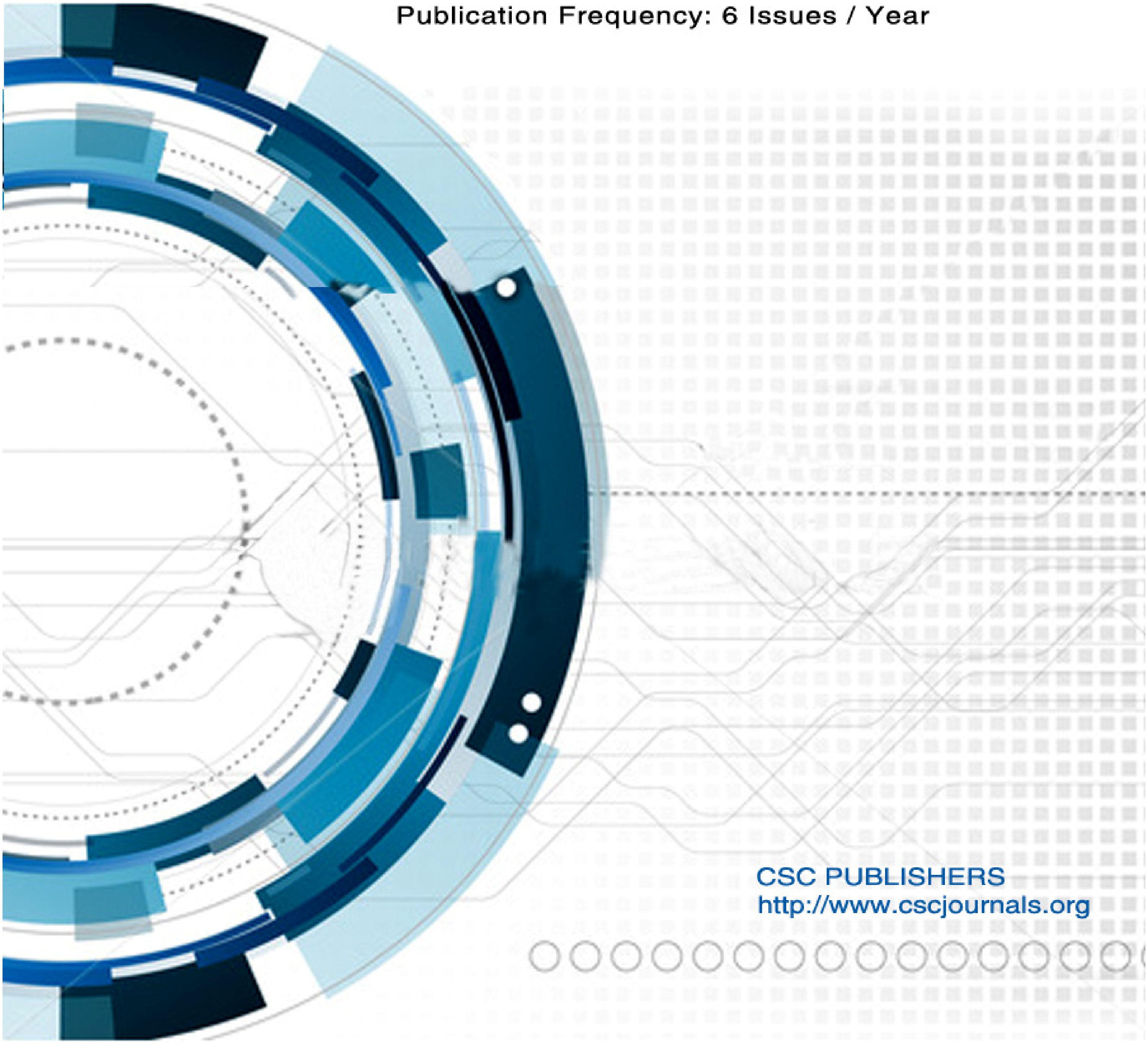
Editor-in-Chief
Dr. Kouroush Jenab

INTERNATIONAL JOURNAL OF ENGINEERING (IJE)

ISSN : 1985-2312

Volume 5 ▪ Issue 5 ▪ December 2011
Publication Frequency: 6 Issues / Year



CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ENGINEERING (IJE)

VOLUME 5, ISSUE 5 2011

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 5, Issue 5, December 2011
Publishing Date: 05-11-2011
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, 2011

EDITORIAL PREFACE

This is the fifth issue of volume five 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 5, 2011, 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)

EDITORIAL BOARD

Editor-in-Chief (EiC)

ASSOCIATE EDITORS (AEiCs)

EDITORIAL BOARD MEMBERS (EBMs)

TABLE OF CONTENTS

Volume 5, Issue 5, December 2011

Pages

Computer Aided Design of Couplings

Adeyeri Michael Kanisuru *sademike2003@yahoo.co.uk*
Engineering and Engineering Technology//Department of Mechanical Engineering
The Federal University of Technology Akure,
P.M.B. 704, Akure Ondo State, Nigeria

Adeyemi Michael Bolaji *mbadeyemi@yahoo.com*
Engineering and Engineering Technology//Department of Mechanical Engineering
University of Ilorin, Ilorin,
P.M.B. 1515, Ilorin, Kwara State, Nigeria

Ajayi Olumuyiwa Bamidele *ajayiob@yahoo.com*
Engineering and Engineering Technology//Department of Mechanical Engineering
Rufus Giwa Polytechnic,Owo Ondo State
P.M.B. 1019, Owo, Ondo State, Nigeria

Abadariki Samson Olaniran *soabadariki@yahoo.com*
Engineering and Engineering Technology//Department of Mechanical Engineering
The Federal University of Technology Akure,
P.M.B. 704, Akure Ondo State, Nigeria.

Abstract

The research work explores computer-aided approach to the design of ten different couplings, viz a viz: flange, solid rigid, hollow rigid, old ham/ cross-sliding, pin type flexible, sleeve, seller cone/ compression, split muff, pulley flange and fairbian’s lap-box couplings. The approach utilizes standard design equations of these couplings and link them together in computer software to determine the design parameters of the couplings. The work reviews the procedural steps involved in the design of couplings and the development of the software package using java as a tool for the design and dratfting of couplings. The design software named COUPLINGCAD combines with sketch template of a single process so as to generate the required parameters of the couplings. The COUPLINGCAD was tested with a number of case studies and the results obtained therein were quite satisfactory.

Keywords: Computer Aided design, Couplings, Couplingcad, Equations, Java

1: INTRODUCTION

A coupling is a device used to connect two shafts together at their ends for the purpose of transmitting power [1]. Couplings do not normally allow disconnection of shafts during operation, though there do exist torque limiting couplings which can slip or disconnect when some torque limit is exceeded [2].

The primary purpose of couplings is to join two pieces of rotating equipment while permitting some degree of misalignment or end movement or both [3]. Shaft couplings are used in machinery for protection against overloads and for power transmission. Most machines are integrated collection of power transmission elements that could be used for the business of moving energy or power from the place where it is generated to where it is to be used. Transmission element in machine tool could be mechanical, hydraulic, pneumatic or electric in nature. Shaft is one of the most important mechanical transmission element which needs to be coupled properly by the use of shaft couplings. A coupling is mainly to connect two shafts semi permanently.

Theoretically, the design and analysis of shaft couplings has been written extensively by several authors and the result put together in textbooks for use by the engineers.

SHAFTCAD software was developed through the research work on Computer-Aided design of power transmission shaft.The research establishedthe ease of designing power transmission shaft through various loadings [4].

A software package for the design of special transmission elements in which the various transmission elements such as brakes, power screws, chains and couplings were integrated into a unit package of which the package could only compute the design parameters without the detailed drawings of the machine components [5].

At present, there are no documented software developed for designing couplings except those one designed for spur gears, clutches, flywheel, rolling bearings, helical gears, power screws and chain drives as engineering transmission element [6],[7].

2 : EQUATIONS FOR DESIGN ANALYSIS

The various design equations needed for the design of these categories of couplings are as discussed below:

2.1 Design of Flange Coupling

For design purpose, it is to be noted that flange coupling transmit large torque. With reference to Figure 1 [8].
The following are the dimensional parameters to be considered when designing flange coupling; with lettering from figure 1

The appropriate number of bolts, i, is $i = 0.2d + 3$ (1)

Where d = shaft diameter

The average value of the diameter of the bolt circle, D_1 in cm

$$D_1 = 2d + 5 \quad (2)$$

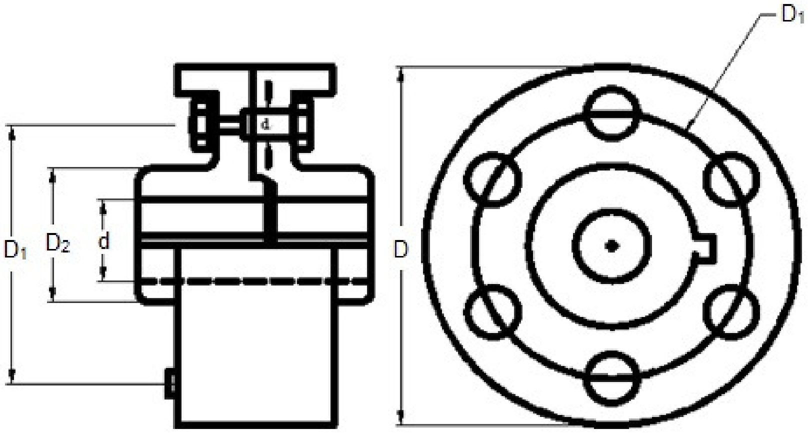


FIGURE 1: Flange coupling.

The hub diameter, D_2, is $D_2 = 1.5d + 2.5cm$ (3)

The outside diameter of flange, D, is $D = 2.5d + 7.5cm$ (4)

The hub length, L, is $L = 1.25d + 1.875$ (5)

The power *N* is in kilowatt, is $N = \frac{d^3 \Pi n \xi \tau_s}{1558400}$ N (6)

Where n = speed (r.p.m), τ_s = design shear stress in shaft and

ξ = factor which takes care of the reduced strength due to keyway.

And $\zeta = 1 + \frac{0.2x + 1.1y}{d}$

Where: x = width of keyway, cm and y = depth of keyway, cm

The torque transmitted by the coupling, M_{tc}, is $M_{tc} = \frac{97400}{N}$ (7)

Where N = Power in Kilowatt

Torque transmitted through bolts, M_{tb}, $M_{tb} = i\left(\frac{\Pi d_1^2}{4}\right)\tau_b \frac{D_1}{2}$ (8)

Torque capacity based on shear of flange, M_{tsf},

$$M_{tsf} = t(\pi D_2)\tau_f \frac{D_2}{2} \qquad (9)$$

where τ_f is the shear stress in flange at the outside hub diameter, kg.

The mean radius, r_m, $r_m = \left(\frac{D + d}{2}\right)cm$ (10)

The tension of load in each bolt, $F_b = \frac{M_{tc}}{i \mu r_m}$ (11)

The preliminary bolt diameter, d_1, is $d_1 = \frac{0.5d}{\sqrt{i}}$ (12)

The allowable or design stress in bolts, τ_b $\tau_b = \frac{779200N}{\pi d_1^2 i n D_1}$ (13)

Flange thickness, $t = 0.25d$ (14)

Design shear stress, $$\tau_f = \frac{2M_{tc}}{\pi D_1^2 t} \quad (15)$$

2.2 Design of Solid Rigid Coupling

Refer figure 2 [8]

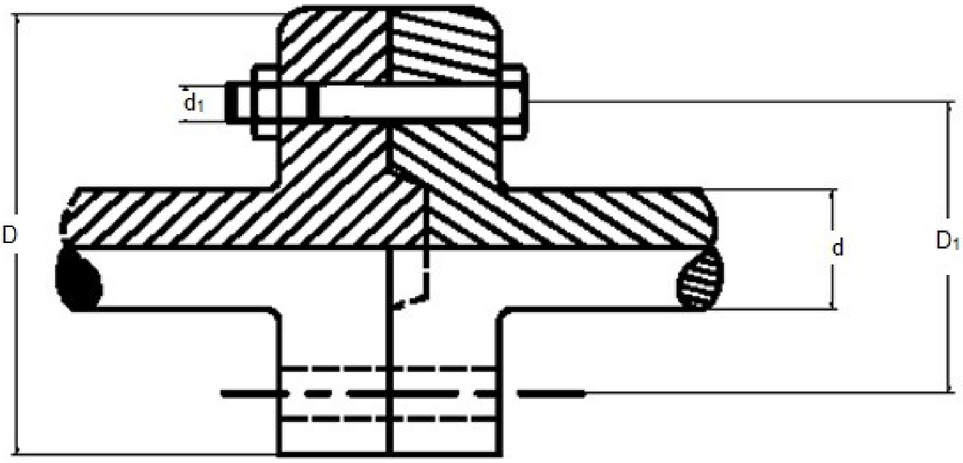


FIGURE 2: Solid rigid coupling

The parameters for design purposes are:

The number of bolts, i, is $$i = \frac{1}{3}d + 5 \quad (16)$$

Flange thickness t, is $$t = (0.25d \text{ to } 0.28d) \quad (17)$$

The diameter of bolt circle, D_1, is $$D_1 = (1.4d \text{ to } 1.6d) \quad (18)$$

The outside diameter of flange D, is $$D = D_1 + (2d \text{ to } 3d) \quad (19)$$

The diameter of bolt, d_1, is $$d_1 = \sqrt{\frac{\zeta d^3 \tau_s}{2iD_1\tau_b}} \quad (20)$$

2.3 Design of Hollow Rigid Coupling

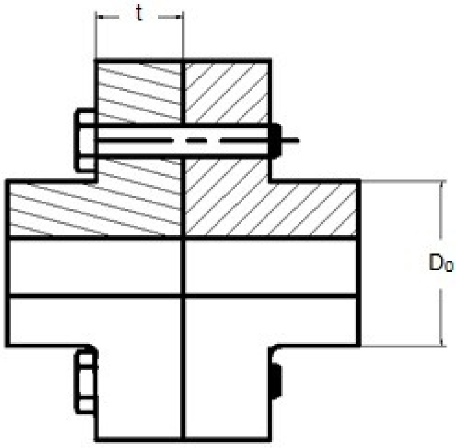


FIGURE 3: Hollow rigid coupling

The parameters for consideration as shown in figure 3 above [8] are as follows:

The outside diameter of Hollow rigid, D_0, is expressed as

$$D_o = 2.5d + 7.5cm \tag{21}$$

Minimum number of bolts, i, $i = \frac{D_o}{2}$ (22)

The diameter of bolt circles, D_1, $D_1 = 1.4D_o$ (23)

The mean diameterw of bolt, d_1 is $d_1 = \sqrt{\frac{(1-k^4)D_o^3\tau_s}{2iD_1\tau_b}}$ (24)

Where $k = \frac{d}{D_o}$

2.4 Design of Oldham Coupling

The length of the boss, L, is $L = 1.75dcm$ (25)

The diameter of the boss, D_2, is $D_2 = 2d$ (26)

The thickness of flange, t,, is $t = 0.75d$ (27)

Also diameter of Disc, D, is $D = 3L$ (28)

Distance between centre lines of shafts in Oldham's, a,

$$a = D - 3d) \tag{29}$$

Breadth of groove, W, is $$\mathrm{W} = D/6 \tag{30}$$

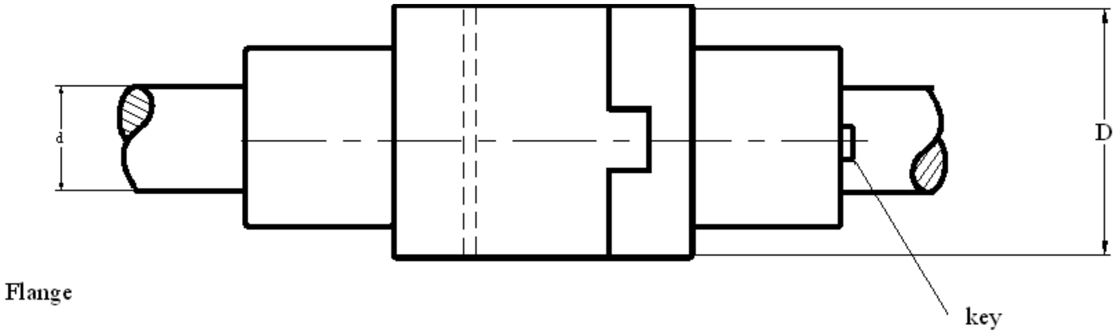


FIGURE 4: Oldham coupling

The thickness of the groove, h_1, is $$\mathrm{h}_1 = \frac{\mathrm{W}}{2} \tag{31}$$

The thickness of central disc, h, is $$\mathrm{h} = \frac{\mathrm{W}}{2} \tag{32}$$

The total pressure on each side of the coupling, F,

$$F = \frac{1}{4} pDh \tag{33}$$

Where $p \neq 85 kgf / cm^2$

The torque transmitted on each side of the coupling, M_{tc}

$$M_{tc} = 2Fh \text{ or } M_{tc} = pD^2h/6 \tag{34}$$

Power transmitted, N, is $$N = \frac{PD^2hn}{430000} \text{ hp} \tag{35}$$

$$N = 1.734 * 10^{-3} PD^2hn \text{ W}$$

Where n = speed in (r.p.m)

2.5 Design of Pin Type Flexible Coupling

With reference to figure 5 and its lettering [8] the below formular are derived.

