are seen identical, except that at wind speed u_w=3.6m/s, the turbine gave a little more air speed. The combined device, however, outperformed the other two at the upper range of the wind speed, about from 2m/s and up.

Comparisons of the turbine and the column found in this test case are similar to the results by Khan (2008b). That author reported that a 250mm column with a flat cover, which was the design used in this study, "did manage to almost keep up" with a 250mm turbine up to a wind speed of 4.1m/s.

With the combined wind-heat (1KW, 2KW, and 3KW), performance of the turbine catches up to the column's when wind speed was over about 3m/s. The combined device always performed better than the turbine at any heat flux and wind speed, and could only match the column at the wind speed of about 2.2m/s, and then demonstrated better performance afterward.

The combined device should benefit from stack effects that the column possessed but the turbine did not, and fan effects that the turbine had but the column did not. Meanwhile, it also suffered from the energy loss caused by the turbine. Consequently, performance of the combined device should be enhanced by the stack effects. On the other hand, since the stack effects mainly depend on the internal heat flux, performance line of the combined device at a specific heat flux is expected to run parallel to that of the turbine, as seen in Figure 6a, 6b, and 6c.

With the external wind alone, there is no stack effect. Accordingly, the performance of the combined device and of the turbine should be the same under the same fan effects. However, Figure 6a shows that at wind speed above 2m/s, the combined device drew more air. Reasons of this are not clear to the authors.

To seek a universal description of the results with the combined wind-heat effects, nondimensional parameters $u^*=u/(u_w+u_b)$ and $1/Ar$ (Etheridge 2002) are used and plotted in Figure 7, where:

- u_w: external wind speed (m/s).

- $u_b=\sqrt{\Delta Tgh/T}$: buoyancy speed (m/s), in which T is the ambient temperature (^oK) and ΔT is the temperature difference between the ambient temperature (inlet temperature) and the air temperature at the outlet (^oK).

- $1/Ar=\rho u_w/\Delta\rho gh$: reciprocal of the Archimedes number, in which ρ is the ambient air density (kg/m^3) and $\Delta\rho$ is the air density difference between the inlet and the outlet (kg/m^3). ρ and $\Delta\rho$ are determined from the air temperature at the inlet and the outlet.

Wind effects are negligible for very small value of $1/Ar$ and buoyancy effects are negligible for very large value of $1/Ar$. Following Etheridge (2002), the test cases of wind alone are assigned to $1/Ar=100$ and the test cases of heat alone are assigned to $1/Ar=0.01$. At $1/Ar=0.01$, values of $u^*=u/u_b$ (as $u_w=0$), are averaged ones from data of three tests of 1KW, 2KW and 3KW for each device, and at $1/Ar=100$, values of $u^*=u/u_w$ (as $u_b=0$), are those corresponding to tests with maximum wind speed but without heat source, i.e. (u_w=3.6m/s, 0KW) for each device.

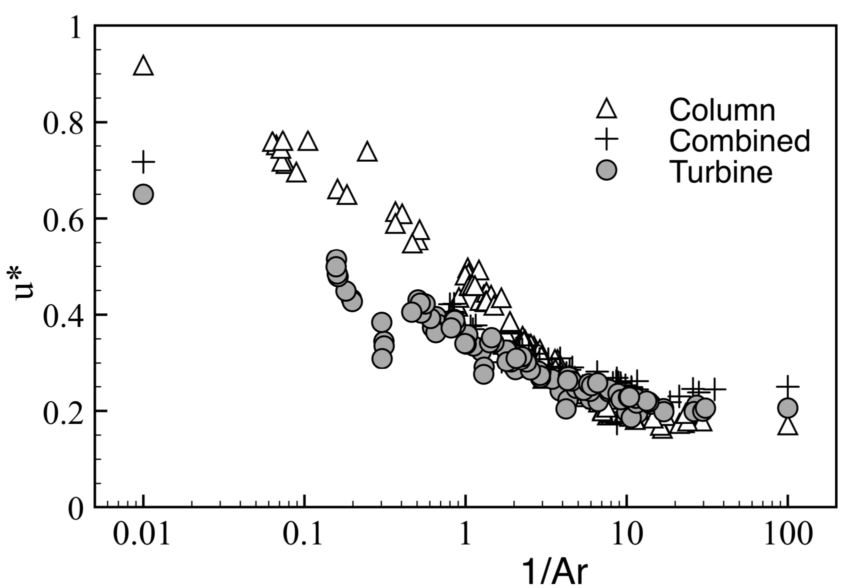


FIGURE 7: Nondimensional air speed through three devices against 1/Ar.

Figure 7 reconfirms that when the buoyancy effects are dominant, the column performs best, followed by the combined device and then the turbine. When wind effects are dominant, the combined device performs best, followed by the turbine and then the column. In addition, from the trend of the data curves in Figure 7, it seems that ratios between the air speed and the external wind speed approach to about 17%, 20%, and 25% for the column, the turbine, and the combined device respectively as the wind effects become strong.

From the above observations, with the purposes of maximum ventilation rate, for cases with strong buoyancy effects (high temperature difference and low wind speed), the column is recommended. For cases with strong wind effects (high wind speed and low temperature difference), the combined device is recommended. For general cases where both effects exist, the combined device should be the best solution.

Since the combined device is actually the turbine with higher throat, enhancing the throat height of an ordinary short-throat turbine should boost its ventilation performance under combined wind-buoyancy effects. This point may be useful for upgrading available short-throat turbines, since replacing their throat is simple, and should be considered for manufacturing new turbine ventilators.

4. SUMMARY

Ventilation performances of three devices, namely a turbine ventilator, a vent column, and a device made by their combination, all with the same throat area, were tested and compared under the effects of wind alone, buoyancy alone and coupled wind-buoyancy experimentally. Compared values were the air speed through the devices. When buoyancy effects were dominant, the column, though the simplest device, was the best. When wind effects were dominant, the combined device performed best. Under coupled wind-buoyancy effects, the combined device was seen to benefit from stack effects of the column to offer higher ventilation air speed than the turbine but still could not match the performance of the column.

From the experiments, two conclusions can be seen. Firstly, the turbine does not help to withdraw hot air from inside buildings as much as the column does, particularly when there is no or little external wind, though the former is much more expensive and advertised to do so effectively. Secondly, with the height of its throat increased, the turbine was seen to perform better. Therefore, ventilation performance of current commercial low-throat turbine ventilators can be boosted by simple modifications of increasing their throat height.

ACKNOWLEDGEMENT

The authors thank: Dr. Huynh, Ba Phuoc at Sydney University of Technology for helpful comments; colleagues at the laboratory of Fluid Mechanics, HCMC University of Technology for helps in making the experimental setup; research fund from VNU-HCM.

REFERENCES

[1] H. Awbi. Ventilation of buildings. London and New York: Spon Press, 2003

[2] D.W. Etheridge. “Nondimensional methods for natural ventilation design”. Building and environment, vol. 37, pp. 1057-1072, 2002.

[3] N. Khan, Y. Su, S. B. Riffat. “A review on wind driven ventilation techniques*”,* Energy and Buildings, vol. 40, pp. 1586-1604, 2008 (a).

[4] N. Khan, S. Su, S.B. Riffat, C. Biggs. “Performance testing and comparison of turbine ventilators”. Renewable Energy, vol. 33, pp. 2441-2447, 2008 (b).

[5] C.M. Lai. “Experiments on the ventilation efficiency of turbine ventilators used for building and factory ventilation”. Energy and Buildings, vol. 35, pp. 927-932, 2003.

[6] C.M. Lai. “Prototype development of rooftop turbine ventilator powered by hybrid wind and photovoltaic energy”. Energy and Buildings, vol. 38, pp. 174-180, 2006.

[7] P. F. Linden. “The fluid mechanics of natural ventilation”. Annual review of fluid mechanics, vol. 31, pp. 201-238, 1999.

[8] A. Revel, B.P. Huynh. “Characterizing roof ventilators”. Presented at 15th Australasian Fluid Mechanics Conference, The University of Sydney, Sydney, Australia, 2004.

Methodology of Mathematical Error-Based Tuning Sliding Mode Controller

Farzin Piltan SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic Engineering SanatkadeheSabze Pasargad. CO (S.S.P. Co), NO:16 ,PO.Code 71347-66773, Fourth floor Dena Apr , Seven Tir Ave , Shiraz , Iran

Bamdad Boroomand SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic Engineering SanatkadeheSabze Pasargad. CO (S.S.P. Co), NO:16 ,PO.Code 71347-66773, Fourth floor Dena Apr , Seven Tir Ave , Shiraz , Iran

Arman Jahed SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic Engineering SanatkadeheSabze Pasargad. CO (S.S.P. Co), NO:16 ,PO.Code 71347-66773, Fourth floor Dena Apr , Seven Tir Ave , Shiraz , Iran

Hossein Rezaie SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic Engineering SanatkadeheSabze Pasargad. CO (S.S.P. Co), NO:16 ,PO.Code 71347-66773, Fourth floor Dena Apr , Seven Tir Ave , Shiraz , Iran

Abstract

Design a nonlinear controller for second order nonlinear uncertain dynamical systems is one of the most important challenging works. This paper focuses on the design of a chattering free mathematical error-based tuning sliding mode controller (MTSMC) for highly nonlinear dynamic robot manipulator, in presence of uncertainties. In order to provide high performance nonlinear methodology, sliding mode controller is selected. Pure sliding mode controller can be used to control of partly known nonlinear dynamic parameters of robot manipulator. Conversely, pure sliding mode controller is used in many applications; it has an important drawback namely; chattering phenomenon which it can causes some problems such as saturation and heat the mechanical parts of robot manipulators or drivers.