The outside diameter, D, is $$D = 4d \tag{36}$$

The clearance, b, $$b = 0.1d \tag{37}$$

The hub diameter, D_2 is $$D_2 = 2d \tag{38}$$

The hub length, L, is $L = 1.75d$ (39)

Diameter of pin at the rock, d_1 $d_1 = d_p cm$ (40)

$$i = 0.2d + 3 \tag{41}$$

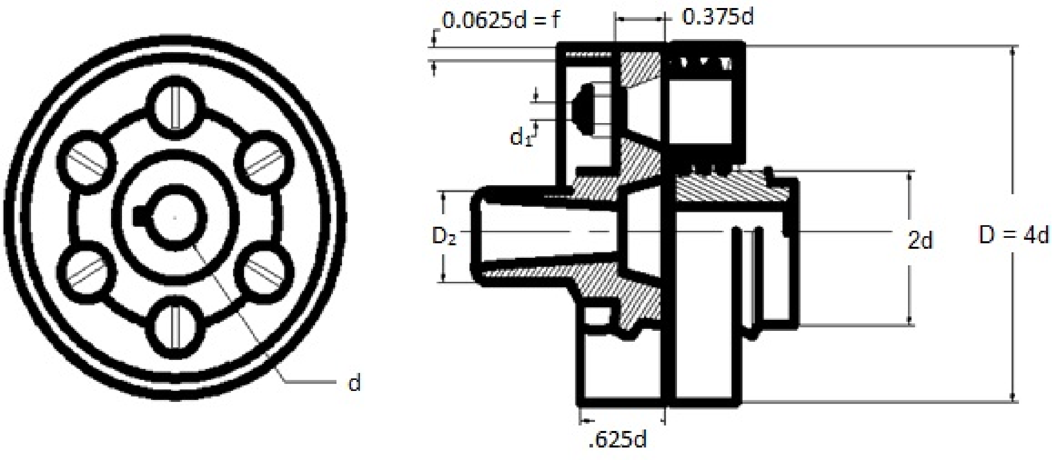


FIGURE 5: Pin type flexible coupling

Force at each Pin, *F*, is defined as $F = 0.785 d_p^{\,2} \tau_p$ (42)

Where τ_p = shear stress in pin = allowable shearing stress kgf/cm^2

Bending stress in Pin, σ_b $\sigma_b = F \dfrac{\left(l/2 + b\right)}{\pi/32\, d_p^{\,3}}$ (43)

The bearing pressure, $P_b\,kgf/cm^2$, is $P_b = \dfrac{F}{Ld^1}$ (44)

Where d_1= outside diameter of the bush and

$$d \ = d_2 + 0.115 d_1 + t$$

Where d_2 = diameter of hole for bolt, cm.

The torque transmitted, M_{tc}, is $M_{tc} = \dfrac{iFD_1}{2}$ (45)

2.6 Sleeve Coupling

With reference to figure 6 and its lettering [8] the below formular are derived:

The outside diameter of sleeve, $D = 2d + 1.3cm$ (46)

The length of the sleeve, L, is $L = 3.5d$ (47)

The length of the key, l, is $l = 3.5d$ (48)

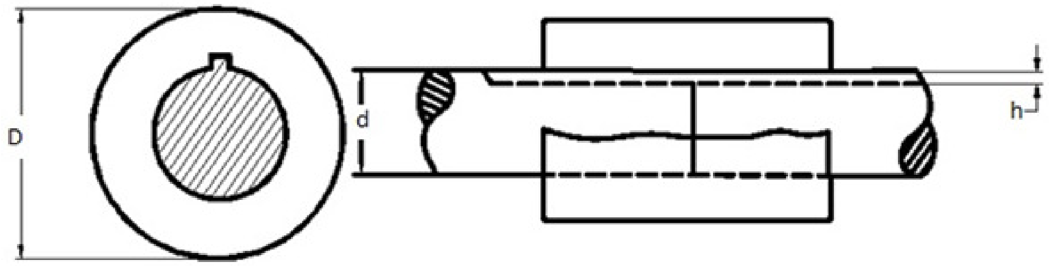


FIGURE 6: Sleeve coupling

The torque transmitted, M_{tc}

$$M_{tc} = \frac{\pi\zeta\tau_d d^2}{144} \tag{49}$$

The width of keyway, b,

$$b = \frac{2M_{tc}}{\tau_{d_2} ld} \tag{50}$$

Where τ_{d_2} = design shear stress in key

The thickness of key, h, is

$$h = \frac{2M_{tc}}{\sigma l_b^{1} ld} \tag{51}$$

Where $\sigma^1{}_b$ = design bearing stress for keys

2.7 Seller Cone Coupling

The length of the box L, is

$$L = \frac{3.65d + 4d}{2} \tag{52}$$

The outside diameter of the conical sleeve, D_1

$$D_1 = \left(\frac{1.875d + 4d}{2}\right) + 1.25cm \tag{53}$$

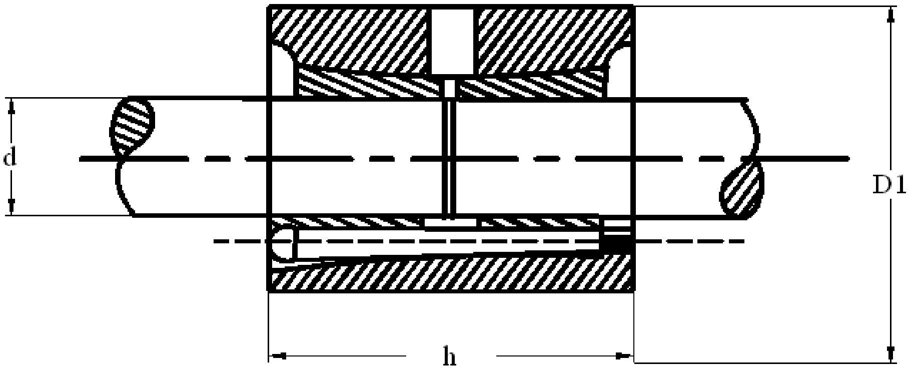


FIGURE 7: Seller cone coupling

The outside diameter of the box D_2 $$D_2 = 3d \tag{54}$$

The length of the conical sleeve, L, is $$L = 1.5d \tag{55}$$

2.8 Design of Split Muff Coupling

Refer Figure 8 [8]

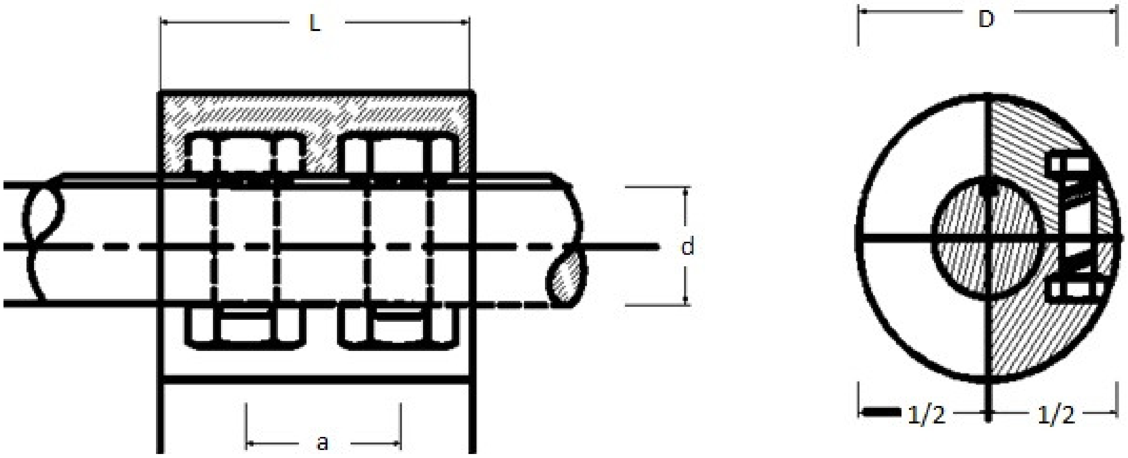


FIGURE 8: Split Muff coupling

The outside diameter of the sleeve, D, is $$D = 2d + 1.3cm \tag{56}$$

The length of the sleeve, L, $$L = (3.5d \;\; or \;\; 2.5d) + 5cm \tag{57}$$

The torque transmitted, M_{tc}, $$M_{tc} = \frac{\pi^2 d_c{}^2 \sigma_t \mu i d}{16} \tag{58}$$

Where d_c = core diameter of the clamping bolts, cm and i = number of bolts

2.8 Design of Pulley Flange Coupling Refer Figure. 9

The parameters for consideration are;

The number of bolts, i, is $i = 0.2d + 3$ (59)

Bolt diameter d_1 is, $d_1 = \frac{0.5d}{\sqrt{i}}$ (60)

The width of flange, l_1 $l_1 = 0.5d + 2.5cm$ (61)

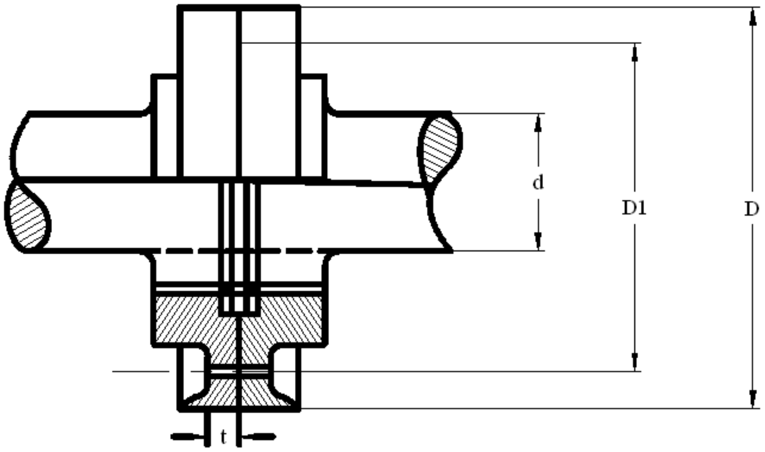


FIGURE 9: Pulley flange coupling

The thickness of the flange, t $t = 0.25d + 0.7cm$ (62)

The hubs length, l, is $l = 1.4d + 1.75cm$ (63)

The hub diameter, D_2 is, $D_2 = 1.8d + 1cm$ (64)

The average value of the diameter of the bolt circle, D_1,

$$D_1 = 2d + 2.5cm \quad (65)$$

The outside diameter of flange, D_f $D_f = 2.5d + 7.5cm$ (66)

2.10 Design of Fairbain's Lap-Box Coupling

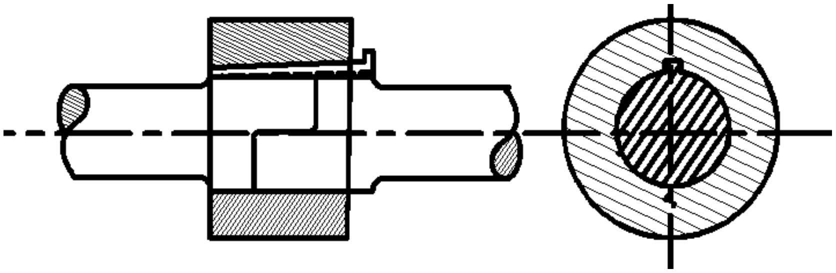


FIGURE 10: Fairbain's lap-box coupling

The outside diameter of sleeve, D, is $D = 2d + 1.3cm$ (67)

Length of lap, l, $l = (0.9d + 0.3)cm$ (68)

The length of sleeve L, is $L = 2.25d + 2c$ (69)

3.0 METHODOLOGY

Using the above design equations and procedures, a CAD System/Software for determining necessary coupling parameters and generating automatic drawings of the shaft for a particular application was developed. The design sequence shown in figure 11 was adopted for easier programming. The software was developed with JAVA programming language, which is users' friendly and readily compatible with Microsoft Windows environment. The development of COUPLINGCAD involves; creating the user interface, setting object properties and writing of codes. And these were later tested to see if the design codes give the right result.

If COULINGCAD is installed on any computer system, when it is clicked to be used, the opening screen features that can be seen is shown in figure 12. As the next button on the opening environment is clicked, this bring out the various couplings (see figure 13) which will give room for users to be able to select the intended type to be considered for any given engineering design problems. For example, if flange coupling is clicked, this takes the user to the design environment as shown in Figure 14.

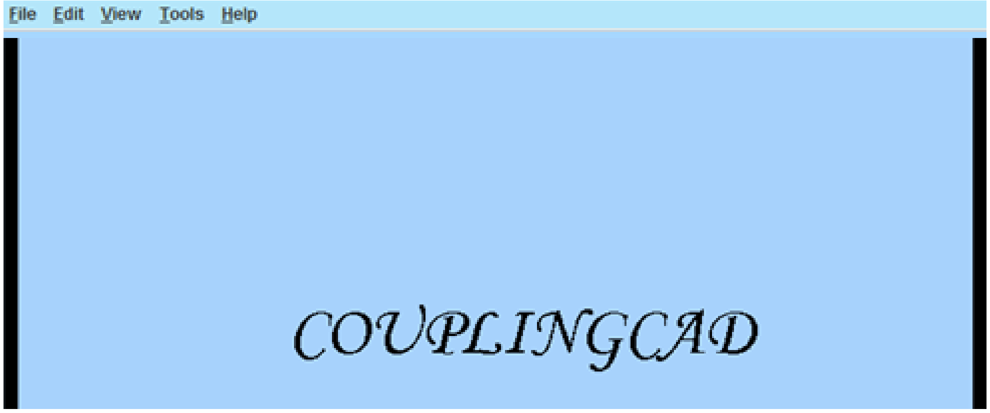
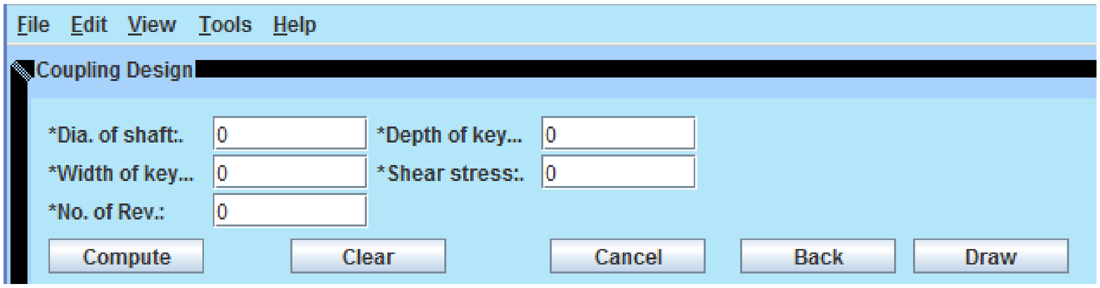


FIGURE 12 : COUPLINGCAD Main Entry Screen

FIGURE 13: COUPLINGCAD Main Menu Globe



*Dia. of shaft:. 0
*Depth of key... 0
*Width of key... 0
*Shear stress:. 0
*No. of Rev.: 0

FIGURE 14: COUPLINGCAD Main Dimension Menu

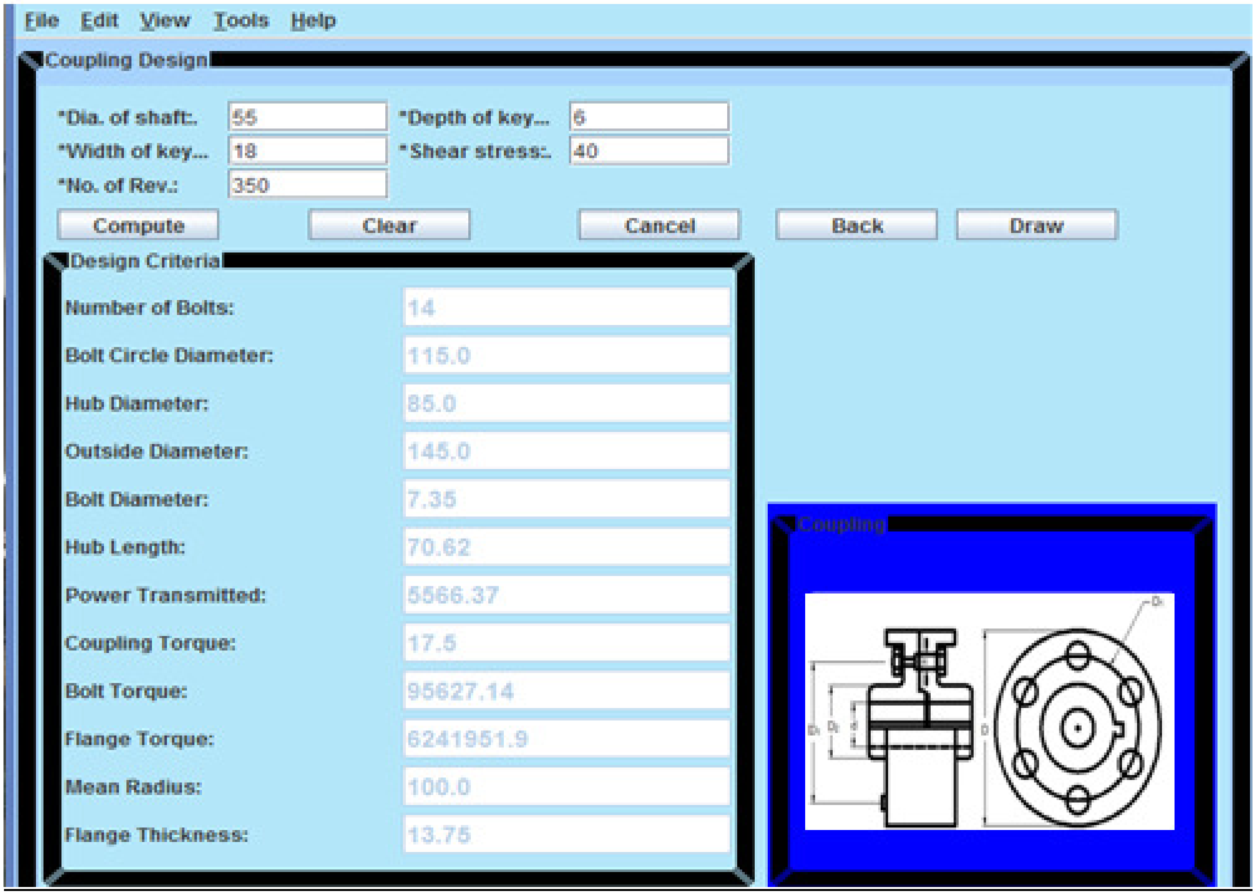
4.0 RESULTS AND DISCUSSION

Case studies of samples problems from standard text materials were considered to test or validate the software and by comparing the results got with manually generated solution. Few of these examples are presented below:

4.1 Case Study I

Design a flange coupling to connect two shafts each of 55cm diameter transmitting at 350 r.p.m. with allowable shear stress of 40N/cm^2. The width and depth of the keyway is 18cm and 6cm respectively.

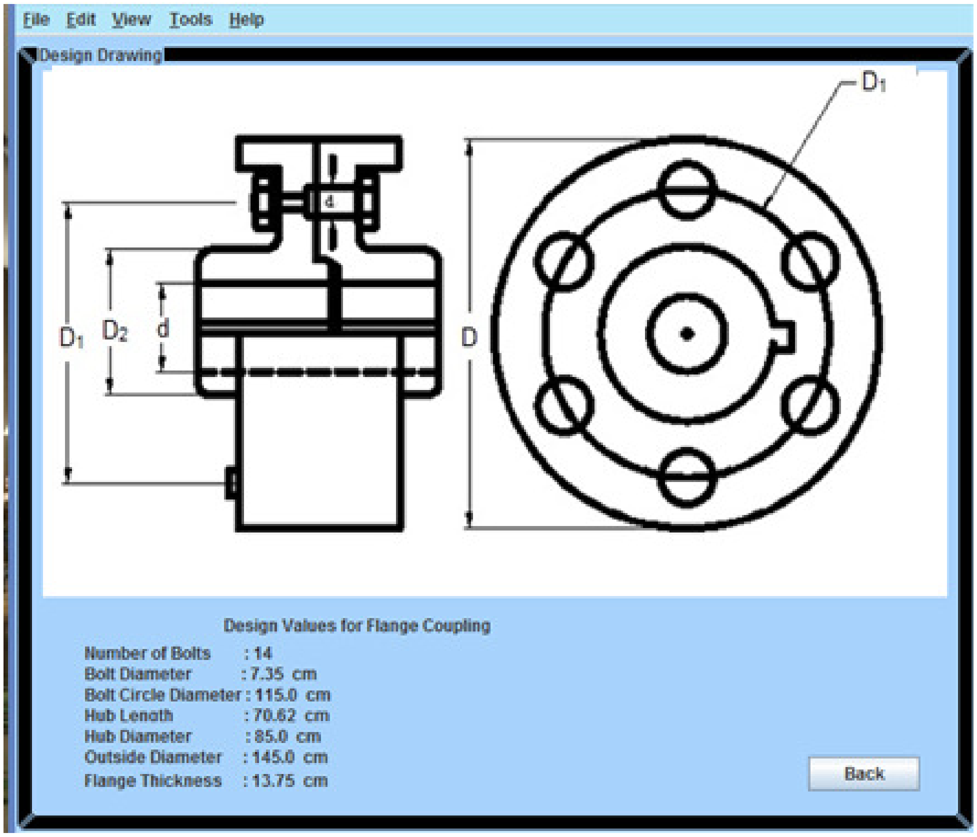
Solution: These values are being input into the package as seen in figure 15. The result is as shown in figure 16



*Dia. of shaft:. 55
*Width of key... 18
*No. of Rev.: 350
*Depth of key... 6
*Shear stress:. 40

Number of Bolts: 14
Bolt Circle Diameter: 115.0
Hub Diameter: 85.0
Outside Diameter: 145.0
Bolt Diameter: 7.35
Hub Length: 70.62
Power Transmitted: 5566.37
Coupling Torque: 17.5
Bolt Torque: 95627.14
Flange Torque: 6241951.9
Mean Radius: 100.0
Flange Thickness: 13.75

FIGURE 15: Snapshot showing the input parameters for flange coupling



```
Design Values for Flange Coupling

Number of Bolts        : 14
Bolt Diameter          : 7.35  cm
Bolt Circle Diameter   : 115.0  cm
Hub Length             : 70.62  cm
Hub Diameter           : 85.0  cm
Outside Diameter       : 145.0  cm
Flange Thickness       : 13.75  cm
```

FIGURE 16: Snapshot showing output parameters design values for flange coupling

Manually Solved Solution to Case Study I

Given parameters are:

Diameter, d= 55cm, allowable stress in shaft, τ_s = 40N/cm^2, Speed, n= 350 r.p.m, Width of keyway, x= 18, Depth of keyway, y= 6cm.

Using the designed equations spelt above, the following parameters were calculated for.

Appropriate number of bolts needed, $I = 0.2d + 3 = 0.2 \times 55 + 3 = 14 bolts$

Bolt circle diameter, $D_1 = 2d + 5 = 2 \times 55 + 5 = 115cm$

The hub diameter, $D_2 = 1.5d + 2.5cm = 1.5 \times 55 + 2.5 = 85cm$

Outside diameter of flange, $D = 2.5d + 7.5 = 2.5 \times 55 + 7.5 = 145cm$

Hub length, $L = 1.25d + 1.875 = 1.25 \times 55 + 1.875 = 70.62cm$

Power transmitted, $N = \dfrac{d^3 \Pi n \xi \tau_s}{1558400}$,

But, ξ = factor which takes care of the reduced strength due to keyway.

$$\zeta = 1 + \frac{0.2x + 1.1y}{d} = 1 + \frac{0.2\times 18 + 1.1\times 6}{55} = 1.185$$

$$N = \frac{55^3 \times \Pi \times 350 \times 1.185 \times 40}{1558400} = 5566.37$$

The torque transmitted by the coupling, M_{tc}

$$M_{tc} = \frac{97400}{n} N = \frac{97400 \times 55566.37}{350} = 17.5Ncm$$

Bolt diameter, $d_1 = \frac{0.5d}{\sqrt{i}} = \frac{0.5\times 55}{\sqrt{14}} = 7.35$

Bolt torque, $M_{tb} = i\left(\frac{\Pi d_1^{\ 2}}{4}\right)\tau_b \frac{D_1}{2}$

The allowable or design stress in bolts, $\tau_b = \frac{779200N}{\pi d_1^{\ 2} inD_1}$

$$\tau_b = \frac{779200\times 5566.37}{\pi \times 7.33^2 \times 14 \times 350 \times 115} = 45\,N/cm^2$$

$$\therefore M_{tb} = 14\left(\frac{\Pi \times 7.33^2}{4}\right)\times 45 \times \frac{115}{2} = 95627.14Ncm$$

Flange torque, $M_{tsf} = t(\pi D_2)\tau_f \frac{D_2}{2}$

Flange thickness, $t = 0.25d = 0.25 \times 55 = 13.75cm$

Design shear stress, $\tau_f = \frac{2M_{tc}}{\pi D_1^{\ 2} t}$

$$\tau_f = \frac{2\times 17.5}{\pi \times 115^2 \times 13.75} = 61\,N/m^2$$

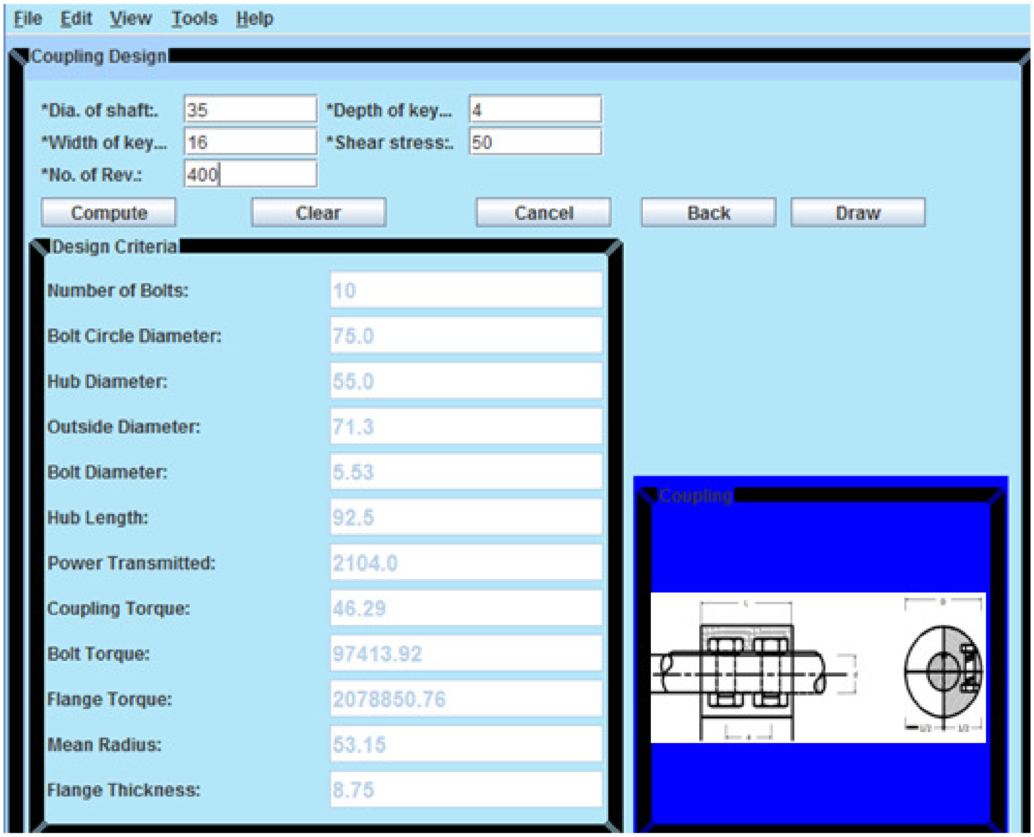
$$M_{tsf} = 13.75(\pi \times 85)61 \times \frac{85}{2} = 6241951.9Ncm$$

The mean radius, $r_m = \left(\frac{D+d}{2}\right)cm = \frac{145+55}{2}cm = 100cm$

4.2 Case Study II

Design a split muff coupling to join two shafts of diameter 35cm, rotating at a speed of 400 r.p.m. and has shear stress of 50N/cm^2. The keyway's width and depth are 16cm and 4cm respectively.

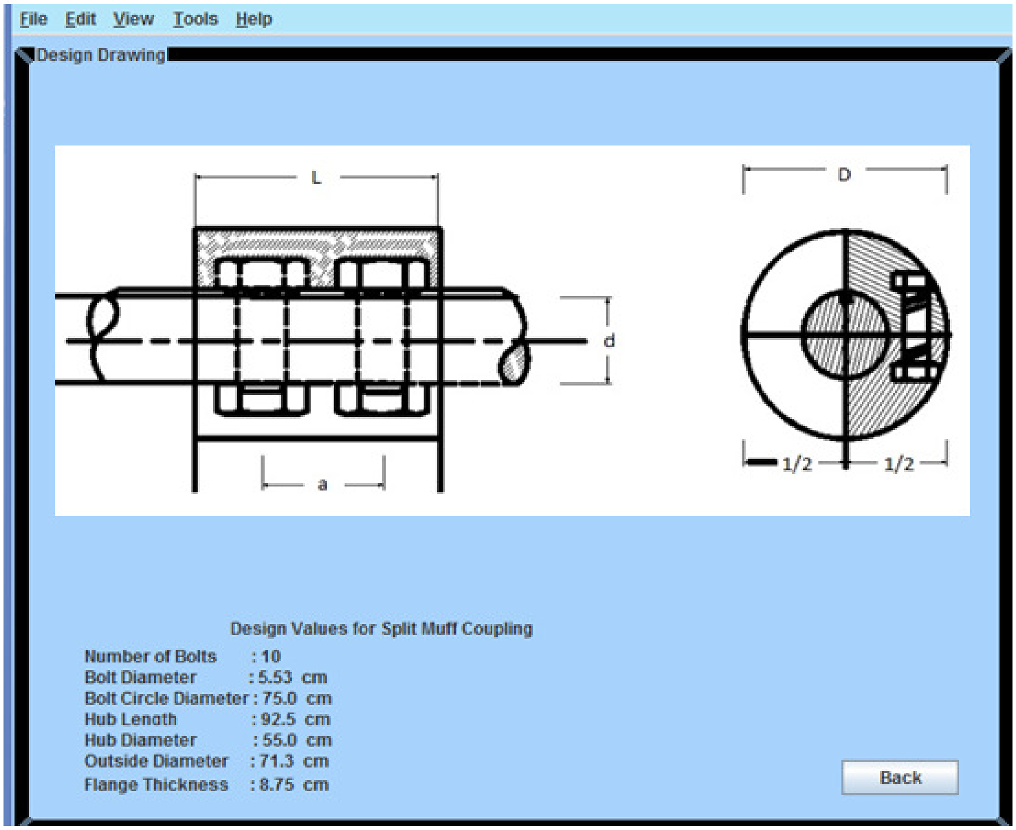
Solution:These values are being input into the package as seen in figure 17. The result is as well shown in figure 18.



*Dia. of shaft:. 35
*Depth of key... 4
*Width of key... 16
*Shear stress:. 50
*No. of Rev.: 400

Number of Bolts: 10
Bolt Circle Diameter: 75.0
Hub Diameter: 55.0
Outside Diameter: 71.3
Bolt Diameter: 5.53
Hub Length: 92.5
Power Transmitted: 2104.0
Coupling Torque: 46.29
Bolt Torque: 97413.92
Flange Torque: 2078850.76
Mean Radius: 53.15
Flange Thickness: 8.75

FIGURE 17: Snapshot showing the input parameters for split-muff coupling



Design Values for Split Muff Coupling

```
Number of Bolts      : 10
Bolt Diameter        : 5.53  cm
Bolt Circle Diameter : 75.0  cm
Hub Length           : 92.5  cm
Hub Diameter         : 55.0  cm
Outside Diameter     : 71.3  cm
Flange Thickness     : 8.75  cm
```

FIGURE18: Snapshot showing output parameters design values for split-muff coupling

Manually Solved Solution to Case Study II

Given parameters are:

Shaft diameter, d=35cm, allowable stress in shaft, $\tau_{s=}$50N/cm^2, speed, n= 400 r.p.m., width of keyway, x= 16cm, depth of keyway, y= 4cm.

Using the designed equations spelt above, the following parameters were calculated for:

Appropriate number of bolts needed, $I = 0.2d + 3 = 0.2 \times 35 + 3 = 10 bolts$

Bolt circle diameter, $D_1 = 2d + 5 = 2 \times 35 + 5 = 75cm$

The sleeve diameter, $D_2 = 1.5d + 2.5cm = 1.5 \times 35 + 2.5 = 55cm$

Outside diameter of sleeve, $D = 2d + 1.3cm = 2 \times 35 + 1.3cm = 71.3cm$

Sleeve length, $L = 2.5d + 5cm = 2.5 \times 35 + 5 = 92.5cm$

Power transmitted, $N = \dfrac{d^3 \Pi n \xi \tau_s}{1558400}$

ξ = factor which takes care of the reduced strength due to keyway.

$$\zeta = 1 + \frac{0.2\times16+1.1\times4}{35} = 1.217$$

$$N = \frac{35^3\times\Pi\times400\times1.217\times50}{1558400} = 2104K\mathrm{w}$$

The torque transmitted by the coupling, M_{tc}

$$M_{tc} = \frac{\pi^2 d_c{}^2\sigma_t\mu id}{16} = \frac{\pi^2\times d_c{}^2\times\sigma_t\times\mu\times10\times35}{16} = 46.29Ncm$$

Bolt diameter, $d_1 = \frac{0.5d}{\sqrt{i}} = \frac{0.5\times35}{\sqrt{10}} = 5.53cm$

Bolt torque, $M_{tb} = i\left(\frac{\Pi d_1{}^2}{4}\right)\tau_b\frac{D_1}{2}$

But, the allowable or design stress in bolts, is

$$\tau_b = \frac{779200N}{\pi d_1{}^2 inD_1} = \frac{779200\times2104}{\pi\times5.53^2\times10\times400\times75} = 57\,N/cm^2$$

$$M_{tb} = 10\left(\frac{\Pi\times5.53^2}{4}\right)\times57\times\frac{75}{2} = 97413.92$$

Sleeve torque, $M_{tsf} = t(\pi D_2)\tau_f\frac{D_2}{2}$

Sleeve thickness, $t = 0.25d = 0.25\times35 = 8.75cm$

Design shear stress, $\tau_f = \frac{2M_{tc}}{\pi D_1{}^2 t} = \frac{2\times46.29}{\pi\times75^2\times8.75} = 60\,N/cm^2$

$$M_{tsf} = 8.75(\pi\times55)60\times\frac{55}{2} = 2078850.76Ncm$$

The mean radius, $r_m = \left(\frac{D+d}{2}\right)cm = \frac{71.3+35}{2}cm = 53.15cm$

4.3 Comparative View of the Results for Case Study I and Case Study II

The means to excellently validate the package developed is to compare the results obtained.

From table 1 shown, comparatively, the results got when manually solved or calculated for is same as the results got using the COUPLINGCAD for the various designed parameters for both case study I and case study II. This implies that the COUPLINGCAD is excellently packaged.

Case Study I				**Case Study II**		
Design Parameters	**COUPLING CAD Result**	**Manually Solved Result**		**Design Parameters**	**COUPLING CAD Result**	**Manually Solved Result**
Number of bolts	14	14		Number of bolts	10	10
Bolt circle diameter	115.0cm	115.0cm		Bolt circle diameter	75.0cm	75.0cm
Hub diameter	85.0cm	85.0cm		Sleeve diameter	55.0cm	55.0cm
Outside diameter	145.0cm	145.0cm		Outside diameter	71.3cm	71.3cm
Bolt diameter	7.35cm	7.35cm		Bolt diameter	5.53cm	5.53cm
Hub length	70.62cm	70.62cm		Sleeve length	92.5cm	92.5cm
Power	5566.37Kw	5566.37Kw		Power	2104.0Kw	2104.0Kw
Coupling torque	17.5	17.5		Coupling torque	46.29Ncm	46.29Ncm
Bolt torque	95627.14	95627.14		Bolt torque	97413.92	97413.92
Flange torque	6241951.9	6241951.9		Sleeve torque	2078850.76	2078850.76
Mean radius	100cm	100cm		Mean radius	53.15cm	53.15cm
Flange thickness	13.75cm	13.75cm		Sleeve thickness	8.75cm	8.75cm

TABLE 1: Tabulated results of designed parameters using the COUPLINGCAD and manually solved approach for both Case study I and II

5.0 CONCLUSION

The case studies considered proved that the COUPLINGCAD is excellently packaged and it could be used in solving problems related to the various couplings of choice as discussed. The COUPLINGCAD is quite accurate with high precision. It is noticed that with it, time management is ensured as calculations are done within a short time which is not so with the manually solved approach.