In order to reduce the chattering this research is used the switching function in presence of mathematical error-based method instead of switching function method in pure sliding mode controller. The results demonstrate that the sliding mode controller with switching function is a model-based controllers which works well in certain and partly uncertain system. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. To solve this problem applied mathematical model-free tuning method to sliding mode controller for adjusting the sliding surface gain (λ). Since the sliding surface gain (λ) is adjusted by mathematical model free-based tuning method, it is nonlinear and continuous. In this research new λ is obtained by the previous λ multiple sliding surface slopes updating factor (α). Chattering free mathematical error-based tuning sliding mode controller is stable controller which eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in mathematical error-based tuning sliding mode controller with switching (sign) function. This controller has acceptable performance in presence of uncertainty (e.g., overshoot=0%, rise time=0.8 second, steady state error = 1e-9 and RMS error=1.8e-12).

Keywords: Nonlinear Controller, Chattering Free Mathematical Error-based Tuning Sliding Mode Controller, Uncertainties, Chattering Phenomenon, Robot Arm, Sliding Mode Controller, Adaptive Methodology.

1. INTRODUCTION

The international organization defines the robot as “an automatically controlled, reprogrammable, multipurpose manipulator with three or more axes.” The institute of robotic in The United States Of America defines the robot as “a reprogrammable, multifunctional manipulator design to move material, parts, tools, or specialized devices through various programmed motions for the performance of variety of tasks”[1]. Robot manipulator is a collection of links that connect to each other by joints, these joints can be revolute and prismatic that revolute joint has rotary motion around an axis and prismatic joint has linear motion around an axis. Each joint provides one or more degrees of freedom (DOF). From the mechanical point of view, robot manipulator is divided into two main groups, which called; serial robot links and parallel robot links. In serial robot manipulator, links and joints is serially connected between base and final frame (end-effector). Most of industrial robots are serial links, which in n degrees of freedom serial link robot manipulator the axis of the first three joints has a known as major axis, these axes show the position of end-effector, the axis number four to six are the minor axes that use to calculate the orientation of end-effector and the axis number seven to n use to reach the avoid the difficult conditions (e.g., surgical robot and space robot manipulator). Dynamic modeling of robot manipulators is used to describe the behavior of robot manipulator such as linear or nonlinear dynamic behavior, design of model based controller such as pure sliding mode controller and pure computed torque controller which design these controller are based on nonlinear dynamic equations, and for simulation. The dynamic modeling describes the relationship between joint motion, velocity, and accelerations to force/torque or current/voltage and also it can be used to describe the particular dynamic effects (e.g., inertia, coriolios, centrifugal, and the other parameters) to behavior of system[1-10]. The Unimation PUMA 560 serially links robot manipulator was used as a basis, because this robot manipulator is widely used in industry and academic. It has a nonlinear and uncertain dynamic parameters serial link 6 degrees of freedom (DOF) robot manipulator. A nonlinear robust controller design is major subject in this work. Controller is a device which can sense information from linear or nonlinear system (e.g., robot manipulator) to improve the systems performance [3]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[5]. Several industrial robot manipulators are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when robot manipulator works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, parameter variations). In some applications robot manipulators are used in an unknown and unstructured environment, therefore strong mathematical tools used in new control methodologies to design nonlinear robust controller with an acceptable performance (e.g., minimum error, good trajectory, disturbance rejection).

Sliding mode controller (SMC) is a significant nonlinear controller under condition of partly uncertain dynamic parameters of system. This controller is used to control of highly nonlinear systems especially for robot manipulators, because this controller is a robust and stable [11-30]. Conversely, pure sliding mode controller is used in many applications; it has two important drawbacks namely; chattering phenomenon, and nonlinear equivalent dynamic formulation in uncertain dynamic parameter. The chattering phenomenon problem can be reduced by using linear saturation boundary layer function in sliding mode control law [31-50]. Lyapunov stability is proved in pure sliding mode controller based on switching (sign) function. The nonlinear equivalent dynamic formulation problem in uncertain system can be solved by using artificial intelligence theorem or online tuning methodology. Fuzzy logic theory is used to estimate the system dynamic. However fuzzy logic controller is used to control complicated nonlinear dynamic systems, but it cannot guarantee stability and robustness. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and adaption law which this method can helps improve the system's tracking performance by online tuning method [51-61].

Literature Review

Chattering phenomenon can causes some problems such as saturation and heats the mechanical parts of robot arm or drivers. To reduce or eliminate the oscillation, various papers have been reported by many researchers which one of the best method is; boundary layer saturation method [1]. In boundary layer linear saturation method, the basic idea is the discontinuous method replacement by linear continuous saturation method with small neighborhood of the switching surface. This replacement caused to considerable chattering reduction. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode controller with boundary layer to improve the industry application [22]. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, Weng and Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. Control of robot arms using conventional controllers are based on robot arm dynamic modelling. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of robot arms. When the system model is unknown or when it is known but complicated, it is difficult or impossible to use conventional mathematics to process this model[32]. In various dynamic parameters systems that need to be training on-line, adaptive control methodology is used. Mathematical model free adaptive method is used in systems which want to training parameters by performance knowledge. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, adaptive method is applied to sliding mode controller. Mohan and Bhanot [40] have addressed comparative study between some adaptive fuzzy, and a new hybrid fuzzy control algorithm for robot arm control. They found that self-organizing fuzzy logic controller and proposed hybrid integrator fuzzy give the best performance as well as simple structure. Temeltas [46] has proposed fuzzy adaption techniques for VSC to achieve robust tracking of nonlinear systems and solves the chattering problem. Conversely system's performance is better than sliding mode controller; it is depended on nonlinear dynamic equqation. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode controller based on *N* fuzzy based linear state-space to estimate the uncertainties. A MIMO FVSC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a nonlinear system [42]. Yoo and Ham [58]have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. This method can only tune the consequence part of the fuzzy rules. Medhafer et al. [59] have proposed an indirect adaptive fuzzy sliding mode controller to control nonlinear system. This MIMO algorithm, applies to estimate the nonlinear dynamic parameters. Compared with the previous algorithm the numbers of fuzzy rules have reduced by introducing the sliding surface as inputs of fuzzy systems. Guo and Woo [60]have proposed a SISO fuzzy system compensate and reduce the chattering. Lin and Hsu [61] can tune both systems by fuzzy rules. Eksin et. al [83] have designed mathematical model-free sliding surface slope in fuzzy sliding mode controller. In above method researchers are used saturation function instead of switching function therefore the proof of stability is very difficult.

Problem Statements

One of the significant challenges in control algorithms is a linear behavior controller design for nonlinear systems (e.g., robot manipulator). Some of robot manipulators which work in industrial processes are controlled by linear PID controllers, but the design of linear controller for robot manipulators is extremely difficult because they are hardly nonlinear and uncertain [1-2, 6]. To reduce the above challenges, the nonlinear robust controller is used to control of robot manipulator. Sliding mode controller is a powerful nonlinear robust controller under condition of partly uncertain dynamic parameters of system [7]. This controller is used to control of highly nonlinear systems especially for robot manipulators. Chattering phenomenon and nonlinear equivalent dynamic formulation in uncertain dynamic parameter are two main drawbacks in pure sliding mode controller [20]. The chattering phenomenon problem in pure sliding mode controller is reduced by using linear saturation boundary layer function but prove the stability is very difficult. In this research the nonlinear equivalent dynamic formulation problem and chattering phenomenon in uncertain system is solved by using on-line tuning theorem [8]. To estimate the system dynamics, mathematical error-based sliding mode controller is designed. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical error-based tuning. This method is based on resolve the on line sliding surface gain (λ) as well as improve the output performance by tuning the sliding surface slope updating factor (α). Mathematical error-based tuning sliding mode controllers is stable model-free controller and eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in mathematical error-based tuning fuzzy sliding mode controller based on switching (sign) function. Section 2, is served as an introduction to the sliding mode controller formulation algorithm and its application to control of robot manipulator. Part 3, introduces and describes the methodology (design mathematical error-based sliding mode controller) algorithms and proves Lyapunov stability. Section 4 presents the simulation results and discussion of this algorithm applied to a robot arm and the final section is describing the conclusion.

2. THEOREM: DYNAMIC FORMULATION OF ROBOTIC MANIPULATOR, SLIDING MODE FORMULATION APPLIED TO ROBOT ARM AND PROOF OF STABILITY

Dynamic of robot arm: The equation of an *n-DOF* robot manipulator governed by the following equation [1, 4, 15-29, 63-74]:

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

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. This robot manipulator dynamic equation can also be written in a following form [1-29]:

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

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. The dynamic terms in equation (2) are only manipulator position. This is a decoupled system with simple second order linear differential dynamics. In other words, the component $\ddot{q}$ influences, with a double integrator relationship, only the joint variableq_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3, 41-62]:

$$\boldsymbol{\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\}} \quad \textbf{(3)}$$

This technique is very attractive from a control point of view.