6.0 REFERENCES

[1] R. N. Peter Childs. Mechanical Design, 1st Edition,NewYorkToronto. John Wiley and Sons Inc 1998, pp.124-143

[2] R. S.Khurmi and J. K. Gupta.Machine Design, 14th edition,New Delhi EURASIA publishing house,.2005, pp. 481-505

[3] D. N. Reshetor. Machine Design, 1st Edition, Moscow.MIR Publishers, 1978. pp.592-645

[4] T.I. Ogedengbe, 2002. Computer-Aided Design of Power Transmission Shaft M.Eng Thesis, Federal University of Technology, Akure, Nigeria.

[5] M.K. Adeyeri Development of Software Package For The Design of Special Transmission Element, M.Eng Thesis, University of Ilorin, Kwara state, Nigeria.2007.

[6] M.K. Adeyeri and M.B. Adeyemi."Development of Software package in Designing chain drives as Engineering Transmission Element". Advanced Materials Research, Trans Tech Publication Switzerland, vol 62-64: pp. 655-663,2009.

[7]. J.A. Akpobi,, R.O. Edopia, and M.H. Oladeinde, Computer Aided Design of Power Transmission Screw. 3rd International Conference on Engineering Research & Development: Advances in Engineering Science and Technology, University of Benin, Benin city,Nigeria, Sept. 6th-8th,2010, pp:1191-1221.

[8]. K Linguaiah and B. R. Narayana.Machine Design Data Handbook. Suma Publishers Jayanagar, Banga, India;1973. pp. 349 – 375.

Evolutionary Algorithm for Optimal Connection Weights in Artificial Neural Networks

G.V.R. Sagar *nusagar@gmail.com*
Assoc. Professor,
G.P.R. Engg. College.
Kurnool AP, 518007, India.

Dr. S. Venkata Chalam *sv_chalam2003@yahoo.com*
Professor,
CVR Engg. College,
Hyderabad AP, India

Manoj Kumar Singh *mksingh@manuroresearch.com*
Director,
Manuro Tech. Research,
Bangalore, 560097, India

Abstract

A neural network may be considered as an adaptive system that progressively self-organizes in order to approximate the solution, making the problem solver free from the need to accurately and unambiguously specify the steps towards the solution. Moreover, Evolutionary computation can be integrated with artificial Neural Network to increase the performance at various levels; in result such neural network is called Evolutionary ANN. In this paper very important issue of neural network namely adjustment of connection weights for learning presented by Genetic algorithm over feed forward architecture. To see the performance of developed solution comparison has given with respect to well established method of learning called gradient decent method. A benchmark problem of classification, XOR, has taken to justify the experiment. Presented method is not only having very probability to achieve the global minima but also having very fast convergence.

Keywords: Artificial Neural Network, Evolutionary Algorithm, Gradient Decent Algorithm, Mean Square Error.

1. INTRODUCTION

An Artificial Neural Network (ANN) is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if “questions. ANN can be viewed as weighted directed graphs in which artificial neurons are nodes and directed edges (with weights) are connections between neurons outputs and neuron inputs. Based on the connection pattern (architecture), ANN can be grouped into two categories: (a) Feed Forward Networks allow signals to travel one-way only, from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer as shown in Fig.(1) (b)Recurrent Networks can have signals traveling in both directions by introducing loops in the network

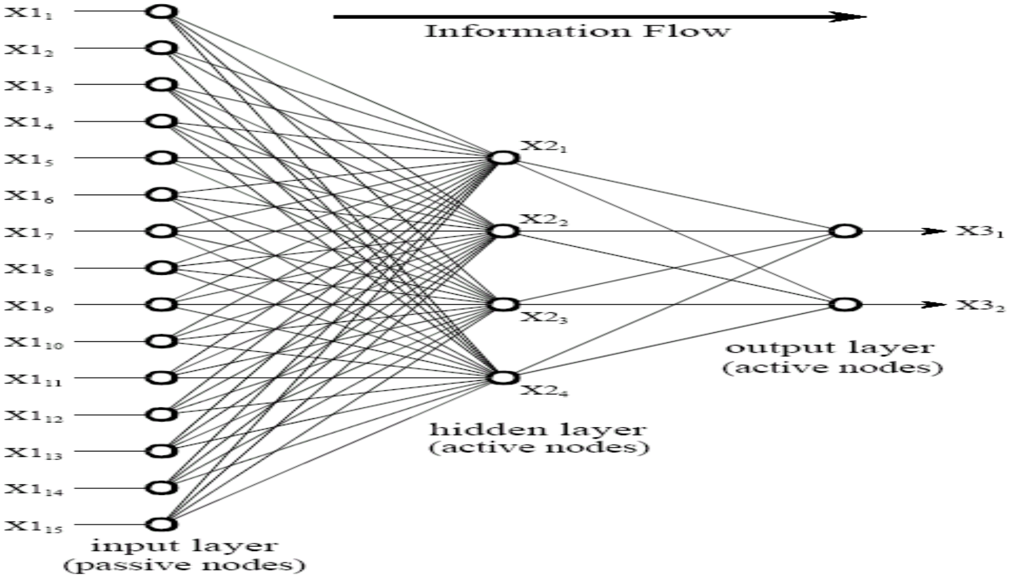


FIGURE 1: Feedforward architecture

Learning in artificial neural systems may be thought of as a special case of machine learning. Learning involves changes to the content and organization of a system's knowledge, enabling it to improve it's performance on a particular task or set of tasks. The key feature of neural networks is that they learn the input/output relationship through training. There are two types of training/learning used in neural networks, with different types of networks using different types of training. These are Supervised and Unsupervised training, of which supervised is the most common for feed forward architecture training modes. Supervised Learning which incorporates an external teacher, so that each output unit is told what its desired response to input signals ought to be. During the learning process global information may be required. Paradigms of supervised learning include error-correction learning. An important issue concerning supervised learning is the problem of error convergence, i.e. the minimization of error between the desired and computed unit values. The aim is to determine a set of weights, which minimizes the error. One well-known method, which is common to many learning paradigms, is the gradient decent based learning.

The idea behind learning in Neural Network is that, the output depends only in the activation, which in turn depends on the values of the inputs and their respective weights. The initial weights are not trained with respect to the inputs, which can result in error. Now, the goal of the training process is to obtain a desired output when certain inputs are given. Since the error is the difference between the actual and the desired output, the error depends on the weights, and we need to adjust the weights in order to minimize the error.

2. GRADIENT DESCENT LEARNING

Gradient descent is a first order optimization algorithm. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or of the approximate gradient) of the function at the current point. If instead one takes steps proportional to the positive of the gradient, one approaches a local maximum of that function; the procedure is then known as gradient ascent. Gradient descent is also known as steepest descent, or the method of steepest descent. A gradient descent based optimization algorithm such as back-propagation (BP) [6] can then be used to adjust connection weights in the ANN iteratively in order to minimize the error. The Gradient descent back-propagation algorithm [7] is a gradient descent method minimizing the mean square error between the actual and target output of multilayer perceptrons. The Back-propagation [8], [9] networks tend to be slower to train than other types of networks and sometimes require thousands of epochs. When a reduced number of neurons are used the Error Back-propagation algorithm cannot converge to the required training error. The most common mistake is in order to speed up the training process and to reduce the training errors, the neural networks with larger number of neurons than required. Such networks would perform very poorly for new patterns nor used for training[10]. Gradient descent is relatively slow close to the minimum. BP has drawbacks due to its use of gradient descent [11, [12]. It often gets trapped in a local minimum of the error function and is incapable of finding a global minimum if the error function is multimodal and/or non differentiable. A detailed review of BP and other learning algorithms can be found in [13], [14], and [15].

3. EVOLUTIONARY ARTIFICIAL NEURAL NETWORK

Evolutionary artificial neural networks (EANN's) refer to a special class of artificial neural networks (ANN's) in which evolution is another fundamental form of adaptation in addition to learning [2] – [5]. Evolutionary algorithms (EA's) are used to perform various tasks, such as connection weight training, architecture design, learning rule adaptation, input feature selection, connection weight initialization, rule extraction from ANN's, etc. One distinct feature of EANN's is their adaptability to a dynamic environment. The two forms of adaptation, i.e., evolution and learning in EANN's, make their adaptation to a dynamic environment much more effective and efficient. Evolution has been introduced into ANN's at roughly three different levels: connection weights, architectures, and learning rules. The evolution of connection weights introduces an adaptive and global approach to training, especially in the reinforcement learning and recurrent network learning paradigm where gradient-based training algorithms often experience great difficulties. The evolution of architectures enables ANN's to adapt their topologies to different tasks without human intervention and thus provides an approach to automatic ANN design as both ANN connection weights and structures can be evolved.

4. EVOLUTION OF CONNECTION WEIGHTS

Weight training in ANN's is usually formulated as minimization of an error function, such as the mean square error between target and actual outputs averaged over all examples, by iteratively adjusting connection weights. Most training algorithms, such as BP. and conjugate gradient algorithms are based on gradient descent. There have been some successful applications of BP in various areas .One way to overcome gradient-descent-based training algorithms' shortcomings is to adopt EANN's, i.e., to formulate the training process as the evolution of connection weights in the environment determined by the architecture and the learning task. EA's can then be used effectively in the evolution to find a near-optimal set of connection weights globally without computing gradient information. The fitness of an ANN can be defined according to different needs. Two important factors which often appear in the fitness (or error) function are the error between target and actual outputs and the complexity of the ANN. Unlike the case in gradient-descent-based training algorithms, the fitness (or error) function does not have to be differentiable or even continuous since EA's do not depend on gradient information. Because EA's can treat large, complex, nondifferentiable, and multimodal spaces, which are the typical case in the real world, considerable research and application has been conducted on the evolution of connection weights .The aim is to find a near-optimal set of connection weights globally for an ANN with a fixed architecture using EA's. Comparisons between the evolutionary approach and conventional training algorithms, such as BP, will be made over XOR classification problem.

4.1 Evolution of Connection Weights Using GA

```
% initialization of population
sz = total weights in architecture;
For i = 1: popsize;
    pop(i)=sz number of random number;
```

```
4.  End

% offspring population creation
5.  For j=1: popsize/2;
6.    pickup two parents randomly through uniform distribution;
7.    cp=cross over position defined by randomly pickup any active node position;
8.    To create offspring, exchange all incoming weights to selected nodes cp between parents;
9.        For each offspring;
10.,          place of mutation, mp = randomly selected active node;
11.             For all incoming weights w to selected node mp;
12.                 w=w+N (0, 1);
13.             End
14.       End
15. End

16. Offspring population, off_pop available;
17. npop= [pop; off_pop];

% Define fitness of each solution,
18. For i=1:2*popsize;
19.     wt=npop(i);
20.     apply wt to ANN architecture to get error value;
21.     define fitness as fit(i)=1/error;
22.   End

% Tournament selection
23. For r =1:2*popsize;
24.   pick P number of Challengers randomly, where P = 10% of popsize;
25.   arrange the tournament w.r.t fitness between rth solution and selected P challengers.;
26.   define score of tournament for rth solution
27. End
28.   Arrange score of all solution in ascending order;
29.   sp=pick up the best half score position ;
30. select next generation solution as solution corresponding to position sp;

31. repeat the process from step 5 until terminating criteria does not satisfy
32. final solution=solution with maximum fitness in last generation.
```

5. EXPERIMENTAL SETUP

A fully interconnected feed forward architecture of size [2-2-1] / [2 3 1] designed .transfer function in the active node is taken as unimodel sigmoid function. Initial random weights are upgraded by gradient decent and genetic algorithm respectively. Various learning rates have applied to capture performance possibilities from gradient decent. To increase the learning and efficiency 'bias' in architecture and 'momentum' in learning have also included when learning given by gradient decent. Population size in GA taken as 20 and 10 independent trails have given to get the generalize behavior. Condition of terminating criteria is taken as fixed iteration and it is equal to 500 for GA. Because GA works with a population at time where as gradient decent takes only one solution in each iteration hence to nullify the effect , more number of iterations have given to gradient decent learning and it is taken as 20*500;

5.1 Performance Shown by Gradient Decent Learning

With the defined size of architecture, bias has applied with +1 input for hidden layer and output layer. Various learning rate taken from 0.1 to 0.9 with the increment of 0.1 along with momentum constant as 0.1.in the Fig(2) performance has shown for architecture size [2 2 1] where as in Fig(3) with size [2 3 1].Mean square error obtained after 10,000 iterations has shown in Table 1 and in Table 2.

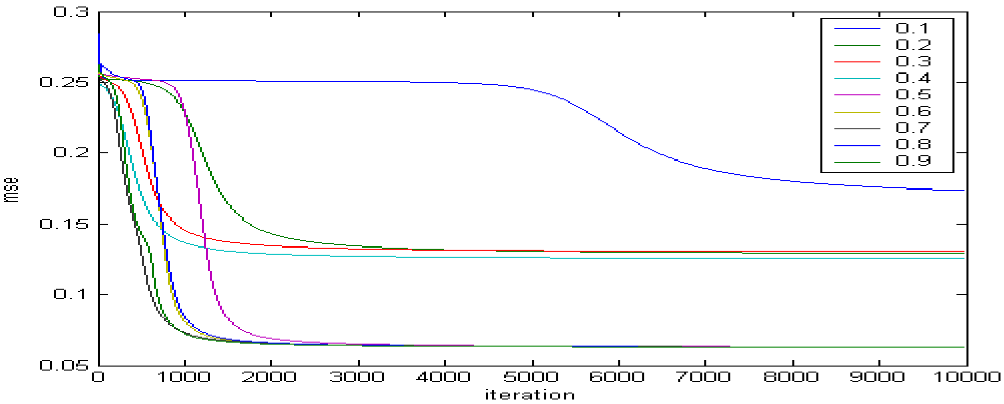


FIGURE 2: MSE performance with various learning rate .

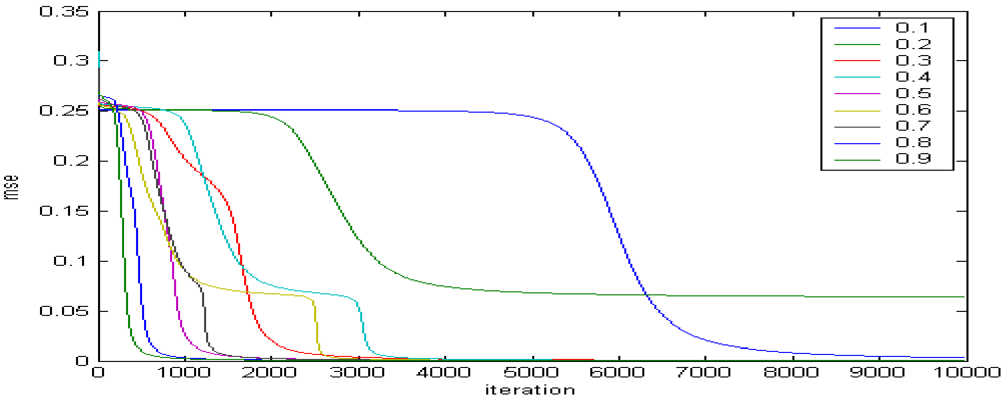


FIGURE 3: MSE performance with various learning rate

Learning rate	MSE([2 2 1])	MSE([2 3 1])
0.1	1.7352 e-001	3.2613 e-003
0.2	1.2920 e-001	6.3886 e-002
0.3	1.3042 e-001	4.4409 e-004
0.4	1.2546 e-001	3.7232 e-004
0.5	6.2927 e-001	2.6024 e-004
0.6	6.2825 e-001	2.3994 e-004
0.7	6.2915 e-001	2.0970 e-004
0.8	6.2868 e-001	1.3542 e-004
0.9	6.2822 e-001	1.2437 e-004

TABLE 1: performance shown by gradient decent

5.2 Performance Shown by GA Based Learning

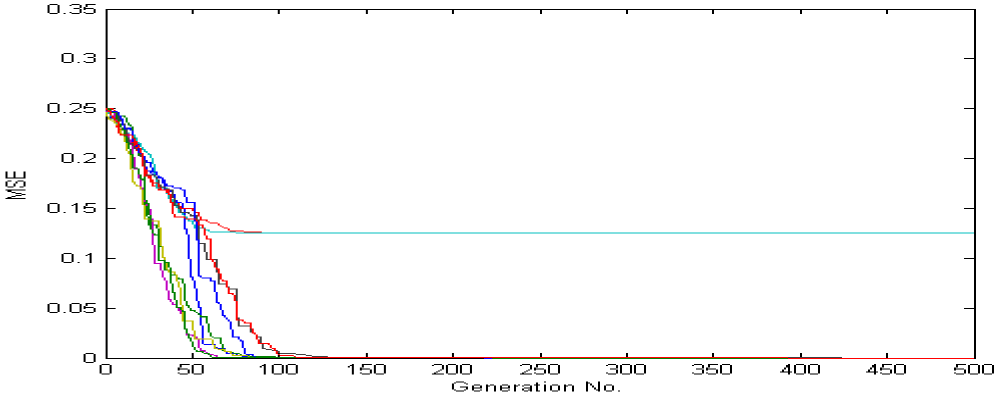


FIGURE 4: MSE performance by GA for different trails in [2 2 1]

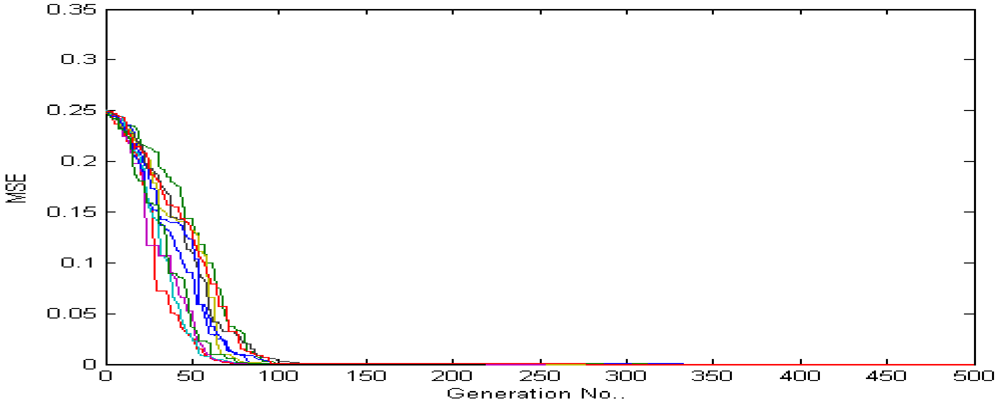


FIGURE 5: MSE performance by GA for different trails in [2 3 1]

Trail No.	MSE([2 2 1])	MSE([2 3 1])
1	8.8709 e-055	3.6112 e-028
2	2.5941 e-022	4.6357 e-031
3	***1.2500 e-001***	7.5042 e-032
4	***1.2500 e-001***	4.2375 e-037
5	3.2335 e-044	6.0432 e-049
6	2.5765 e-041	1.1681 e-035
7	9.6010 e-022	9.3357 e-032
8	3.5481 e-047	4.4852 e-030
9	3.9527 e-050	1.1725 e-033
10	6.3708 e-023	2.5171 e-036

TABLE 2: performance shown by GA for different trails

Results shown in Table1 indicate the difficulties associated with gradient decent based learning rule. Performance is very poor with the architecture size [2 2 1] for all learning rates. In fact learning failed for this case. This is indication of stuckness in local minima. For architecture [2 3 2] there is an improvement in reduction of mean square error, and with higher value of learning rate equal to 0.9, best performance has obtained. Convergence characteristics for both cases have shown in Fig (1) and in Fig (2). convergence characteristics performance of developed form of GA for weight adjustment shown in Fig(4) and in Fig(5).in both cases it is very clear that very fast convergence with high reliability can be achieve by GA (except for trail number 3 and 4)as shown in Table 2.