Sliding Mode methodology: Consider a nonlinear single input dynamic system is defined by [6]:

$$\boldsymbol{x^{(n)} = f(\vec{x}) + b(\vec{x})u} \quad \textbf{(4)}$$

Where u is the vector of control input, $\boldsymbol{x^{(n)}}$ is the $\boldsymbol{n^{th}}$ derivation of $\boldsymbol{x}$, $\boldsymbol{x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T}$ is the state vector, $\boldsymbol{f(x)}$ is unknown or uncertainty, and $\boldsymbol{b(x)}$ is of known *sign* function. The main goal to design this controller is train to the desired state; $\boldsymbol{x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d{}^{(n-1)}]^T}$, and trucking error vector is defined by [6]:

$$\boldsymbol{\widetilde{x} = x - x_d = [\widetilde{x}, \dots, \widetilde{x}^{(n-1)}]^T} \quad \textbf{(5)}$$

A time-varying sliding surface $\boldsymbol{s(x,t)}$ in the state space $\boldsymbol{R^n}$ is given by [6]:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\tilde{x} = 0 \tag{6}$$

where λ is the positive constant. To further penalize tracking error, integral part can be used in sliding surface part as follows [6]:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\left(\int_0^t \tilde{x}\,dt\right) = 0 \tag{7}$$

The main target in this methodology is kept the sliding surface slope $s(x,t)$ near to the zero. Therefore, one of the common strategies is to find input U outside of $s(x,t)$ [6].

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)| \tag{8}$$

where ζ is positive constant.

If **S(0)>0**$\rightarrow \frac{d}{dt}S(t) \le -\zeta$ **(9)**

To eliminate the derivative term, it is used an integral term from t=0 to t=t_{reach}

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt}S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \tag{10}$$

Where t_{reach} is the time that trajectories reach to the sliding surface so, suppose S($t_{reach} = 0$) defined as

$$0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta} \tag{11}$$

and

$$if\ S(0) < 0 \rightarrow 0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta} \tag{12}$$

Equation (12) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0 \tag{13}$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt} + \lambda)\quad \tilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \tag{14}$$

The derivation of S, namely, $\dot{S}$ can be calculated as the following;

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \tag{15}$$

suppose the second order system is defined as;

$$\ddot{x} = f + u \ \rightarrow \dot{S} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \tag{16}$$

Where f is the dynamic uncertain, and also since $S = 0\ and\ \dot{S} = 0$, to have the best approximation ,$\widehat{U}$ is defined as

$$\widehat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \tag{17}$$

A simple solution to get the sliding condition when the dynamic parameters have uncertainty is the switching control law:

$$U_{dis} = \widehat{U} - K(\vec{x},t)\cdot \mathrm{sgn}(s) \tag{18}$$

where the switching function $\mathrm{sgn}(S)$ is defined as [1, 6]

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{19}$$

and the $K(\vec{x},t)$ is the positive constant. Suppose by (8) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S}\cdot S = \left[f - \hat{f} - K\mathrm{sgn}(s)\right]\cdot S = \left(f - \hat{f}\right)\cdot S - K|S| \tag{20}$$

and if the equation (12) instead of (11) the sliding surface can be calculated as

$$s(x,t) = (\frac{d}{dt} + \lambda)^2 \left(\int_0^t \tilde{x}\, dt\right) = (\dot{x} - \dot{x}_d) + 2\lambda(\dot{x} - \dot{x}_d) - \lambda^2(x - x_d) \tag{21}$$

in this method the approximation of U is computed as [6]

$$\widehat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{x} - \dot{x}_d) + \lambda^2(x - x_d) \tag{22}$$

Based on above discussion, the sliding mode control law for a multi degrees of freedom robot manipulator is written as [1, 6]:

$$\tau = \tau_{eq} + \tau_{dis} \tag{23}$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems and τ_{eq} for first 3 DOF PUMA robot manipulator can be calculate as follows [1]:

$$\tau_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{24}$$

and τ_{dis} is computed as [1];

$$\tau_{dis} = K \cdot \mathrm{sgn}(S) \tag{25}$$

by replace the formulation (25) in (23) the control output can be written as;

$$\tau = \tau_{eq} + K.\mathrm{sgn}(S) \tag{26}$$

By (26) and (24) the sliding mode control of PUMA 560 robot manipulator is calculated as;

$$\tau = \left[M^{-1}(B + C + G) + \dot{S}\right]M + K \cdot \mathrm{sgn}(S) \tag{27}$$

where $S = \lambda e + \dot{e}$ in PD-SMC and $S = \lambda e + \dot{e} + (\frac{\lambda}{2})^2 \sum e$ in PID-SMC.

Proof of Stability: the lyapunov formulation can be written as follows,

$$V = \frac{1}{2}S^T.M.S \tag{28}$$

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2}S^T.\dot{M}.S + S^T M\dot{S} \tag{29}$$

the dynamic equation of IC engine can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + B + C + G \tag{30}$$

it is assumed that

$$S^T\left(\dot{M} - 2B + C + G\right)S = 0 \tag{31}$$

by substituting (30) in (29)

$$\dot{V} = \frac{1}{2}S^T\dot{M}S - S^TB + CS + S^T\left(M\dot{S} + B + CS + G\right) = S^T\left(M\dot{S} + B + CS + G\right) \tag{32}$$

suppose the control input is written as follows

$$\widehat{U} = \widehat{U_{Nonlinear}} + \widehat{U_{dis}} = \left[\widehat{M^{-1}}(B + C + G) + \dot{S}\right]\widehat{M} + K.sgn(S) + B + CS + G \tag{33}$$

by replacing the equation (33) in (32)

$$\dot{V} = S^T(M\dot{S} + B + C + G - \widehat{M}\dot{S} - \widehat{B + C}S + G - Ksgn(S) = S^T\left(\widetilde{M}\dot{S} + \widetilde{B + C}S + G - Ksgn(S)\right) \tag{34}$$

it is obvious that

$$\left|\tilde{M}\dot{S} + \widetilde{B + CS} + G\right| \le \left|\tilde{M}\dot{S}\right| + \left|\widetilde{B + CS} + G\right| \tag{35}$$

the Lemma equation in robot arm system can be written as follows

$$K_u = \left[\left|\tilde{M}\dot{S}\right| + |B + CS + G| + \eta\right]_i , i = 1, 2, 3, 4, \dots \tag{36}$$

the equation (11) can be written as

$$K_u \ge \left|\left[\tilde{M}\dot{S} + B + CS + G\right]_i\right| + \eta_i \tag{37}$$

therefore, it can be shown that

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i |S_i| \tag{38}$$

Consequently the equation (38) guaranties the stability of the Lyapunov equation. Figure 1 is shown pure sliding mode controller, applied to robot arm.

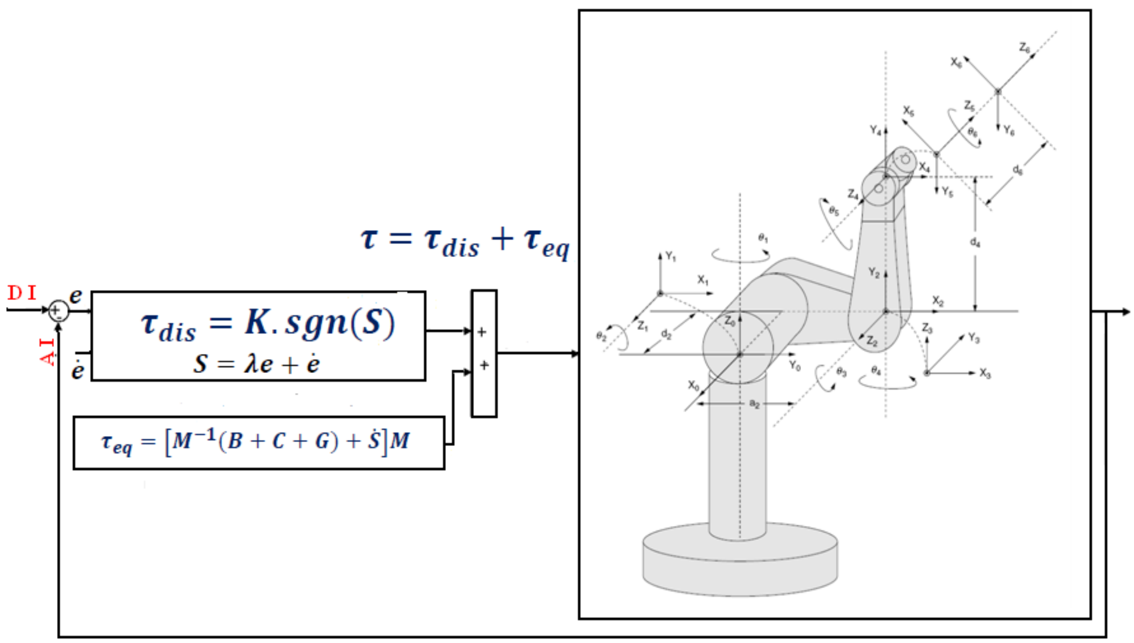


FIGURE 1: Block diagram of a sliding mode controller: applied to robot arm

2. METHODOLOGY: DESIGN MATHEMATICAL ERROR-BASED CHATTERING FREE SLIDING MODE CONTROLLER WITH SWITCHING FUNCTION

Sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical error-based tuning method which this method can helps to eliminate the chattering in presence of switching function method and improves the system's tracking performance by online tuning method. In this research the nonlinear equivalent dynamic (equivalent part) formulation problem in uncertain system is solved by using on-line mathematical error-based tuning theorem. In this method mathematical error-based theorem is applied to sliding mode controller to estimate the nonlinear equivalent part. Sliding mode controller has difficulty in handling unstructured model uncertainties and this controller's performance is sensitive to sliding surface slope coefficient. It is possible to solve above challenge by combining mathematical error-based tuning method and sliding mode controller which this methodology can help to improve system's tracking performance by on-line tuning (mathematical error-based tuning) method. Based on above discussion, compute the best value of sliding surface slope coefficient has played important role to improve system's tracking performance especially when the system parameters are unknown or uncertain. This problem is solved by tuning the surface slope coefficient ($\boldsymbol{\lambda}$) of the sliding mode controller continuously in real-time. In this methodology, the system's performance is improved with respect to the classical sliding mode controller. Figure 2 shows the mathematical error-based tuning sliding mode controller. Based on (23) and (27) to adjust the sliding surface slope coefficient we define $\hat{f}(x|\lambda)$ as the fuzzy based tuning.