6. CONCLUSION

Determination of optimal weights in ANN in the phase of learning has obtained by using the concept of genetic algorithm. Because of direct form realization in defining the solution of weights there is no extra means required to represent the solution in population. Proposed method of weights adjustment has compared with the gradient decent based learning and it has shown proposed method outperform at every level for XOR classification problem. Even with lesser number of hidden nodes where gradient decent method is completely fail for learning, proposed method has shown very respectable performance in terms of convergence as well as accuracy also. Defined solution of learning has generalized characteristics from application point of view and having simplicity in implementation.

7. REFERENCES

[1] X. Yao, “Evolution of connectionist networks,” in Preprints Int. Symp. AI, Reasoning & Creativity, Queensland, Australia, Griffith Univ., pp. 49–52. 1991.

[2] “A review of evolutionary artificial neural networks,” Int. J. Intell. Syst., vol. 8, no. 4, pp. 539–567, 1993.

[3] “Evolutionary artificial neural networks,” Int. J. Neural Syst., vol. 4, no. 3, pp. 203–222, 1993.

[4] T. Dartnall, Ed. Dordrecht, “The evolution of connectionist networks,” in Artificial Intelligence and Creativity, The Netherlands: Kluwer, pp. 233–243, 1994.

[5] A. Kent and J. G. Williams, “Evolutionary artificial neural networks,” in Encyclopedia of Computer Science and Technology, vol. 33, , Eds. New York: Marcel Dekker, pp. 137–170, 1995.

[6] G. E. Hinton, “Connectionist learning procedures,” Artificial Intell., vol. 40, no. 1–3, pp. 185–234, Sept. 1989

[7] Rumelhart D. E., Hinton G. E., Williams R. J. “Learning representations by back propagating errors”, .Nature, 323, 533-536, 1986.

[8] Rumclhart D. E., Hinton G. E., Williams R. J.: “ Learning errors” , Nature, Vol. 323, pp. 533-536, 1986.

[9] Wcrobs P. J,: ” Back-propagation: Past and Future”, Proc. Neural Networks, San Diego, CA, 1, 343-354, 1988.

[10] Wilamowski B. M,: “ Neural Network Architectures and Learning Algorithms: How not to be Frustrated with Neural Networks, IEEE Industrial Electronics Magazine “, Vol. 3 No. 4, pp. 56-63, Dec. 2009.

[11] R. S. Sutton, “Two problems with back-propagation and other steepest-descent learning procedures for networks,” in Proc. 8th Annual Conf. Cognitive Science Society. Hillsdale, NJ: Erlbaum, pp. 823–831, 1986

.

[12] D. Whitley, T. Starkweather, and C. Bogart, “Genetic algorithms and neural networks: Optimizing connections and connectivity,” Parallel Comput., vol. 14, no. 3, pp. 347–361, 1990

[13] J. Hertz, A. Krogh, and R. Palmer, Introduction to the Theory of Neural Computation. Reading, MA: Addison-Wesley, 1991.

[14] D. R. Hush and B. G. Horne, “Progress in supervised neural networks,” IEEE Signal Processing Mag., vol. 10, pp. 8–39, Jan. 1993.

[15] Y. Chauvin and D. E. Rumelhart, Eds., Back-propagation: Theory, Architectures, and Applications. Hillsdale, NJ: Erlbaum, 1995.

[16] J. Hertz, A. Krogh, and R. Palmer, Introduction to the Theory of Neural Computation. Reading, MA: Addison-Wesley, 1991.

[17] D. R. Hush and B. G. Horne, “Progress in supervised neural networks,” IEEE Signal Processing Mag., vol. 10, pp. 8–39, Jan. 1993.

[18] Y. Chauvin and D. E. Rumelhart, Eds., Backpropagation: Theory, Architectures, and Applications. Hillsdale, NJ: Erlbaum, 1995.

[19] M. F. Møller, “A scaled conjugate gradient algorithm for fast supervised learning,” Neural Networks, vol. 6, no. 4, pp. 525–533, 1993.

[20] K. J. Lang, A. H. Waibel, and G. E. Hinton, “A time-delay neural network architecture for isolated word recognition,” Neural Networks, vol. 3, no. 1, pp. 33–43, 1990.

[21] S. Knerr, L. Personnaz, and G. Dreyfus, “Handwritten digit recognition by neural networks with single-layer training,” IEEE Trans. Neural Networks, vol. 3, pp. 962–968, Nov. 1992.

[22] S. S. Fels and G. E. Hinton, “Glove-talk: A neural network interface between a data-glove and a speech synthesizer,” IEEE Trans. Neural Networks, vol. 4, pp. 2–8, Jan. 1993.

[23] D. Whitley, T. Starkweather, and C. Bogart, “Genetic algorithms and neural networks: Optimizing connections and connectivity,” Parallel Comput., vol. 14, no. 3, pp. 347–361, 1990.

[24] D. Whitley, “The GENITOR algorithm and selective pressure: Why rank-based allocation of reproductive trials is best,” in Proc. 3rd Int. Conf. Genetic Algorithms and Their Applications, J. D. Schaffer, Ed. San Mateo, CA: Morgan Kaufmann, pp. 116–121, 1989.

[25] P. Zhang, Y. Sankai, and M. Ohta, “Hybrid adaptive learning control of nonlinear system,” in Proc. 1995 American Control Conf. Part 4 (of 6), pp. 2744–2748, 1995.

Reflectivity and Braggs Wavelength in FBG

Dinesh Arora ecedinesh@rediffmail.com
Research Scholar (ECE), Singhania University,
Jhunjhunu, Rajasthan-333515, INDIA

Dr. Jai Prakash ecedinesh@rediffmail.com
Professor in Department of Applied Physics,
Indo Global College of Engineering,
Mohali, Punjab-140109,INDIA

Hardeep Singh* hardeep_saini17@yahoo.co.in
Associate Professor in Department of ECE,
Indo Global College of Engineering,
Mohali, Punjab-140109, INDIA

Dr.Amit Wason wasonamit13@gmail.com
Professor in Department of ECE,
Rayat & Bahra Institute of Engineering & Bio-Technology,
Mohali, Punjab-140104, INDIA

Abstract

We have presented an analytical model of splitters based on Fiber Bragg Grating used to detect aBragg wavelength from the number of wavelengths which are traveling in an optical fiber. The number of grids and grating length can be used as a wavelength shifter. This paper presents experimental results that are used to show the effect of number of grids,the lengthof the grating on the Bragg wavelength and reflectivity of Fiber Bragg Grating (FBG).The pitch of grating is directly proportional to thegrating length and inversely proportional to number of grids. When the grating length is fixed and the number of grids is increased, the Bragg wavelength decreases resulting in increased reflectivity. This increased reflectivity is very small. Further when the number of grids is kept constant and the grating length is increased the Bragg wavelengthincreases. The effect of this increase in grating length on reflectivity is a very small. In our model, the effectiveness of the grating in extracting the Braggs wavelength is nearly 100%.

Keywords- Fiber Bragg Grating,Bragg Wavelength, Reflection, Number of Grids,Grating length, Pitch

1. INTRODUCTION

With the rapid growth of the Internet, capacity requirement is increasing day by day. This increase in the requirement of capacity can be easily met by the existing optical fiber communication technology.The transmission properties of an optical wave-guide are dictated by its structural characteristics, which have a major effect in determining how an optical signal is affected as it propagates along the fiber[1].Light propagation occurs in the guiding region of waveguide on principle of total internal reflection at the material interfaces. For this to occur, the guiding region must have a refractive index of greater value than the materials surrounding it.

1.1 Fiber Bragg Grating

The Fiber Bragg Grating (FBG) was initially demonstrated by Ken Hill., K.O. [2].FBG is a periodic perturbation of the refractive index along the fiber length in the fiber’s core which is formed by exposure of the core to an intense optical interference pattern.Germanium, a dopant used in many optical fiber cores, is photosensitive to Ultraviolet (UV) light. A grating is a selective wavelength filter in the core of an optical fiber. It is made byexposing a section of the fiber to UV light through aphase mask. An interference pattern of maxima andminima is formed causing a permanent periodic changeto the index of the core. A small amount of light isreflected at each index variation. At the "center wavelength”or “Bragg wavelength,” all the reflections addcoherently. The grating reflects light in a narrow wavelength range, centred at the so-called Bragg wavelength.

1.1.1 Grating Fabrication Technique

Many techniques have been developed for the fabrication of FBG i.e. transverse holographic(G.Meltz, 1989), phase mask (K.O.Hill, 1993) and point-by-point techniques [3].When UV light radiates on an optical fiber, the refractive index is changed permanently; this effect is known as ‘photosensitivity.’ Out of these three, the phase mask is the most common technique due to its simple manufacturing process, great flexibility and high performance.

Transverse holographic technique: The light from an UV source is split into two beams that are brought together so that they intersect; the intersecting light beams form an interference pattern that is focused using cylindrical lenses on to the core of optical fiber [3].The fiber cladding is transparent to UV light, whereas the core absorbs the light strongly. Due to this light beam the core is irradiated from the side, thus giving rise to its name transverse holographic techniques. The holographic technique for grating fabrication has two principal advantages.

- Bragg gratings could be photo imprinted in the core without removing the glass cladding.
- The period of the induced grating depends on the angle between the two interfering coherent UV light beams.

Phase Mask Technique: In this technique the phase mask is placed between the UV light source and the optical fiber. The shadow of the phase mask then determines the grating structure based on the transmitted intensity of light striking the fiber [4] & [5].

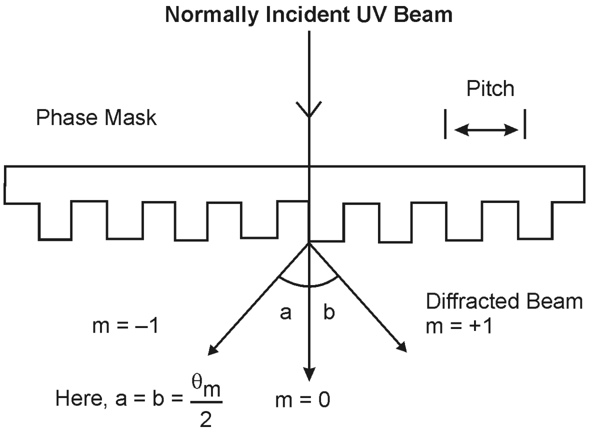


FIGURE1: Phase mask fabrication techniques of Grating

Point-by-Point technique: In this technique each index perturbation is written point by point. Here, the laser has a narrow beam that is equal to the grating period. This method of FBG inscription deep gratings have been written in a range of optical fibers at arbitrary wavelengths. It can be used to write gratings with periods of approximately 1 μm and above in a range of optical fibers [5]. In point by point technique,a step change of refractive index is induced along the core of the fiber at a time. A single pulse of the UV light passes through a mask to the core of the fiber containing a slit and thus the refractive index (n) of the corresponding core section increases locally. The fiber is then translated through a distance corresponding to the gratingpitch (Λ) in parallel direction to the fiber axis, this process is repeated to form the grating structure in the fiber core.

1.1.2 Grating Structure

The structure of the FBG can vary via the refractive index, or the grating period. The grating period can be either uniform or graded or localized and distributed in a superstructure. The refractive index profile in

FBG can be uniform or apodized, and the refractive index offset is positive or zero. There are six common structures for FBGs;

- Uniform positive-only index change,
- Gaussian apodized,
- Raised-cosine apodized,
- Chirped,
- Discrete phase shift, and
- Superstructure.

1.1.3 Grating Principle

When light passes through the FBG, the narrowband spectral component at the Bragg wavelength is reflected by the FBG. The Bragg wavelength is given by the Equation (1)[6].

$$\lambda_B = 2n_{eff}\Lambda \tag{1}$$

Where n_{eff} and Λ are the effective refractive index of the fiber and the pitch of the grating respectively.Parameters of FBG,such as period of refractive index perturbation, magnitude of refractive index, grating length and numbers of grids, give optical properties of FBG.

2. LITERATURE SURVEY

M.S. Ab-Rahman, *et al*. [7],investigated the effect of the refractive index of cladding $(n_{cld})$to the Bragg wavelength and reflectivity of the grating.They found that the effect of the η_{cld} was not linear. The Bragg wavelength shifted periodically with the change of n_{cld}.The power also varied in a quadratic manner with a change of η_{cld}.D.W.Huang,*et al.* [8], worked on reflectivity-tunable FBG reflector with acoustically excited transverse vibration of the fiber.They observed that when the transverse vibration induced the coupling between the core and cladding, the Bragg reflectivity varied from its original value to zero. With this technique, they varied the Bragg reflectivity after a fiber grating was fabricated.C.Caucheteur,*etal.*[9], investigated the polarization properties of Bragg grating. They concluded that FBGs prepared by high-intensity laser pulse were characterized by high value polarization-dependent loss (PDL) and differential group delay(DGD).F.Z.Zhang,*et al.* [10],examined the effect of the zeroth-order diffraction of the phase masks on FBG in polymer optical fiber by observing and analysing the micrographs of the grating. When the strain was larger than 2%, the viscoelasticity of the polymer fiber was noticed. The 60 nm Bragg wavelength shift was observed when they investigated the strain response by stretching the polymer optical fiber up to 6.5%.of the polymer optical fiber.B.A.Tahir, *et al.*[11], described the FBG sensing system for strain measurement. They calculated the reflectivity by keeping the grating length constant and varying the index modulation amplitude of the Grating.In their model, the average reflectivity was 96% and negligible change in reflectivity was observed by variation in index modulation.Also, if applied strain was uniform then Bragg wavelength (λ_B) shift occurred without modification of initial spectrum shape. Good linear response was observed between applied strain and Bragg wavelength shift.F.Zeng,*et al.* [12], proposed an approach to implement optical microwave filters using an FBGS with identical reflectivity. The spectrum profile of the broadband light source can be controlled using an optical filter, which could be used to control the filter coefficients to suppress the filters side-lobes.M.ANDO,*et al.*[13], 2004, investigated the dependence of FBG characteristics on grating length. They concluded that under the standard FBG fabrication condition, the exposure time of the FBG to excimer laser irradiation for a given transmission time was inversely proportional to the length of the grating. During their investigation, they fixed the value of amplitude of refractive index modulation of the grating even when the length was varied.S.Ugale,*et al.*[14], found that the reflectivity increases with increase in grating length as well as index difference.

3. ANALYTICAL MODEL

In this paper, we have investigated the effect of number of grids and grating length which will be useful in designing the wavelength splitter with the help of FBG. The Analytical model has been proposed for the reflectivity of grating which is given by Equation (2).

$$R = \frac{\sinh^2\left[kl\sqrt{1-\left(\frac{\delta}{k}\right)^2}\right]}{\cosh^2\left[kl\sqrt{1-\left(\frac{\delta}{k}\right)^2}\right]-\left(\frac{\delta}{k}\right)^2} \qquad (2)$$

Where ‘l’ is the length of the grating, ‘k’is the coupling coefficient;δ is detuning factor and $\left(\frac{\delta}{k}\right)$is the detuning ratio. The detuning parameter for FBG of period is $\delta = s - \left(\frac{\pi}{\Lambda}\right)$, $\Lambda$is known as pitch or grating period as in Equation (3).

$$\Lambda = \frac{1}{N} \qquad (3)$$

N is number of grids or number of grating periods. Pitch of grating also depends upon the value of effective refractive index and Brag wavelength as shown in Equation (4).

$$\Lambda = \frac{\lambda_B}{2n_{eff}} \qquad (4)$$

For sinusoidal variation in index perturbation, the coupling co-efficient for 1st order Bragg grating is $k = \frac{\pi\eta}{\lambda_B}$ where η is overlap integral between forward and reverse propagating mode. *V* is normalized frequency as given in Equation (5).

$$V = \frac{2\pi a n_{eff}\sqrt{2\Delta}}{\lambda_i} \qquad (5)$$

Where

- n_{eff} is effective refractive index
- a is radius of core
- Δ is index difference

4. SIMULATION,RESULT & DISCUSSION

The work was carried out on Software MATLAB 7.2 of Mathworks. The analytical model thus constructed has investigated with variation of some of the design parameters of FBG. We have fixed some of the parameters i.e. index difference between core and cladding, radius of core and index amplitude of the grating. As per Equation (5), the pitch of grating is inversely proportional to the number of grids and directly proportional to grating length. Moreover, the number of grids also affects the reflectivity of grating. Table1.shows the relation between the number of grids and Braggswavelength. Moreover, with the increase in number of grids, there is a negligible increase in reflectivity.