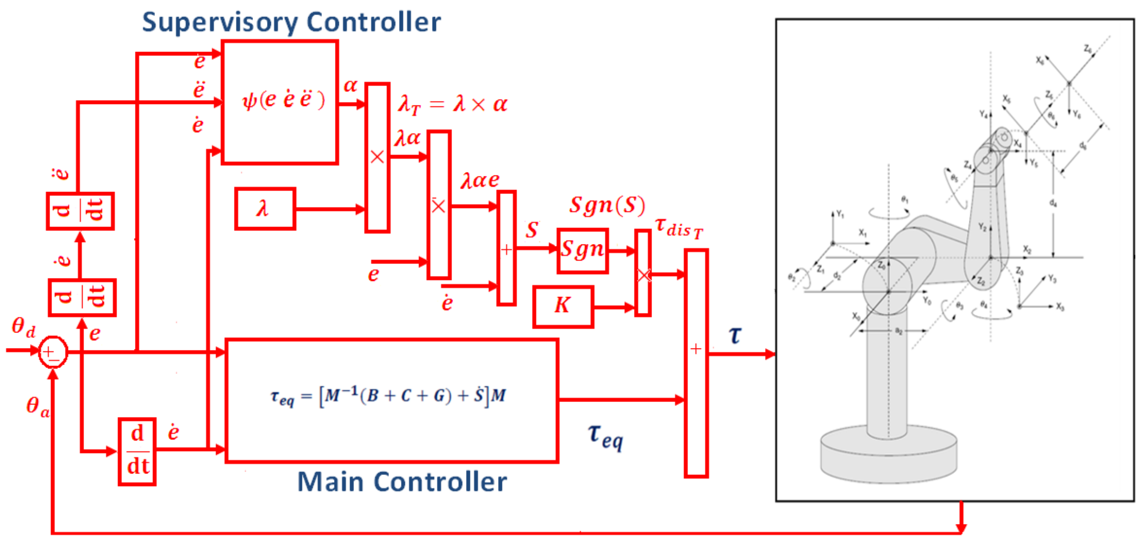


FIGURE 2: Block diagram of a mathematical error-based sliding mode controller: applied to robot arm

$$\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) = \boldsymbol{\lambda}^{\boldsymbol{T}}\boldsymbol{\alpha} \quad \textbf{(39)}$$

If minimum error ($\boldsymbol{\lambda}^*$) is defined by;

$$\boldsymbol{\lambda}^* = \boldsymbol{arg\ min}\ [(\boldsymbol{Sup}|\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{x})|)] \quad \textbf{(40)}$$

Where λ^T is adjusted by an adaption law and this law is designed to minimize the error's parameters of $\boldsymbol{\lambda} - \boldsymbol{\lambda}^*$. adaption law in mathematical error-based tuning sliding mode controller is used to adjust the sliding surface slope coefficient. Mathematical error-based tuning part is a supervisory controller based on the following formulation methodology. This controller has three inputs namely; error (e), change of error ($\dot{e}$) and the second derivative of error ($\ddot{e}$) and an output namely; gain updating factor(α). As a summary design a mathematical error-based tuning is based on the following formulation:

$$\boldsymbol{\alpha} = \boldsymbol{e}^2 - \frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C\right)^5}{1+|e|} + \mathbf{C} \quad \textbf{(41)}$$

$$\dot{\boldsymbol{e}}(*) = \dot{\boldsymbol{e}}(\boldsymbol{t}) \quad \boldsymbol{if}\ \dot{\boldsymbol{e}}(\boldsymbol{t}) \geq \dot{\boldsymbol{e}}(\boldsymbol{t}-\mathbf{1})$$
$$\dot{\boldsymbol{e}}(*) = \dot{\boldsymbol{e}}(\boldsymbol{t}-\mathbf{1}) \quad \boldsymbol{if}\ \dot{\boldsymbol{e}}(\boldsymbol{t}-\mathbf{1}) > \dot{\boldsymbol{e}}(\boldsymbol{t})$$

Where (α) is gain updating factor, ($\ddot{e}$) is the second derivative of error, ($\dot{e}$) is change of error, (e) is error and C is a coefficient.

Proof of Stability: The Lyapunov function in this design is defined as

$$V = \frac{1}{2} S^T M S + \frac{1}{2} \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \phi_j \tag{42}$$

where γ_{sj} is a positive coefficient, $\phi = \lambda^* - \lambda$, θ^* is minimum error and λ is adjustable parameter. Since $\dot{M} - 2V$ is skew-symetric matrix;

$$S^T M \dot{S} + \frac{1}{2} S^T \dot{M} S = S^T (M\dot{S} + VS) \tag{43}$$

If the dynamic formulation of robot manipulator defined by

$$\tau = M(q)\ddot{q} + V(q, \dot{q})\dot{q} + G(q) \tag{44}$$

the controller formulation is defined by

$$\tau = \widehat{M}\ddot{q}_r + \widehat{V}\dot{q}_r + \widehat{G} - \lambda S - K \tag{45}$$

According to (43) and (44)

$$M(q)\ddot{q} + V(q, \dot{q})\dot{q} + G(q) = \widehat{M}\ddot{q}_r + \widehat{V}\dot{q}_r + \widehat{G} - \lambda S - K \tag{46}$$

Since $\dot{q}_r = \dot{q} - S$ and $\ddot{q}_r = \ddot{q} - \dot{S}$

$$M\dot{S} + (V + \lambda)S = \Delta f - K \tag{47}$$

$$M\dot{S} = \Delta f - K - VS - \lambda S$$

The derivation of V is defined

$$\dot{V} = S^T M \dot{S} + \frac{1}{2} S^T \dot{M} S + \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \dot{\phi}_j \tag{48}$$

$$\dot{V} = S^T (M\dot{S} + VS) + \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \dot{\phi}_j$$

Based on (46) and (47)

$$\dot{V} = S^T (\Delta f - K - VS - \lambda S + VS) + \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \dot{\phi}_j \tag{49}$$

where $\Delta f = [M(q)\ddot{q} + V(q, \dot{q})\dot{q} + G(q)] - \sum_{l=1}^{M} \lambda^T \alpha$

$$\dot{V} = \sum_{J=1}^{M} [S_j (\Delta f_j - K_j)] - S^T \lambda S + \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \dot{\phi}_j$$

suppose α is defined as follows

$$\alpha_j = e^2 - \frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C\right)^5}{1 + |e|} + \mathrm{C} \tag{50}$$

according to 48 and 49;

$$\dot{V} = \sum_{J=1}^{M} \left[S_j (\Delta f_j - \lambda^T [e^2 - \frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C\right)^5}{1 + |e|} + \mathrm{C}] \right] - S^T \lambda S + \sum_{J=1}^{M} \frac{1}{\gamma_{sj}} \phi^T . \dot{\phi}_j \tag{51}$$

Based on $\phi = \theta^* - \theta \rightarrow \theta = \theta^* - \phi$

$$\dot{V}=\sum_{J=1}^{M}\left[S_j(\Delta f_j-\theta^{*T}[e^2-\frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C\right)^5}{1+|e|}+\mathrm{C}]+\phi^T[e^2-\frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C\right)^5}{1+|e|}+\mathrm{C}]\right]-S^T\lambda S+\sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j \tag{52}$$

$$\dot{V}=\sum_{J=1}^{M}\left[S_j(\Delta f_j-(\lambda^*)^T[e^2-\frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C\right)^5}{1+|e|}+\mathrm{C}]\right]-S^T\lambda S+\sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi_j^T[e^2-\frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C)^5}{1+|e|}+\mathrm{C}+\dot{\phi}_j])$$

where $\dot{\theta}_j=e^2-\frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C)^5}{1+|e|}+\mathrm{C}$ is adaption law, $\dot{\emptyset}_j=-\dot{\theta}_j=-[e^2-\frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C)^5}{1+|e|}+\mathrm{C}]$

$\dot{V}$ is considered by

$$\dot{V}=\sum_{j=1}^{m}[S_j\,\Delta f_j-\left((\lambda_j^*)^T e^2-\frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C)^5}{1+|e|}+\mathrm{C}\right)]-S^T\lambda S \tag{53}$$