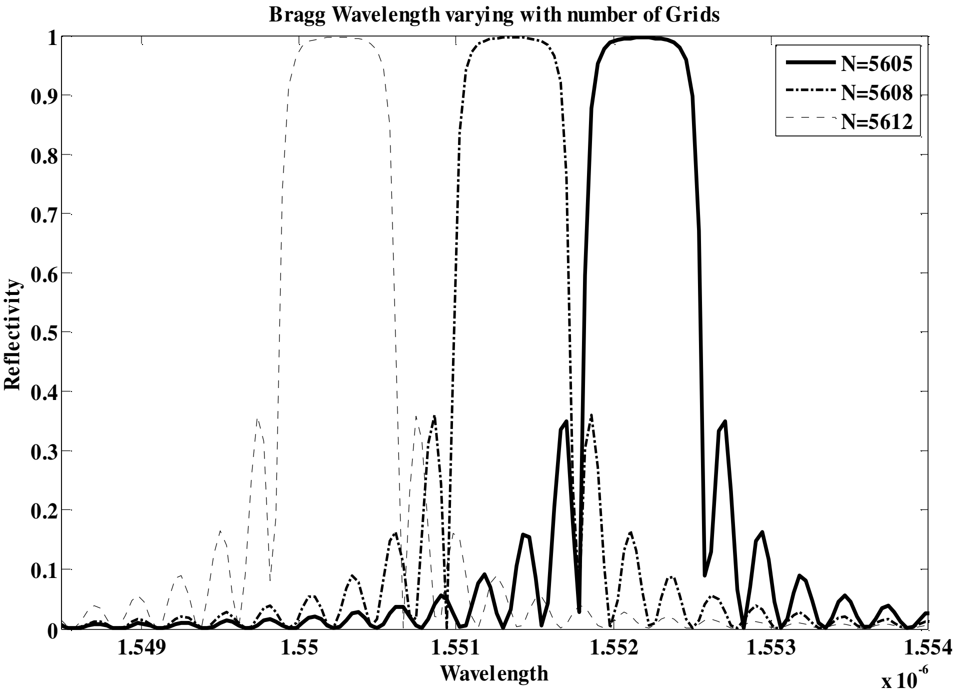


FIGURE 2: Bragg wavelength varying with number of grids of Grating N=5605, 5608 and 5612

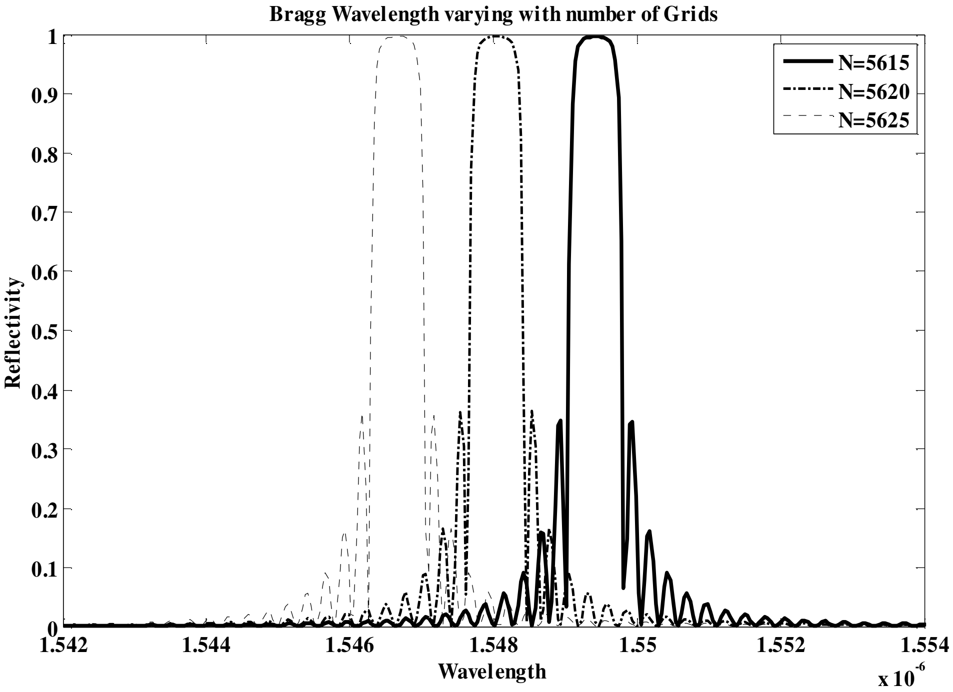


FIGURE 3: Bragg wavelength varying with number of grids of Grating N=5615, 5620 and 5625

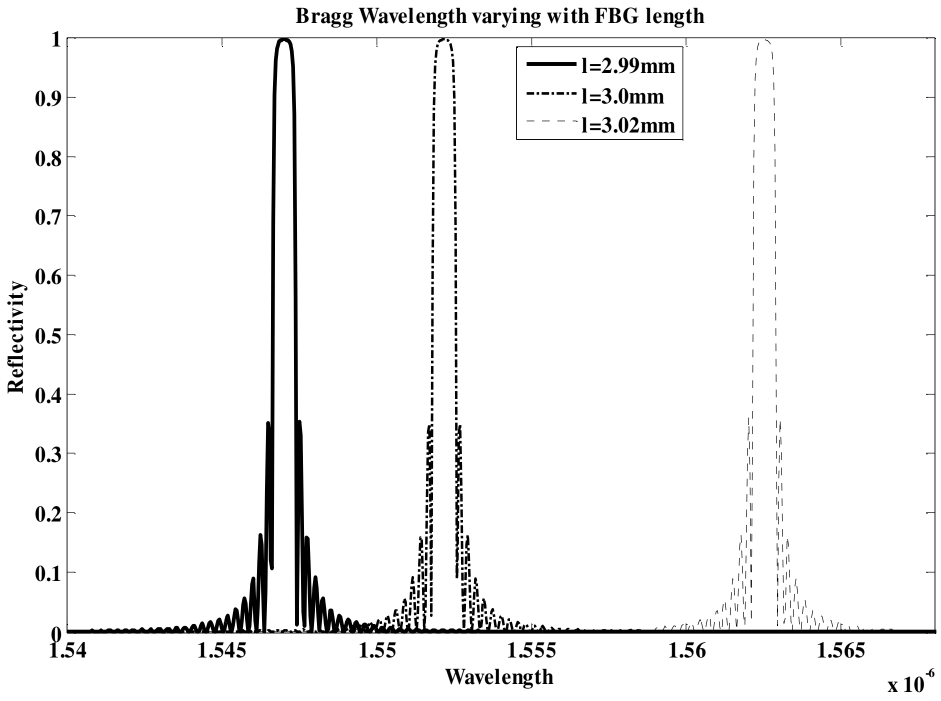


FIGURE 4: Bragg wavelength varying with Grating length l=2.99mm,3.00mm and 3.02mm

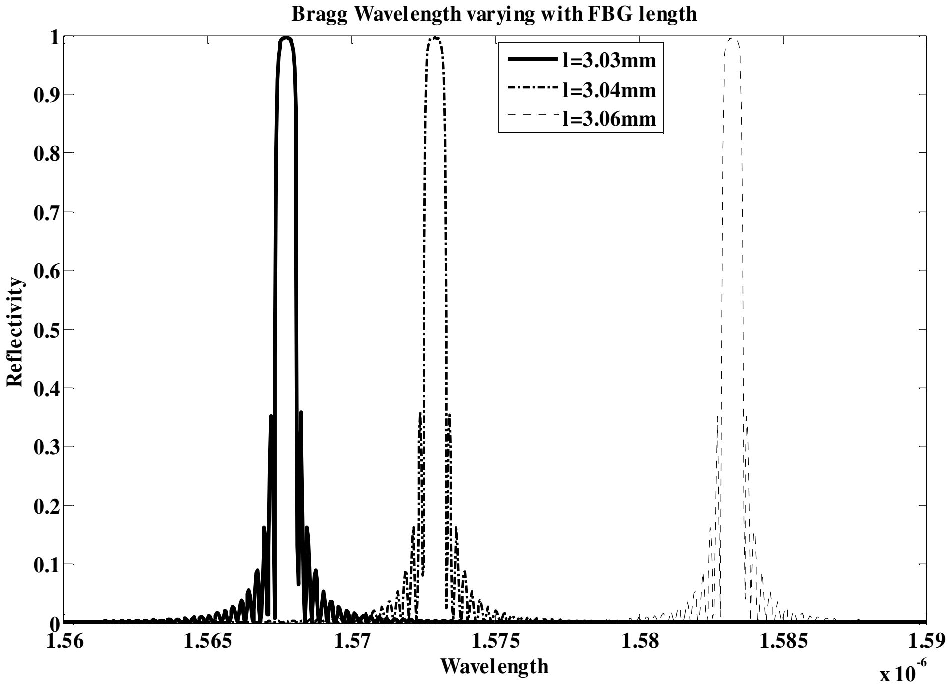


FIGURE 5: Bragg wavelength varying with Grating length l=3.03mm, 3.04mm and 3.06mm

The results shown in Figure 2 and Figure 3depict the reflectivity andBragg wavelength variation with the number of grids. It hasbeen shown in Figure 2 also, that the Bragg wavelength varies with the number of grids and it decreases with the increase in number of grids. Bragg wavelength was calculated to be 1552nm for 5605 number of grids, when grating length was fixed at 3mm. It was found that the Braggs wavelengthdecreased with the increase in number of grids. Itdecreased to 1551nm and 1550nm when the effective number of grids was increased to 5608 and 5612.For a fix value of grating, there is no change in Reflectivity with the increase in the number of grids.In Figure 3 the results show thatBragg wavelength of the grating was calculated to be 1549 for the number of grids as 5615, which was decreased to 1548, 1547, when the number of grids of grating was increased to 5620 and 5625 respectively. Moreover, with the increase in the number of grids, there was a little increase in the reflectivity of FBG. The reflectivity was increased to 99.65% with 5650 grids when thegrating length was 3mm and to 99.61% for number of grids up to 5612.

N(Number of Grids)	Bragg Wavelength(nm)	Reflectivity (%)
5605	1552	99.61
5608	1551	99.61
5612	1550	99.61
5615	1549	99.62
5620	1548	99.62
5625	1547	99.63
5640	1545	99.63
5650	1540	99.65

TABLE 1: Characteristics of FBG with the Variation of number of Grids

Now, on same model of FBG splitter, we have fixed the number of grids (N) and studied the effect of varyinggrating lengthon Bragg wavelength and reflectivity. As per Equation (3),$\Lambda = l/N$ the pitch of grating is directly to the gratinglength. In this section, the number of Grids (N) is fixed at 5605 and the grating length is varied to observe the effect on Bragg wavelength and reflectivity of the grating.

Grating length(mm)	Bragg Wavelength(nm)	Reflectivity (%)
2.99	1547	99.6
3.00	1552	99.6
3.02	1563	99.59
3.03	1568	99.58
3.04	1573	99.58
3.05	1583	99.56
3.06	1599	99.52
3.07	1630	99.45

TABLE 2: Characteristics of FBG with the Variation of Grating length

In Figure 5, the Bragg wavelength calculated was 1568 nm for 3.03mmgratinglength. With increase in the grating length, the pitch of grating increased but the Bragg wavelength of the FBG decreased and at the same time the reflectivity of the FBG also decreased.For the grating length of 3.04mm and 3.06mm, the Bragg wavelength noted was 1573 nm and 1583nm respectively. The Table 2 also showed that for increase of 0.02mm in the grating length, the Bragg wavelength shifted from 1573nm to 1583nm. The reflectivity also decreased with the increase ingrating length. It was 99.60 % for 2.99mm grating length and 99.45% for 3.15mm grating length.

The experimental results show the effect of number of grids, the length of the grating on the Bragg wavelength and reflectivity ofFBG. It is clear that the pitch of grating is directly proportional to the grating length and inversely proportional to number of grids. When the grating length is fixed and the number of grids is increased, the Bragg wavelength decreases resulting in increased reflectivity. This increased reflectivity is very small. Further when the number of grids is kept constant and the grating length is increased the Bragg wavelength increases. The effect of this increase in grating length on reflectivity is a very small. The effectiveness of the grating in extracting the Braggs wavelength is nearly 100%.

6. CONCLUSION

In our work, we have analysed the effect of number of grids and grating length of FBG onreflectivity and Bragg wavelength by keeping other parameters constant. The pitch of grating is directly proportional to grating length and inversely proportional to number of grids.On increasingthenumber of grids, keeping the grating length as fixed, the Bragg wavelength decreases and at the same time, the reflectivity increases. Thereflectivity increasesby 0.02% with increase in the number of grids by 25 and at the same time, Bragg wavelength shifted by 7nm.Also, when the grating length is varied by 0.02mm, keeping the number of grid constant,the Bragg wavelength shifts by 10nm and reflectivity decreases by 0.02%. The effectiveness of the grating in extracting the Braggs wavelength is nearly 100%.

7. REFERENCES

[1] G.Keiser, “Optical Fiber communication”,Second Edition,Singapore,McGraw-Hill series in Electrical Engineering, 1991.

[2] K.O.Hill, Y.Fujii, and D.C.Johnson,“Photosensitivity in optical fiber waveguides:Application to reflection filter fabrication”,*Applied Physics Letters*,VOL.32,647-649, 1978.

[3] Z.Zhou, T.W.Graver, L.Hsu and J.Ou,“Techniques of Advanced FBG sensor:fabrication,demodulation,encapsulation and their application in the structural health monitoring of Bridges”,*Pacific Science Review*,VOL.5,116-121,2003.

[4] K.O.Hill,and G. Meltz, “Fiber Bragg Grating Technology Fundamentals and Overview”,*Journal of Light wave technology*,VOL.15, NO.8, 1997.

[5] G.D.Marshall, M.Arms and M.J. Withford,“Point by point femtosecond laser inscription of fiber and waveguide Bragg grating for photonic device fabrication”,*Proceeding of the 2nd Pacific international conference on Application of Lasers and optics*,2006.

[6] Z.Zhou and J.Ou,“Techniques of temperature compensation for FBG strain sensors used in long-term structural monitoring”,*Proceeding of Asian pacific fundamental problems of Opto- and Microelectronics*(APCOM 2004) Russia,465-471, 2004.

[7] M.S.Ab-Rahman and M.Fauzi, “Effect of Refractive index of cladding on Bragg wavelength and reflection –the Application”,*Australian Journal of Basic and Applied Science*,VOL.3, NO.3, 2876-2882, 2009.

[8] D.W.Huang,W.F.Liy,C.W.Wuand C.C.Yang,“Reflectivity Tunable fiber Bragg Grating Reflectors”,*IEEE Photonics Technology letters*,VOL.12,NO.2, 2000.

[9] C.Caucheteur, P.Megret, T.Ernst and D.N.Nikogosya,"Polarization properties of FBG inscribed byhigh intensity femtosecond 264 nm pulses",*Science Direct*,Optical Communication 271, 303-308,2007.

[10] F.Z. Zhang,C.Zhang and M.T.Xiao, "Inscription of polymer Optical Fiber Bragg grating at 962 nmand its potential in strain sensing",*IEEE photonics technology letters*, VOL.22, NO.21, 2010.

[11] B.A.Tahir, J.Ali and R.A.Rahman, "Strain measurement using FBG sensor",*American Journal ofApplied Science*(special Issue),40-48, 2005.

[12] F.Zeng and J.Yao, "All optical microwave filters using uniform FBG with Identical Reflectivity", *Journal of Lightwave Technology*,VOL.23,NO.3, 2005.

[13] M.Ando,M.Yamauchi,K.Nakayama,K.Moriyama and Y.Masuda,"Dependence of FBG Characteristics on its Length",*Japanese Journal of Applied Physics*,VOL.43,NO.7A,4234- 4235, 2004.

[14] S.Ugaleand V.Mishra,"Fiber Bragg Grating modelling, Characterization and Optimization with different index profiles",*International Journal of Engineering Science and Technology*,VOL.2, NO. 9,4463-4468,2010.

Fault Tolerant Congestion Based Algorithms in OBS Network

Hardeep Singh *hardeep_saini17@yahoo.co.in*
Research Scholar (ECE), Singhania University,
Jhunjhunu, Rajasthan-333515, INDIA

Dr. Jai Prakash *ecedinesh@rediffmail.com*
Professor in Department of Applied Physics,
Indo Global College of Engineering,
Mohali, Punjab-140109, INDIA

Dinesh Arora *ecedinesh@rediffmail.com*
Swami Devi Dyal Institute of Engineering & Technology,
Barwala, Haryana-134009, INDIA

Dr.Amit Wason *wasonamit13@gmail.comProfessor*
in Department of ECE,
Rayat & Bahra Institute of Engineering & Bio-Technology,
Mohali, Punjab-140104, INDIA

Abstract

In Optical Burst Switched networks, each light path carry huge amount of traffic, path failures may damage the user application. Hence fault-tolerance becomes an important issue on these networks. Blocking probability is a key index of quality of service in Optical Burst Switched (OBS) network. The Erlang formula has been used extensively in the traffic engineering of optical communication to calculate the blocking probability. The paper revisits burst contention resolution problems in OBS networks. When the network is overloaded, no contention resolution scheme would effectively avoid the collision and cause blocking.It is important to first decide, a good routing algorithm and then to choose a wavelength assignment scheme. In this paper we have developed two algorithms, Fault Tolerant Optimized Blocking Algorithm (FTOBA) and Fault Tolerant Least Congestion Algorithm (FTLCA) and then compare the performance of these algorithms on the basis of blocking probability. These algorithms are based upon the congestion on path in OBS network and based on the simulation results, we shows that the reliable and fault tolerant routing algorithms reduces the blocking probability.