The minimum error is defined by

$$e_{mj}=\Delta f_j-\left((\lambda_j^*)^T e^2-\frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right)-C)^5}{1+|e|}+\mathrm{C}\right) \tag{54}$$

Therefore $\dot{V}$ is computed as

$$\dot{V}=\sum_{j=1}^{m}[S_j\,e_{mj}]-S^T\lambda S \tag{55}$$

$$\le \textstyle\sum_{j=1}^{m}|S_j\,||e_{mj}|-S^T\lambda S$$

$$=\sum_{j=1}^{m}|S_j\,||e_{mj}|-\lambda_j {S_j}^2$$

$$=\sum_{j=1}^{m}|S_j\,|(|e_{mj}|-\lambda_j S_j) \tag{56}$$

For continuous function $g(x)$, and suppose $\varepsilon>0$ it is defined the fuzzy logic system in form of

$$Sup_{x\in U}|f(x)-g(x)|<\epsilon \tag{57}$$

the minimum approximation error (e_{mj}) is very small.

$$if\ \ \lambda_j=\alpha\quad that\quad \alpha|S_j|>e_{mj}\,(S_j\neq 0)\qquad then\quad \dot{V}<0\,for\,(S_j\neq 0) \tag{58}$$

3. RESULTS

Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical error-based tuning in a single IC chip or combining sliding mode controller by fuzzy logic method (FSMC). These methods can improve the system's tracking performance by online tuning method or soft computing method. Proposed method is based on resolve the on line sliding surface slope as well as improve the output performance by tuning the sliding surface slope coefficient. The sliding surface gain ($\boldsymbol{\lambda}$) of this controller is adjusted online depending on the last values of error (e), change of error ($\dot{e}$) and power two of derivative of error ($\ddot{e}$) by sliding surface slope updating factor (α). Fuzzy sliding mode controller is based on applied fuzzy logic in sliding mode controller to estimate the dynamic formulation in equivalent part. Mathematical error-based tuning sliding mode controller is stable model-based controller which does not need to limits the dynamic model of robot manipulator and eliminate the chattering phenomenon without to use the boundary layer saturation function.

This section is focused on compare between Sliding Mode Controller (SMC), Fuzzy Sliding Mode Controller (FSMC) and mathematical error-based tuning Sliding Mode Controller (MTSMC). These controllers were tested by step responses. In this simulation, to control position of PUMA robot manipulator the first, second, and third joints are moved from home to final position without and with external disturbance. The simulation was implemented in Matlab/Simulink environment. **trajectory performance, torque performance, disturbance rejection, steady state error and RMS error** are compared in these controllers. These systems are tested by band limited white noise with a predefined 10%, 20% and 40% of relative to the input signal amplitude. This type of noise is used to external disturbance in continuous and hybrid systems and applied to nonlinear dynamic of these controllers.

Tracking Performances
Based on (27) in sliding mode controller; controllers performance are depended on the gain updating factor (K) and sliding surface slope coefficient (λ). These two coefficients are computed by trial and error in SMC. The best possible coefficients in step FSMC are; $K_p = K_v = K_i = 18$, $\emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 3, \lambda_2 = 6, \lambda_3 = 6$ and the best possible coefficients in step SMC are; $\lambda_1 = 1\ , \lambda_2 = 6, \lambda_3 = 8$; $K_p = K_v = K_i = 10$; $\emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1$. In mathematical error-based tuning sliding mode controller the sliding surface gain is adjusted online depending on the last values of error (e), change of error ($\dot{e}$) and the second derivation of error ($\ddot{e}$) by sliding surface slope updating factor (α). Figure 3 shows tracking performance in mathematical error-based tuning sliding mode controller (MTSMC), fuzzy sliding mode controller (FSMC) and SMC without disturbance for step trajectory.

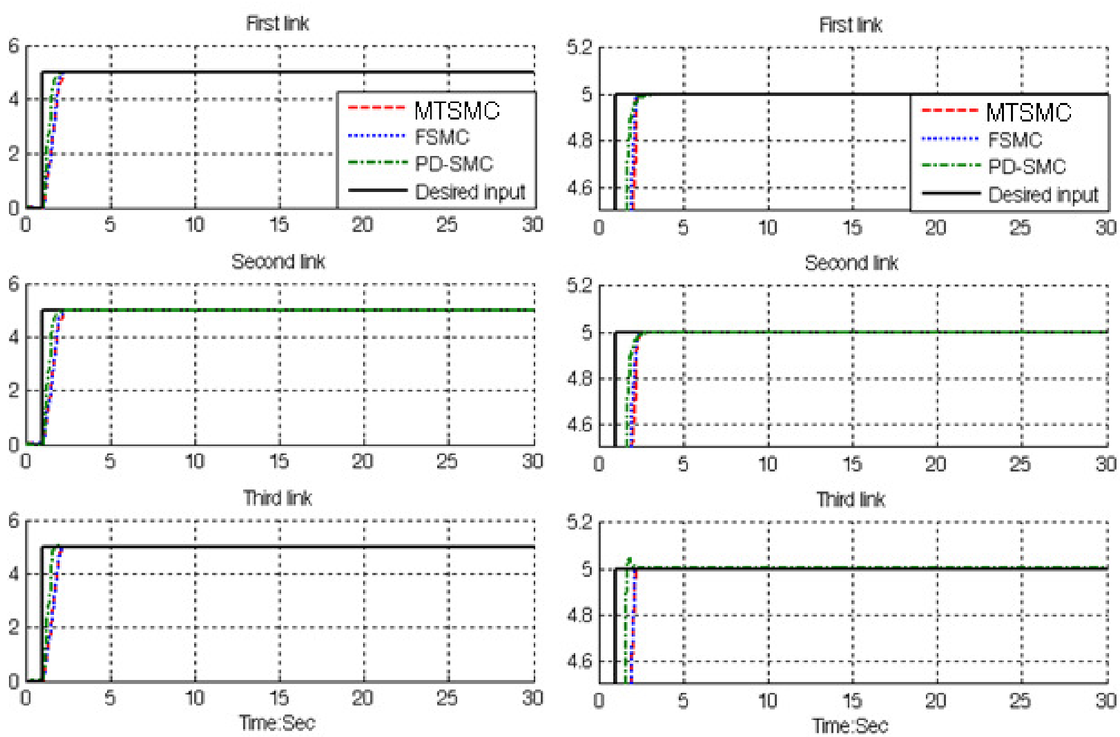


FIGURE 3 FSMC, MTSMC, desired input and SMC for first, second and third link step trajectory performance without disturbance

Based on Figure 3 it is observed that, the overshoot in MTSMC is 0%, in SMC's is 1% and in FSMC's is 0%, and rise time in MTSMC's is 0.6 seconds, in SMC's is 0.483 second and in FSMC's is about 0.6 seconds. From the trajectory MATLAB simulation for MTSMC, SMC and FSMC in certain system, it was seen that all of three controllers have acceptable performance.

Disturbance Rejection

Figures 4 to 6 show the power disturbance elimination in MTSMC, SMC and FSMC with disturbance for step trajectory. The disturbance rejection is used to test the robustness comparisons of these three controllers for step trajectory. A band limited white noise with predefined of 10%, 20% and 40% the power of input signal value is applied to the step trajectory. It found fairly fluctuations in SMC and FSMC trajectory responses.

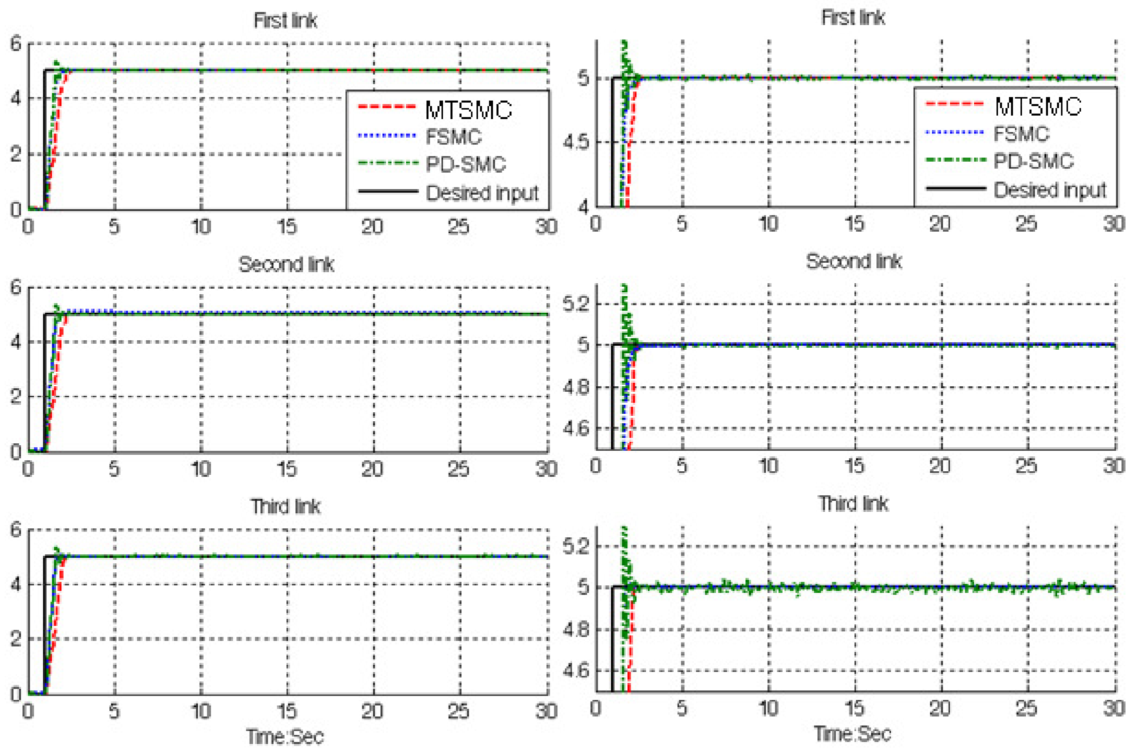


FIGURE 4 : Desired input, MTSMC, FSMC and SMC for first, second and third link trajectory with 10%external disturbance: step trajectory