Keywords: Optical Burst Switching Network; Congestion; Contention Resolution; Blocking Probability; Erlang Formula.

1.INTRODUCTION

Optical Burst Switching(OBS) is a technique to support bursty traffic over wavelength-Division-Multiplexed(WDM) networks[1].WDM offers the capability to handle the increasing demand of network traffic [2].Today up to several *Tbits/sec* traffic can be carried by the optical link over long distance.With the introduction of WDM in optical communication, the discrepancy between optical transmission capacity and electronic switching capability increases [3].An OBS network is a collection of interconnected OBS nodes.An ingress OBS node assembles packets from local access network, for example,Internet Protocol(IP) packets,into burst and sends out a corresponding control packets (CP) for each data burst. The optical networks have the capacity to carry terra bytes of data per second through each node. The edge routers feed data into these networks. The basic diagram for WDM and network nodes is shown in Figure1.The data is typically carried over 10 Gbps wavelength channels. Once a channel is setup between source and destination, it can only carry packet traffic between selected source-destination pairs.

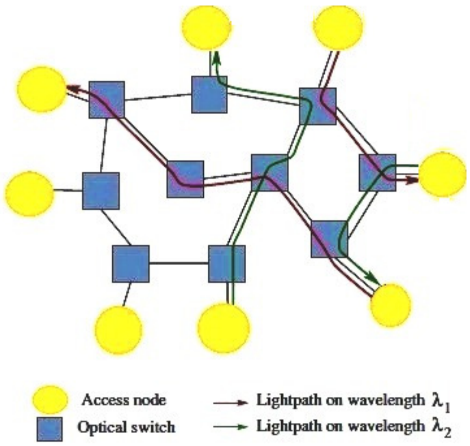


FIGURE 1: A wavelength-routed optical WDM network with lightpath connections

2. ROUTING STRATEGIES

A number of routing strategies have been proposed for OBS networks by researchers. These strategies can be classified as alternative, multi-path or single –path routing strategies. In general, a routing algorithm can be classified as static or adaptive. A static routing algorithm is one in which the routing procedure does not vary with time. But adaptive routing algorithms use network state information at the time of connection establishment [4].Fixed routing is widely used static routing technique in which every s-d pair is assigned a single path. A call is blocked if its associated path is not available. In alternative routing each s-d pair is assigned a set of paths. In alternative routing, when the burst contention occurs, deflective mechanisms react to it and re-routes a blocked burst from the primary to alternative route. Alternative routing in OBS network can be either adaptive or non-adaptive. In adaptive alternative routing, a strategy is proactive calculation of alternative paths as well as their dynamic selection. The calculation of alternative paths can be performed in an optimized way. In non-adaptive both primary and alternative routing paths are fixed (static) and in most cases calculated with Dijkstra algorithm. A number of alternative paths can be given from a node to the destination. The aim of multi-path routing strategies is to distributing the traffic over a number of routing paths in order to reduce the network congestion.The path selection can be either according to a given probability or according to congestion on each path we can also say according to path congestion rank. Both adaptive (dynamic) or non-adaptive (static) strategies are considered for single path routing in OBS networks.

3. FAULT TOLERANT ALGORITHMS IN WDM/OBS NETWORKS

The ability of network to with-stand failures is called as fault-tolerance. The failures in OBS networks can be classified into two categories i.e. wavelength level and fiber level failures [5].The wavelength level failure impacts the quality of transmission of each individual lightpath and fiber level failure affect all the light-paths on an individual fiber. The fault tolerance schemes can be classified into path protection and path restoration. In path protection, backup resources are reserved during connection setup and primary and backup paths are computed before a failure occurs. In path restoration, the source and destination nodes of each connection traversing the failed link participate in distributed dynamically discover an end-to-end backup route. If no routes are available for broken connection, the connection is dropped.

Random Packet Assembly Admission Control (RPAAC) algorithm is a traffic engineering mechanism which monitors the network congestion and proactively drops incoming packets at ingress nodes before they may actually become harmful to the network [6].This algorithm is performed via adjusting the value of the packet selection probability, which regulates the size of bursts and percentage of proactively dropped traffic, on attempts to prevent or optimize network congestion. Reliable and fault tolerant routing (RFTR) algorithm was proposed by G. Ramesh *et al*[5].In order to establish the primary path, this algorithm uses the concept of load balancing. For source-destination pairs, finding a route of light paths for the network with least congestion is called as load balancing. The traffic is routed over the lightly loaded links.Algorithm for solving the Dynamic Routing and Wavelength Assignment (DRWA) problem in wavelength routed optical networks was proposed by D.Bisbal *et al*[7].This algorithm provides the low call blocking probability and also employ very short computation time.The blocking performance of DRWA algorithm is measured in terms of the mean call blocking probability. A review of DRWA algorithm can be found in Zang *et al*.[8].In response to source –destination connection request, a route is chosen from pre-calculated set, and then a wavelength is assigned to it following a wavelength assignment policy. If the selected route cannot be established on any wavelength, a new route is selected. If none of the routes in the set has an available wavelength, then call is blocked. Assigning wavelength to different paths in a manner that minimizes the number of wavelengths used under the wavelength-continuity constraint.There are various types of wavelength assignment heuristicsthat attempt to reduce the blocking probability i.e. First-Fit, Random, Least-Used, Most-Used, Min-Product, Least-Loaded[9], Max-Sum, Relative Capacity Loss, Wavelength Reservation and Protecting Threshold.Zing *et al*[8],introduced a new wavelength assignment algorithm called Distributed Relative Capacity Loss(DRCL),which is based on Relative Capacity Loss(RCL).They compared the performance of DRCL with RCL(with fixed routing) and FF(with fixed and adaptive routing) in terms of blocking probability and concluded that it perform better than FF(with adaptive routing) in the reasonable region.

Z.Jing *et al*[1]investigated a novel fault-tolerant node architecture using a resilient buffer(R-buffer).In their model buffer is attached for each outgoing link.The outgoing data burst will be tappedand stored in a buffer for short period of time (T_s) such that the bursts are expected to reach the other end of link if no failure is detected on this link during T_s. In case of link failure,burst stored in the buffer will be sent out via the backup routes. The data stored in buffer will be discarded after time period T_s so that the space of the buffer can be reused for future use. M.Ahmed *et al*[4]present adaptive routing i.e. Adaptive Unconstrained Routing (AUR) and wavelength assignment and evaluate their performance on the basis of blocking probability. Unconstrained routing scheme consider all paths between the s-d pairs. This is accomplished by executing a dynamic shortest path algorithm with link cost obtained from network state information at the time of connection request. This scheme is called AUR.They examined the performance of AUR in conjunction with different wavelength assignment schemes i.e. Random, Least-Used (SPREAD) and Most-Used (PACK) on the basis of blocking probability as a function of load per s-d pair.The Most-Used scheme has best performance, followed by Random and then by Least-Used. A new class of alternate routing was also proposed by H.Hiroaki *et al*[10] to achieve better performance of the network with different numbers of hop counts. Normally, the connection with shorter hop counts is likely accepted while the one with more hops encounters more call blocking. In optical network without wavelength conversion,the performance is degraded as the number of hop counts is increased[11].In alternate routing method with limited trunk reservation[10],connections with more hops are provided more alternate routed in proportion to the number of hop counts.

L.Kungmeng *et al*[2] also investigated on class of adaptive routing called Dynamic Wavelength Routing (DWR), in which wavelength converters are not used in the optical network. They introduced two algorithms: Least Congestion with Least Nodal-degree Routing (LCLNR) and Dynamic Two-end Wavelength Routing (DTWR) algorithms. Their objective is to maximize the wavelength utilization and reduce the blocking probability in the optical network. In their algorithms a route is determined by calculating their cost or weight function. In LCLNR algorithm, avoid routing dynamic traffic through congested links, thus reducing blocking probability. They concluded that number of connected calls by LCLNR algorithm is slightly decreased when the traffic load is increased.But in DTWR, number of call connected is increased with higher traffic load.Their results show that DWR does not increase the blocking probability when DTWR selects longer routes.X.Masip-Bruin *et al*[9]proposed a routing scheme in which the routes are determined based upon the twin criteria of minimizing the number of hops and balancing the network load, resulting in the reduction of both network congestion and blocking probability. Their proposed Minimum Coincidence Routing (MCR) algorithm was based on either the hop length or wavelength availability. The MCR algorithm exploits the concept of minimum coincidence between paths to balance the traffic load, thereby reducing the network congestion. This algorithm computes the end-to-end paths by considering the routes that have fewest shared links and minimum hops. The research on optimization of blocking probability on OBS networks was also done by Z.Rosberg *et al*[12].They introduced a reduced load fixed point approximation model to evaluate blocking probability. Also they compare the route blocking probabilities using Just-Enough-Time (JET), Segmentation, Least Remaining

Hop-count (LRHF) and Most Traversed Hop-count (MTHF) policies. In MTHF, bursts that have traversed the most number of hops have the highest priority. MTHF improves the blocking probabilities of long routes provided that their prefixes do not collide or equal priority routes.LRHF has an effect similar to MTHF, but on short routes. So LRHF and MTHF priority can be used for service differentiation between long and short routes.

3. PROPOSED FAULT TOLERANT ALGORITHMS

In this paper we propose two algorithms Fault Tolerant Optimized Blocking Algorithm (FTOBA) and Fault Tolerant Least Congestion Algorithm (FTLCA). The objective of our algorithms is to minimize blocking probability. Our analytical models are designed under the following assumptions:

- A call connection request of s-d pair is based on a Poisson distribution with arrival rate ⍰.The average service holding time is exponentially distributed with mean 1/μ. The offered congestion (Erlangs) per node is $\rho = \eta/\mu$.
- Each station has array of transmitters and receivers, where λ is the wavelength carried by the fiber.
- The optical network is set of nodes interconnected by single-fiber links.
- Each fiber-link is bi-directional and each link has λ wavelength channel.
- No Queueing of connection request. If a connection is blocked, it immediately discarded.
- Link loads are independent.
- We have assumed dynamic path allocation in this paper.

To calculate the blocking probability we will use the Erlang-b formula as in equation (1). The Erlang formula has been used extensively in the traffic engineering of optical communication. Erlang is defined as dimensionless unit of traffic intensity. It is dependent on observation time. The maximum that a facility can be in use is 100% of the time. If the observation time is 10 minutes, and facility is in use for the full time, then that is 1 Erlang.

$$P_{b\,(C,\lambda)} = \frac{\frac{C^{\lambda}}{\lambda!}}{\sum_{i=0}^{\lambda}\frac{C^{i}}{i!}} \quad (1)$$

Where$P_{b\,(C,\lambda)}$ is the Blocking Probability for C congestion and λwavelength.

Fault Tolerant Least Congestion Algorithm (FTLCA)

The FTLCA algorithm is basically on congestion on paths between the s-d pairs. The blocking probability mostly occurs due unbalancing of congestion on paths between s-d paths. First algorithmselects the s-d pair, and then calculates the number of available paths between the selected s-d pair. After the calculation of number of available paths, checking of congestion on each path will be done. Then algorithm sorts the values of congestion in increasing order.Normally we assume that the path with minimum congestion will offer least blocking probability.On this criterion algorithm selects the first path in order of congestion. After the selection of path, the checking of path for fault that leads to blocking probability. If fault exitsthen select the second path in order of congestion, otherwise call will be established on selected path.

Fault Tolerant Optimized Blocking Algorithm (FTOBA)

Similar to FTLCA, the FTOBA is also congestion based but in this algorithm blocking probability on each will be calculated. Firstalgorithm selects the s-d pair, and then calculates the number of available paths between the selected s-d pair. After the calculation of number of available paths, checking of congestion on each path will be done.After calculating the blocking probability for each path, arrange the paths in increasing order of blocking probability. The first path will be selected in order of blocking probabilities. Then algorithm checks the call which isblocked or not, because least blocking, does not mean that there is no fault when we chose the path with least blocking probability at the time of call establishment the call may be blocked due to any fault. If call is blocked then select the next path in the order of blocking probability. If call not blocked then, call is established.

Then flow chart shown in Figure 2 and Figure 3 more illustrate the mechanism of FTOBA and FTLCA algorithms.

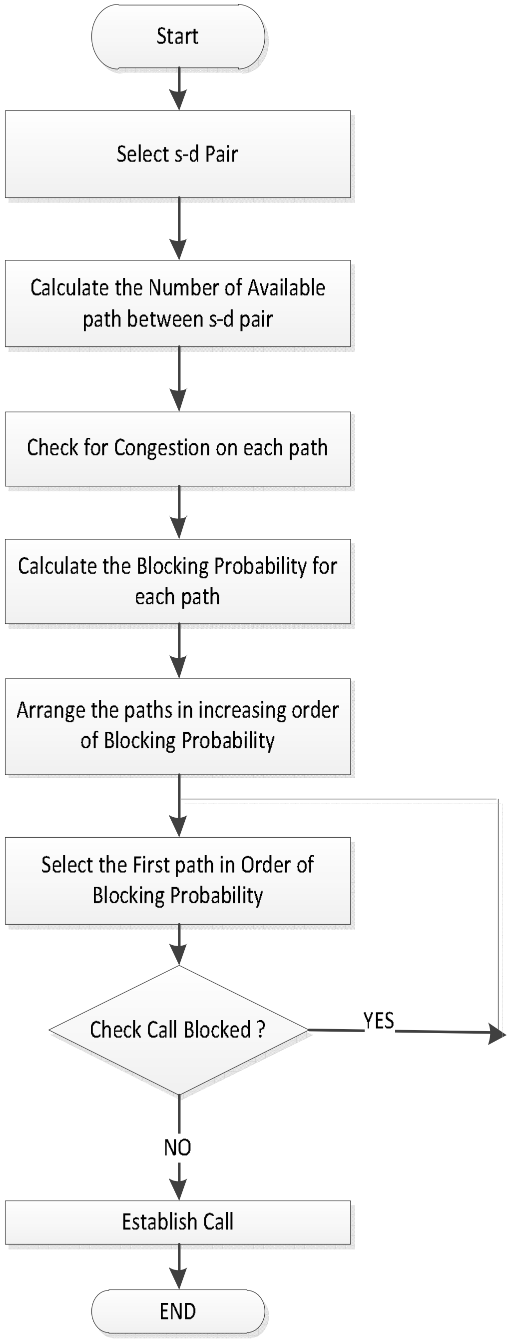


FIGURE 2: Fault Tolerant Optimized Blocking Algorithm(FTOBA)

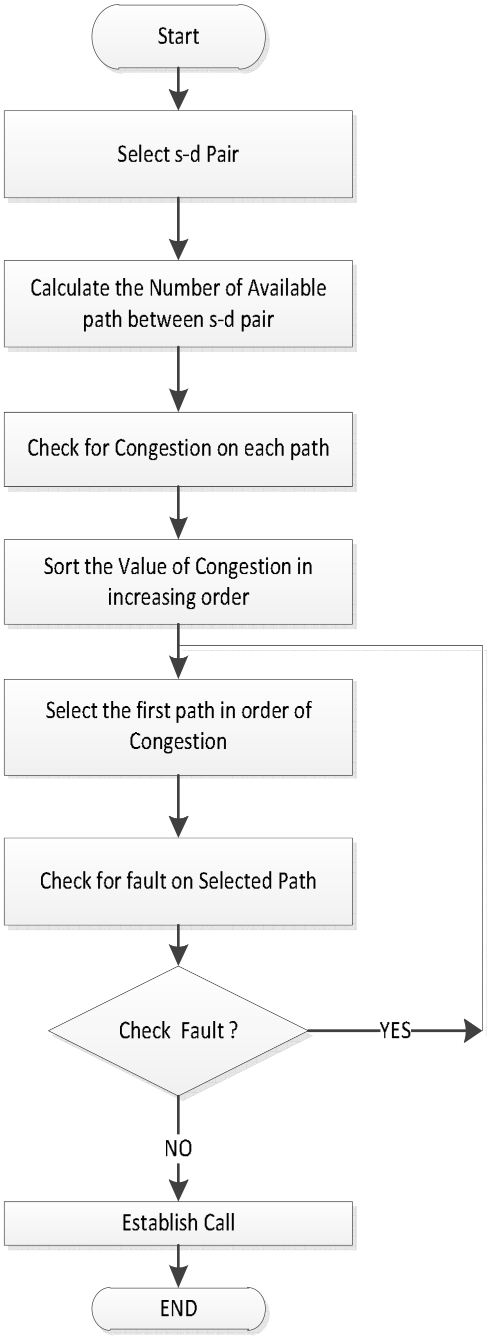


Figure3: Fault Tolerant Least Congestion Algorithm (FTLCA)

4. RESULTS AND DISCUSSION

The simulation is carried out on simulationsoftware MATLAB 7.5 of Mathwork. Both the algorithms i.e. FTOBA and FTLCA are compared depending upon wavelengths, congestion and number of paths. We have fixed the value of paths P=25 and congestion in Erlangs, number of wavelengths is varied. The dynamic path allocation has been adopted for these algorithms. Due to dynamic routing algorithm the congestion on every path will be in random at different time of call establishment.