Based on Figure 4; by comparing step response trajectory with 10% disturbance of relative to the input signal amplitude in MTSMC, FSMC and SMC, MTSMC's overshoot about (0%) is lower than FSMC's **(0.5%)** and SMC's (**1%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.63 second)** and MTSMC's **(0.65 second).** Besides the Steady State and RMS error in MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =1.08e-12 and RMS error=1.5e-12)** are bout lower than FSMC **(Steady State error =1.08e-6 and RMS error=1.5e-6)** and SMC's **(Steady State error=1.6e-6 and RMS error=1.9e-6)**.

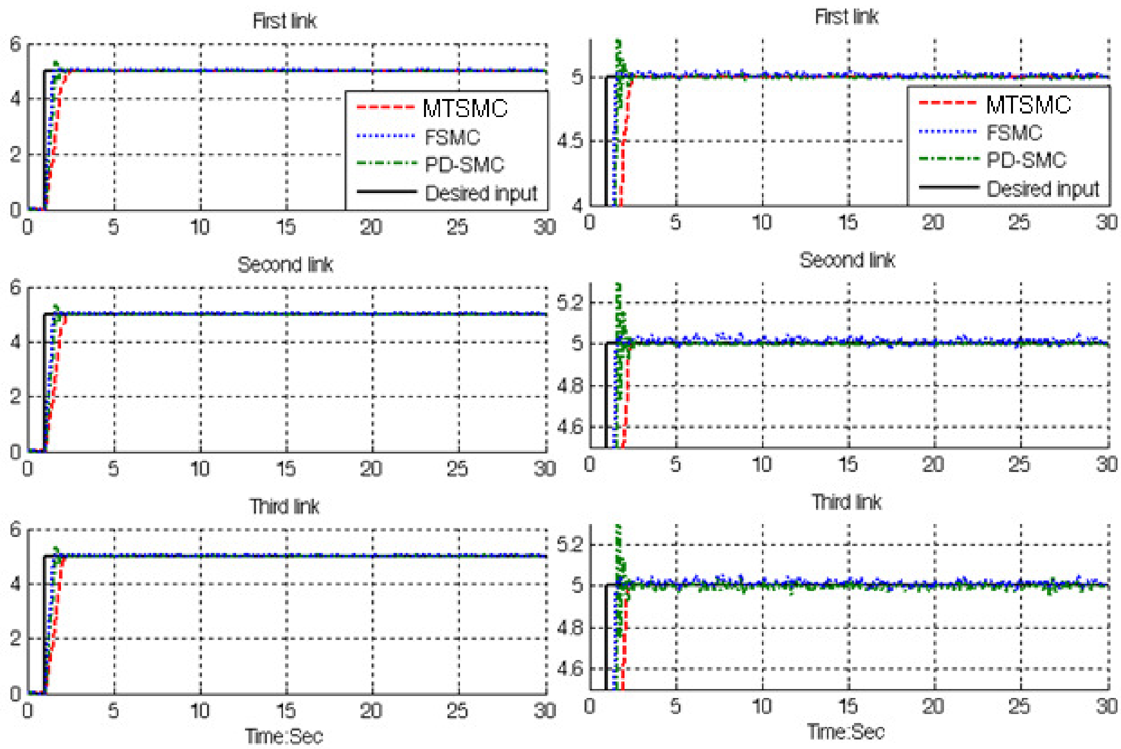


FIGURE 5: Desired input, MTSMC, FSMC and SMC for first, second and third link trajectory with 20%external disturbance: step trajectory

Based on Figure 5; by comparing step response trajectory with 20% disturbance of relative to the input signal amplitude in MTSMC, FSMC and SMC, MTSMC's overshoot about **(0%)** is lower than FSMC's **(1.8%)** and SMC's (**2.1%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.63 second)** and MTSMC's **(0.66 second).** Besides the Steady State and RMS error in FTFSMC, FSMC and PD-SMC it is observed that, error performances in MTSMC **(Steady State error =1.2e-12 and RMS error=1.8e-12)** are about lower than FSMC **(Steady State error =1.7e-5 and RMS error=2e-5)** and SMC's **(Steady State error=1.8e-5 and RMS error=2e-5)**. Based on Figure 6, it was seen that, MTSMC's performance is better than FSMC and SMC because MTSMC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas FSMC and SMC cannot.

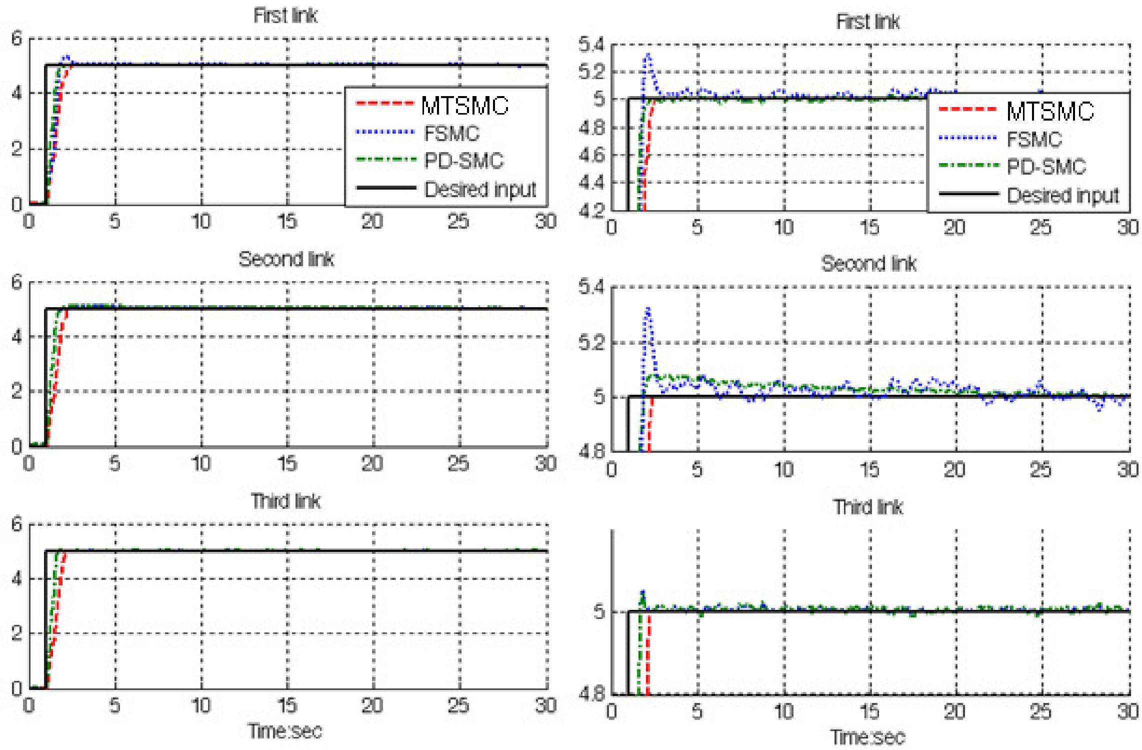


FIGURE 6 : Desired input, MTSMC, FSMC and SMC for first, second and third link trajectory with 40%external disturbance: step trajectory

Based on Figure 6; by comparing step response trajectory with 40% disturbance of relative to the input signal amplitude in MTSMC, SMC and FSMC, MTSMC's overshoot about **(0%)** is lower than FSMC's **(6%)** and PD-SMC's (**8%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.7 second)** and MTSMC's **(0.8 second).** Besides the Steady State and RMS error in MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =1.3e-12 and RMS error=1.8e-12)** are about lower than FSMC **(Steady State error =10e-4 and RMS error=0.69e-4)** and SMC's **(Steady State error=10e-4 and RMS error=11e-4).** Based on Figure 7, FSMC and SMC have moderately oscillation in trajectory response with regard to 40% of the input signal amplitude disturbance but MTSMC has stability in trajectory responses in presence of uncertainty and external disturbance. Based on Figure 6 in presence of 40% unstructured disturbance, MTSMC's is more robust than FSMC and SMC because MTSMC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas FSMC and SMC cannot.

Torque Performance

Figures 7 and 8 have indicated the power of chattering rejection in MTSMC, SMC and FSMC with 40% disturbance and without disturbance.

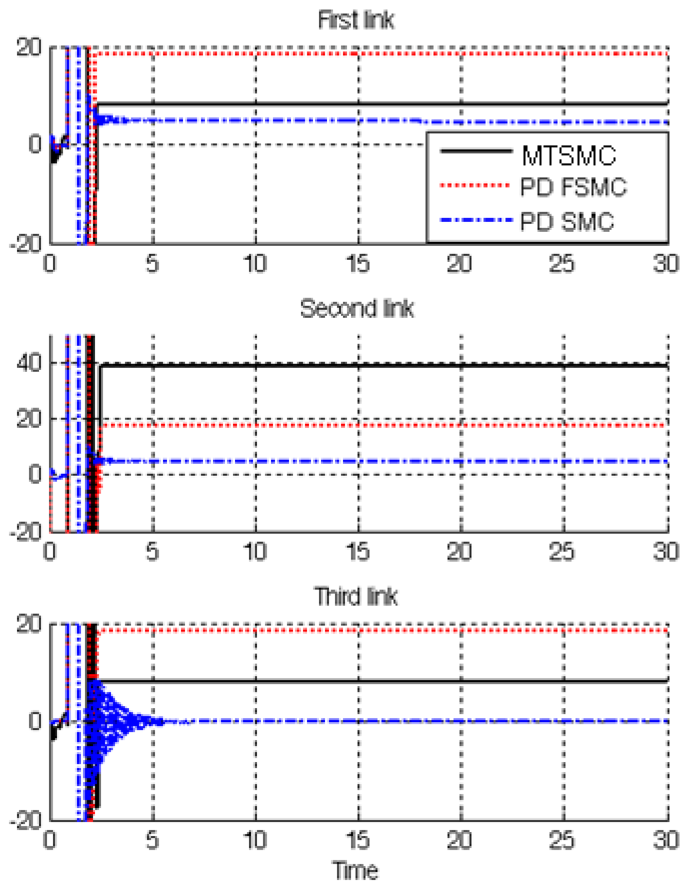


FIGURE 7 : MTSMC, SMC and FSMC for first, second and third link torque performance without disturbance

Figure 7 shows torque performance for first three links PUMA robot manipulator in MTSMC, SMC and FSMC without disturbance. Based on Figure 7, MTSMC, SMC and FSMC give considerable torque performance in certain system and all three of controllers eliminate the chattering phenomenon in certain system. Figure 8 has indicated the robustness in torque performance for first three links PUMA robot manipulator in MTSMC, SMC and FSMC in presence of 40% disturbance. Based on Figure 8, it is observed that SMC and FSMC controllers have oscillation but MTSMC has steady in torque performance. This is mainly because pure SMC with saturation function and fuzzy sliding mode controller with saturation function are robust but they have limitation in presence of external disturbance. The MTSMC gives significant chattering elimination when compared to FSMC and SMC. This elimination of chattering phenomenon is very significant in presence of 40% disturbance. This challenge is one of the most important objectives in this research.

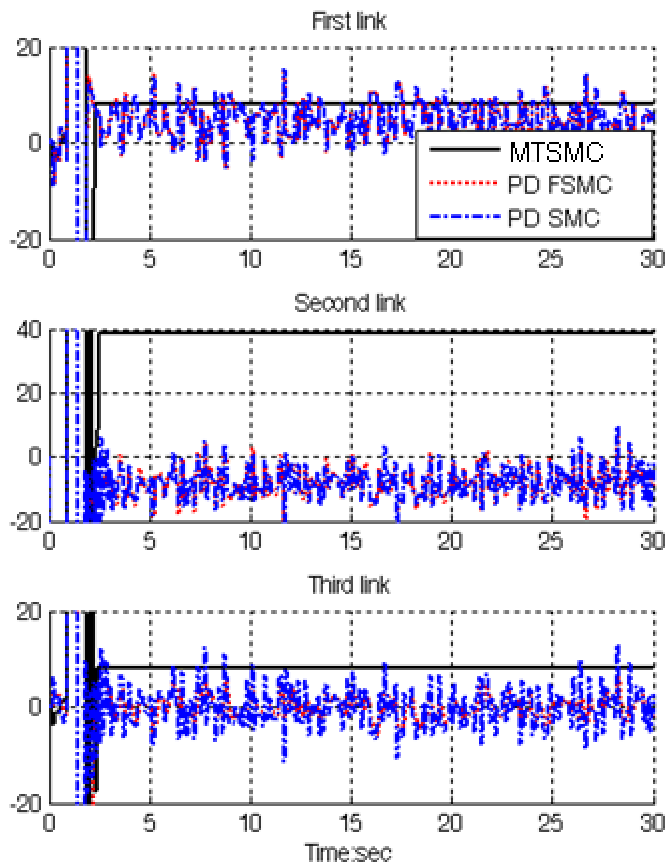


FIGURE 8: MTSMC, SMC and FSMC for first, second and third link torque performance with40% disturbance

Based on Figure 8 it is observed that, however mathematical tuning error-based sliding mode controller (MTSMC) is a model-based controller that estimate the nonlinear dynamic equivalent formulation by system's performance but it has significant torque performance (chattering phenomenon) in presence of uncertainty and external disturbance. SMC and FSMC have limitation to eliminate the chattering in presence of highly external disturbance (e.g., 40% disturbance) but MTSMC is a robust against to highly external disturbance.

Steady State Error

Figure 9 is shown the error performance in MTSMC, SMC and FSMC for first three links of PUMA robot manipulator**.** The error performance is used to test the disturbance effect comparisons of these controllers for step trajectory. All three joint's inputs are step function with the same step time (step time= 1 second), the same initial value (initial value=0) and the same final value (final value=5). Based on Figure 4, MTSMC's rise time is about 0.6 second, SMC's rise time is about 0.483 second and FSMC's rise time is about 0.6 second which caused to create a needle wave in the range of 5 (amplitude=5) and the different width. In this system this time is transient time and this part of error introduced as a transient error. Besides the Steady State and RMS error in MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =0.9e-12 and RMS error=1.1e-12)** are about lower than FSMC **(Steady State error =0.7e-8 and RMS error=1e-7)** and SMC's **(Steady State error=1e-8 and RMS error=1.2e-6)**.

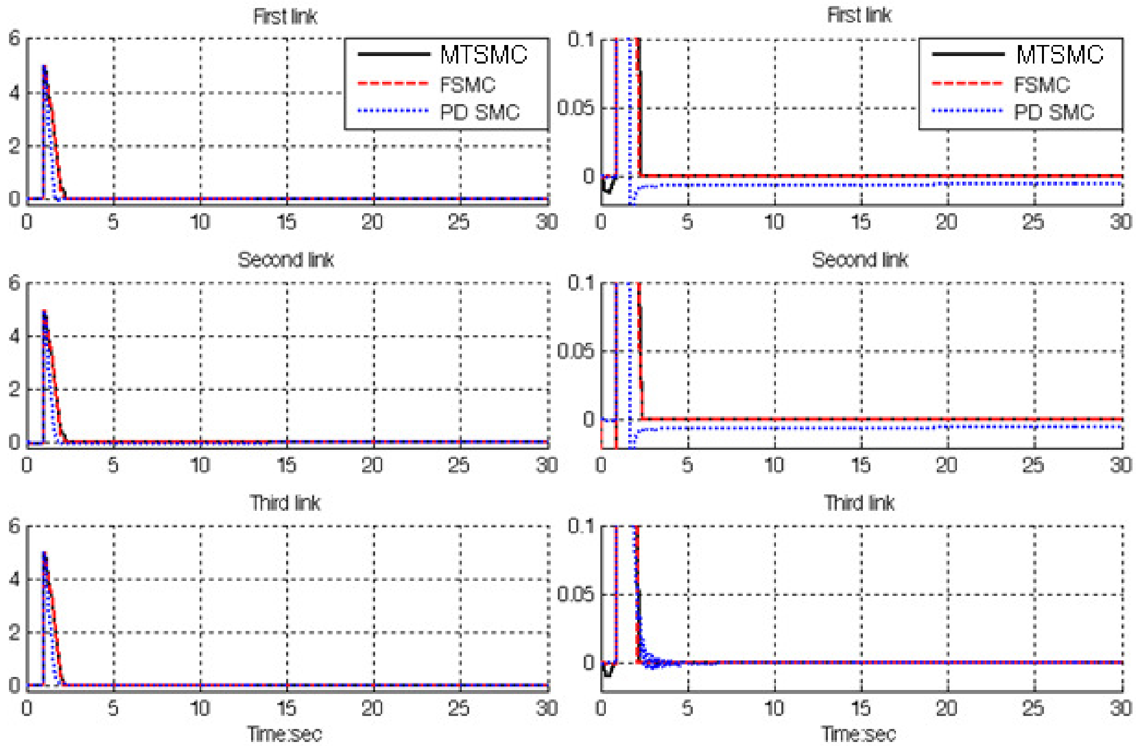


FIGURE 9 : MTSMC, SMC and FSMC for first, second and third link steady state error without disturbance: step trajectory

The MTSMC gives significant steady state error performance when compared to FSMC and SMC. When applied 40% disturbances in MTSMC the RMS error increased approximately 0.0164% (percent of increase the MTSMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{1.8e{-}12}{1.1e{-}12} = 0.0164\%$), in FSMC the RMS error increased approximately 6.9% (percent of increase the FSMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{0.69e{-}4}{1e{-}7} = 6.9\%$)in SMC the RMS error increased approximately 9.17% (percent of increase the PD-SMC RMS error= $\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbance\ RMS\ error} = \frac{11e{-}4}{1.2e{-}6} = 9.17\%$). In this part MTSMC, SMC and FSMC have been comparatively evaluation through MATLAB simulation, for PUMA robot manipulator control. It is observed that however MTSMC is independent of nonlinear dynamic equation of PUMA 560 robot manipulator but it can guarantee the trajectory following and eliminate the chattering phenomenon in certain systems, structure uncertain systems and unstructured model uncertainties by online tuning method.

4. CONCLUSION

Refer to this research, a mathematical error-based tuning sliding mode controller (MTSMC) is proposed for PUMA robot manipulator. Pure sliding mode controller with saturation function and fuzzy sliding mode controller with saturation function have difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining fuzzy sliding mode controller and mathematical error-based tuning. Since the sliding surface gain (λ) is adjusted by mathematical error-based tuning method, it is nonlinear and continuous. The sliding surface slope updating factor (α) of mathematical error-based tuning part can be changed with the changes in error, change of error and the second change of error. Sliding surface gain is adapted on-line by sliding surface slope updating factor. In pure sliding mode controller and fuzzy sliding mode controller the sliding surface gain is chosen by trial and error, which means pure sliding mode controller and error-based fuzzy sliding mode controller have to have a prior knowledge of the system uncertainty. If the knowledge is not available error performance and chattering phenomenon are go up. In mathematical error-based tuning sliding mode controller the sliding surface gain are updated on-line to compensate the system unstructured uncertainty. The stability and convergence of the mathematical error-based tuning sliding mode controller based on switching function is guarantee and proved by the Lyapunov method. The simulation results exhibit that the mathematical error-based tuning sliding mode controller works well in various situations. Based on theoretical and simulation results, it is observed that mathematical error-based tuning sliding mode controller is a model-based stable control for robot manipulator. It is a best solution to eliminate chattering phenomenon with saturation function in structure and unstructured uncertainties.

REFERENCES

[1] T. R. Kurfess, *Robotics and automation handbook*: CRC, 2005.

[2] J. J. E. Slotine and W. Li, *Applied nonlinear control* vol. 461: Prentice hall Englewood Cliffs, NJ, 1991.

[3] K. Ogata, *Modern control engineering*: Prentice Hall, 2009.

[4] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, *Linear control system analysis and design with MATLAB*: CRC, 2003.

[6] B. Siciliano and O. Khatib, *Springer handbook of robotics*: Springer-Verlag New York Inc, 2008.

[7] I. Boiko, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* No. 3, vol. 2, pp. 171-179, 1986.

[10] H. Zhang and R. P. Paul, "A parallel solution to robot inverse kinematics," *IEEE conference proceeding,* 2002, pp. 1140-1145.

[11] J. Kieffer, "A path following algorithm for manipulator inverse kinematics," *IEEE conference proceeding,* 2002, pp. 475-480.

[12] Z. Ahmad and A. Guez, "On the solution to the inverse kinematic problem(of robot)," *IEEE conference proceeding,* 1990, pp. 1692-1697.

[13] 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, No. 2,* vol. 27, pp. 332-343, 2002.

[14] M. W. Spong and M. Vidyasagar, *Robot dynamics and control*: Wiley-India, 2009.

[15] A. Vivas and V. Mosquera, "Predictive functional control of a PUMA robot," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," *IEEE conference proceeding,* 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE conference proceeding,* 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* No. 1, vol. 48, pp. 2-3, 2001.

[21] J. J. Slotine and S. Sastry, "Tracking control of non-linear systems using sliding surfaces, with application to robot manipulators†," *International Journal of Control,* No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* No. 2*,* vol. 40, pp. 421-434, 1984.

[23] R. Palm, "Sliding mode fuzzy control," *IEEE conference proceeding,*2002, pp. 519-526.

[24] C. C. Weng and W. S. Yu, "Adaptive fuzzy sliding mode control for linear time-varying uncertain systems," *IEEE conference proceeding,* 2008, pp. 1483-1490.

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics Journal,* No. 1, vol. 10, pp. 239-263, 2000.

[26] P. Kachroo and M. Tomizuka, "Chattering reduction and error convergence in the sliding-mode control of a class of nonlinear systems," *Automatic Control, IEEE Transactions on,* No. 7, vol. 41, pp. 1063-1068, 2002.

[27] H. Elmali and N. Olgac, "Implementation of sliding mode control with perturbation estimation (SMCPE)," *Control Systems Technology, IEEE Transactions on,* No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," *IEEE conference proceeding,* 2002, pp. 996-1000.

[29] Y. Li and Q. Xu, "Adaptive Sliding Mode Control With Perturbation Estimation and PID Sliding Surface for Motion Tracking of a Piezo-Driven Micromanipulator," *Control Systems Technology, IEEE Transactions on,* No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," *IEEE conference proceeding,* 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, “Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain,” International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] Farzin Piltan, A. R. Salehi and Nasri B Sulaiman.," Design artificial robust control of second order system based on adaptive fuzzy gain scheduling," world applied science journal (WASJ), 13 (5): 1085-1092, 2011

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, "Design Mathematical Tunable Gain PID-Like Sliding Mode Fuzzy Controller with Minimum Rule Base," International Journal of Robotic and Automation, 2 (3): 146-156, 2011.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli "Design Adaptive Fuzzy Robust Controllers for Robot Manipulator," World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , "Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line Tunable Gain: Applied to Robot Manipulator," International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, "Design On-Line Tunable Gain Artificial Nonlinear Controller," Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, " Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine," International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, "Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller," Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, "Novel Sliding Mode Controller for robot manipulator using FPGA," Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, "Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System," International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, "Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ," International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, "Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System," International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, "Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method," International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, “Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control,” International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, “Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ,” International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, “Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm,” International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, “Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ,” International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, “Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator,” International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator,” International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, “An Adaptive sliding surface slope adjustment in PD Sliding Mode Fuzzy Control for Robot Manipulator,” International Journal of Control and Automation , 4 (3): 65-76, 2011.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, “Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ,” International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, “Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, “On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory,” International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator,” International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, “Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine,” International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.," Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine," world applied science journal (WASJ), 15 (3): 422-428, 2011

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* No. 1, vol. 1, pp. 23-29, 2006.

[60] Y. Guo and P. Y. Woo, "An adaptive fuzzy sliding mode controller for robotic manipulators," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* No. 2, vol. 33, pp. 149-159, 2003.

[61] C. M. Lin and C. F. Hsu, "Adaptive fuzzy sliding-mode control for induction servomotor systems," *Energy Conversion, IEEE Transactions on,* No. 2, vol. 19, pp. 362-368, 2004.

[62] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," *Carleton university Ottawa,*2007.

[63] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," *IEEE Conference,* 2004, pp. 3588-3592.

[64] B. S. R. Armstrong, "Dynamics for robot control: friction modeling and ensuring excitation during parameter identification," 1988.

[65] C. L. Clover, "Control system design for robots used in simulating dynamic force and moment interaction in virtual reality applications," 1996.

[66] K. R. Horspool, *Cartesian-space Adaptive Control for Dual-arm Force Control Using Industrial Robots*: University of New Mexico, 2003.

[67] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," *IEEE Conference,* 2002, pp. 510-518.

[68] P. I. Corke and B. Armstrong-Helouvry, "A search for consensus among model parameters reported for the PUMA 560 robot," *IEEE Conference,* 2002, pp. 1608-1613.

[69] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[70] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* 2002, pp. 160-168.

INSTRUCTIONS TO CONTRIBUTORS

The *International Journal of Engineering (IJE)* is devoted in assimilating publications that document development and research results within the broad spectrum of subfields in the engineering sciences. The journal intends to disseminate knowledge in the various disciplines of the engineering field from theoretical, practical and analytical research to physical implications and theoretical or quantitative discussion intended for both academic and industrial progress.

Our intended audiences comprises of scientists, researchers, mathematicians, practicing engineers, among others working in Engineering and welcome them to exchange and share their expertise in their particular disciplines. We also encourage articles, interdisciplinary in nature. The realm of International Journal of Engineering (IJE) extends, but not limited, to the following:

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJE.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with volume 6, 2012, IJE appears in more focused issues. Besides normal publications, IJE intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

IJE LIST OF TOPICS

The realm of International Journal of Engineering (IJE) extends, but not limited, to the following:

- Aerospace Engineering
- Biomedical Engineering
- Civil & Structural Engineering
- Control Systems Engineering
- Electrical Engineering
- Engineering Mathematics
- Environmental Engineering
- Geotechnical Engineering
- Manufacturing Engineering
- Mechanical Engineering
- Nuclear Engineering
- Petroleum Engineering
- Telecommunications Engineering
- Agricultural Engineering
- Chemical Engineering
- Computer Engineering
- Education Engineering
- Electronic Engineering
- Engineering Science
- Fluid Engineering
- Industrial Engineering
- Materials & Technology Engineering
- Mineral & Mining Engineering
- Optical Engineering
- Robotics & Automation Engineering

CALL FOR PAPERS

Volume: 6 - **Issue:** 4 - August 2012

i. Paper Submission: May 31, 2012 **ii. Author Notification:** July 15, 2012

iii. Issue Publication: August 2012

CONTACT INFORMATION

Computer Science Journals Sdn BhD
B-5-8 Plaza Mont Kiara, Mont Kiara
50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6207 1607
006 03 2782 6991

Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
006 03 2782 6991

FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org