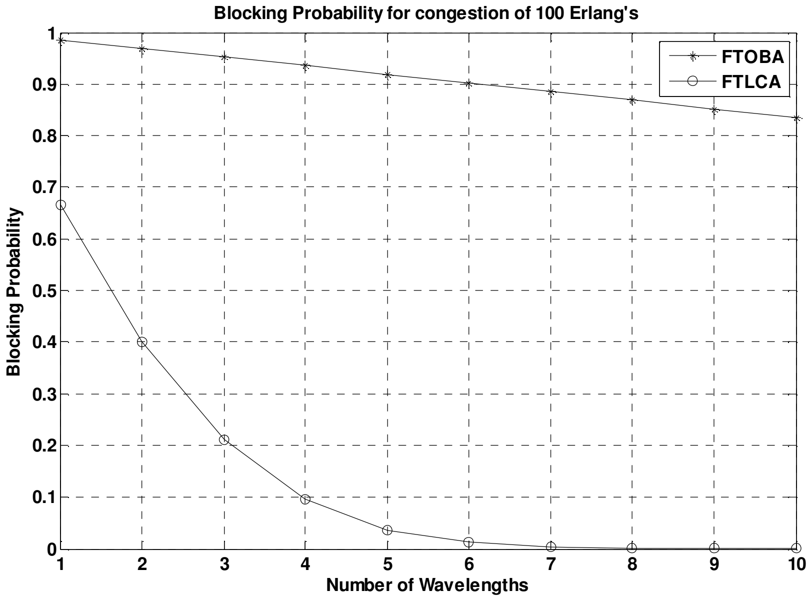


FIGURE 4: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 100 Erlangs and Wavelengths is 10

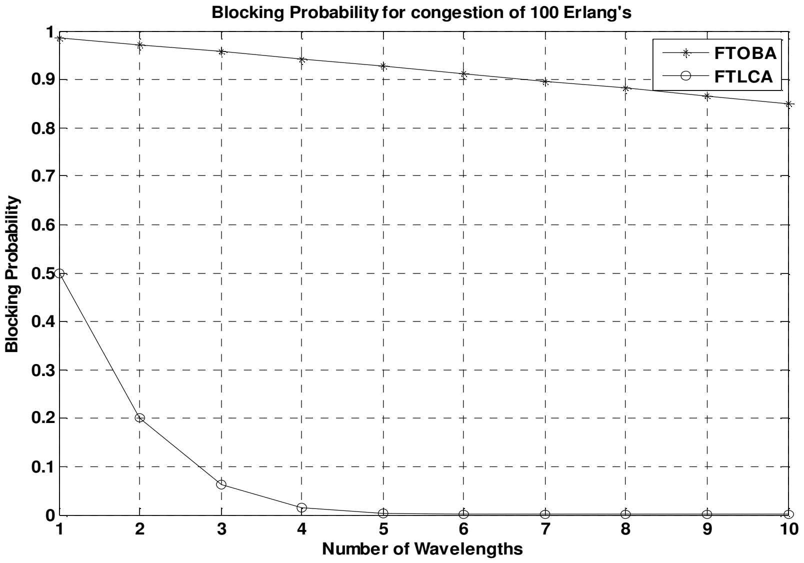


FIGURE 5: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 100 Erlangs and Wavelengths is 10

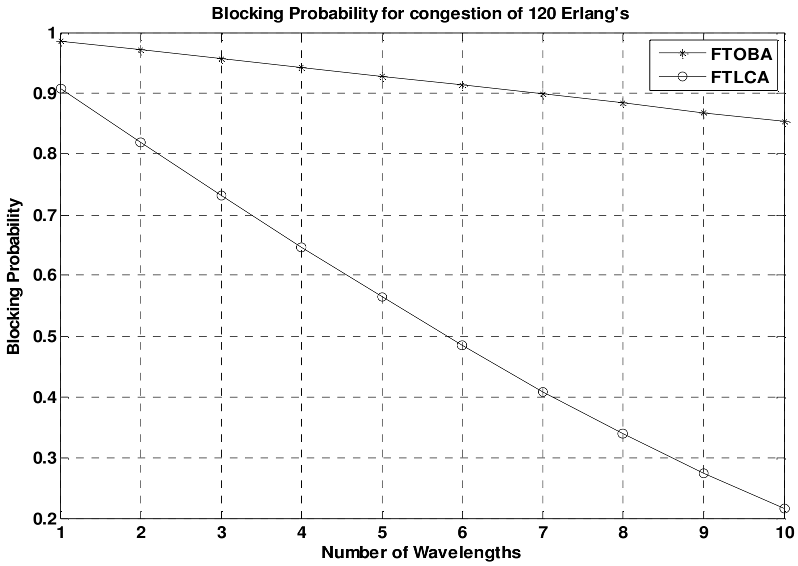


FIGURE 6: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 120 Erlangs and Wavelengths is 10

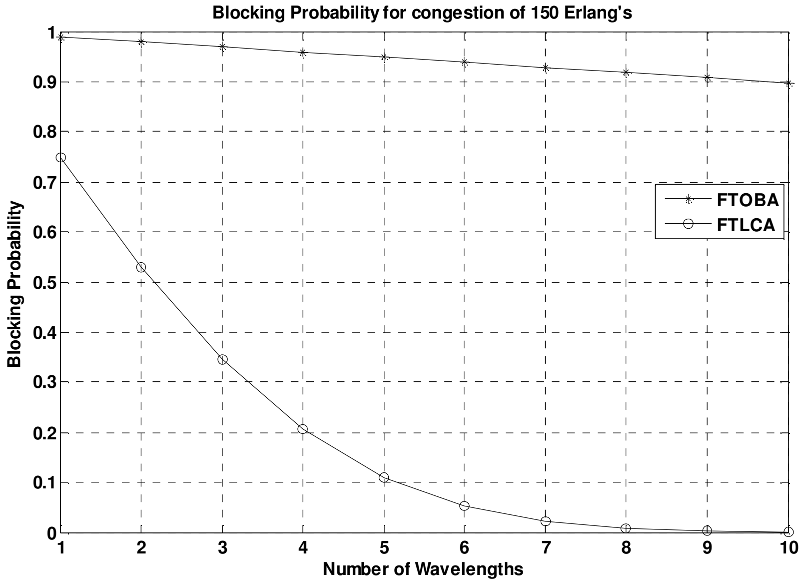


FIGURE 7: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 150 Erlangs and Wavelengths is 10

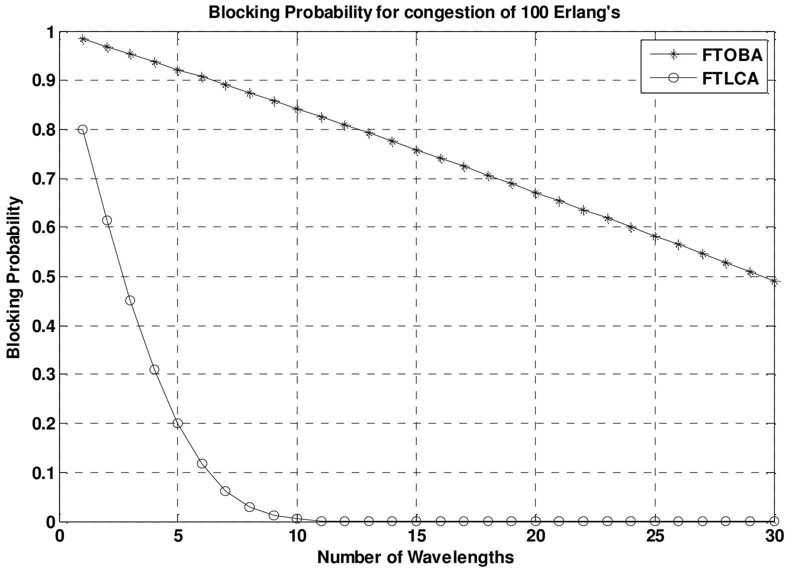


FIGURE 8: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 100 Erlangs and Wavelengths is 30

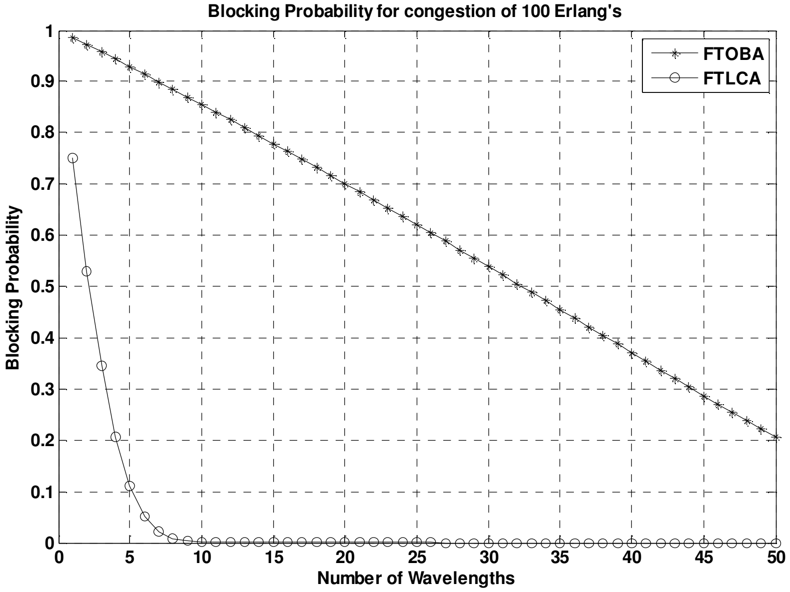


FIGURE 9: Comparison of FTOBA & FTLCA algorithms on the basis of Blocking Probability for Congestion of 100 Erlangs and Wavelengths is 50

The effect of random value of congestion on each path while using FTOBA and FTLCA algorithms is shown in Figure 4 and Figure 5.The Blocking probability is different for same number of wavelength in both the algorithms. But the effect of dynamic routing or random value of congestion is more in FTLCA algorithm as compared to FTOBA algorithm. The blocking probability with FTLCA algorithm is almost zero for'7' number of available wavelength and it remains zero for up to '10' number of wavelengths as shown in Figure 4.But in FTOBA blocking probability is lies between 80%-90% for these number of wavelengths.The blocking probability is decreased with increase in number of wavelengths in both the algorithms. The Figure 4& Figure 5 shows the performance of both the algorithms when congestion in Erlangs on each path is 100 and there are ‘10’ number of wavelengths.

Next we increase the maximum value of limit congestion on each path i.e. 120,150 Erlangs and number of wavelength as ‘10’, results are shown in Figure6 and Figure 7 respectively.It is observed that the increase in congestion does not affect the FTOBA algorithm.The blocking probability is nearly same as is in the case when maximum congestion on each path is 100 Erlangs.But in FTLCA algorithm the blocking probability is increased with increase in congestion. If we can limit the maximum value of congestion to a particular value than these algorithms are very effective.

In second part, we limit the maximum value of congestion 100 Erlangs and increased the number of wavelengths to 30 and 50 as shown in Figure 8 and Figure 9 respectively. With the increase in number of wavelengths the blocking probability decreases. As shown in Figure 9,for FTLCA the blocking probability is zero at ‘10’ number of wavelengths and it remains zero up to ‘50’ number of wavelengths. Similarly in FTOBA algorithm the blocking probability decreases with the increase in number of wavelengths. The blocking probability is 20% at ‘50’ number of wavelength when maximum congestion on each path is 100 Erlangs.

5. CONCLUSION

In this paper, we have presented fault tolerant algorithms for the routing in optical network.We conclude that the performance of FTLCA is better than the FTOBA routing algorithm in optimizing the blocking probability to setup lightpath in network. It has been observed that the value of blocking probability is reduced with the increase in number of wavelengths. These algorithms are better than conventional algorithms as complexity of these algorithms is very less. Also these algorithms can be implemented in on-line path allocation process. If we can limit the maximum value of congestion to a particular value than these algorithms are very effective. Results have been proved that these algorithms can be used effectively in faulty OBS networks to yield better results.

6. REFERENCES

[1]. J.Zhang,L.Song and B.Mukherjee, “A Fault-Tolerant OBS Node Architecture with Resilient Buffers”, *OpticalCommunication (ECOC) 33rd European Conference and Exhibition*, 978-3-8007-3042-1, Sep.2007.

[2]. L.Kungmeng,D.Habibi,Q.V.Phung and H.N.Nguyen, “Dynamic Wavelength Routing in all-Optical Mesh Network”,*Asia-Pacific Conference on Communication*,Perth,Western Australia,177-182.0-7803-9132-2/05,IEEE 2005.

[3]. A.Goel, V.J.Gond, R.K.Sethi and A.K.Singh, “Performance Analysis of Continuous wavelength optical Burst switching networks”, *International Journal of Engineering*, Vol.3, No.6, 609-621.

[4]. A.Mokhtar and M.Azizoglu, “Adaptive Wavelength Routing in All-Optical Networks”,IEEE/ACM Transactions on Networking,Vol.6,No.2,197-206,April 1998.

[5]. G.Ramesh and S.S.Vadivelu, “A Reliable and Fault-Tolerant Routing for Optical WDM Networks”,*International Journal of Computer science and Information Security*,Vol.6,No.2,48-54,2009.

[6]. D.Callegati,F.Matera,T.Franzl and C.Munoz, “Report on Y2 activities and new integration strategy-Building the future optical network in Europe”,FP7-ICT-216863/DEIS-UNIBO/R/PU/D11.2,2009.

[7]. D.Bisbal,I.D.Miguel,F.Gonzalez,J.Blas,J.C.aguado and M.Lopez, “Dynamic Routing and wavelength Assignment in Optical Networks by Means of Genetic Algorithms”,*Photonic network Communication*, Vol.7,No.1,43-58,2004.

[8]. H.Zang, J.P.Jue and B.Mukherjee, “A Review of Routing and wavelength assignment approaches for wavelength-routed optical WDM networks”, *Optical Network magazine/SPIE*, 47-60,Jan.2000.

[9]. X.Masip-Bruin,M.German,A.Castro,E.Marin-Tordera, R.Serral-Gracia and E.grampin, “The Minimum coincidence routing approach in wavelength-routed optical WDM networks”, 978-1-4244-4550-9/09,IEEE 2009.

[10]. H.Harai,M.Murata and H.Miyahara, “Performance of Alternate Routing methods in All-optical switching networks”, 0-8186-7780-5/97,IEEE-1997.

[11]. M.Kovacevic and A.Acampora, “Benefits of Wavelength Translation in All-Optical clear- channel Networks”,*IEEE Journal on selected areas in communications*,Vol.14, No.5,868-880, Jun.1996.

[12]. Z.Rosberg,H.L.Vu,M.Zukerman and J.White, ‘‘Blocking Probabilities of optical burst switching networks based on reduced load fixed point approximations’’, 0-7803-7753-2/03,IEEE- 2003.

Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator

Farzin Piltan *SSP.ROBOTIC@yahoo.com*
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia 43400
Serdang, Selangor, Malaysia

N. Sulaiman *nasri@eng.upm.edu.my*
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia 43400
Serdang, Selangor, Malaysia

Samira Soltani *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

Samaneh Roosta *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

Atefeh Gavahian *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

In this research, an artificial chattering free adaptive fuzzy sliding mode control design and application to uncertain robotic manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Regarding to the positive points in sliding mode controller, fuzzy logic controller and adaptive method, the output has improved. Each method by adding to the previous controller has covered negative points. The main target in this research is design of model free estimator on-line sliding mode fuzzy algorithm for robot manipulator to reach an acceptable performance. Robot manipulators are highly nonlinear, and a number of parameters are uncertain, therefore design model free controller using both analytical and empirical paradigms are the main goal. Although classical sliding mode methodology has acceptable performance with known dynamic parameters such as stability and robustness but there are two important disadvantages as below: chattering phenomenon and mathematical nonlinear dynamic equivalent controller part. To solve the chattering fuzzy logic inference applied instead of dead zone function. To solve the equivalent problems in classical sliding mode controller this paper focuses on applied fuzzy logic method in classical controller. This algorithm works very well in certain environment but in uncertain or various dynamic parameters, it has slight chattering phenomenon. The system performance in sliding mode controller and fuzzy sliding mode controller are sensitive to the sliding function. Therefore, compute the optimum value of sliding function for a system is the next challenge. This problem has solved by adjusting sliding function of the adaptive method continuously in real-time. In this way, the overall system performance has improved with respect to the classical sliding mode controller. This controller solved chattering phenomenon as well as mathematical nonlinear equivalent part by applied fuzzy supervisory method in sliding mode fuzzy controller and tuning the sliding function.

Keywords: chattering phenomenon, chattering free adaptive sliding mode fuzzy control, nonlinear controller, fuzzy logic controller, sliding mode controller, mathematical nonlinear dynamic equivalent controller part.

1. INTRODUCTION

A robot system without any controllers does not to have any benefits, because controller is the main part in this sophisticated system. The main objectives to control robot manipulators are stability and robustness. Many researchers work on designing the controller for robotic manipulators in order to have the best performance. Control of any systems divided in two main groups: linear and nonlinear controller [1].

However, one of the important challenge in control algorithms is to have linear controller behavior for easy implementation of nonlinear systems but these algorithms however have some limitation such as controller working area must to be near system operating point and this adjustment is very difficult especially when the system dynamic parameters have large variations and when the system has hard nonlinearities [1]. Most of robot manipulators which work in industry are usually controlled by linear PID controllers. But the robot manipulator dynamic functions are, nonlinear with strong coupling between joints (low gear ratio), structure and unstructured uncertainty and Multi-Inputs Multi-Outputs (MIMO) which, design linear controller is very difficult especially if the velocity and acceleration of robot manipulator be high and also when the ratio between joints gear be small [2]. To eliminate above problems in physical systems most of control researcher go toward to select nonlinear robust controller.

One of the most important powerful nonlinear robust controllers is Sliding Mode Controller (SMC). Sliding mode control methodology was first proposed in the 1950 [3, 4]. This controller has been analyzed by many researchers in recent years. Many papers about the main theory of SMC are proposed such as references [1, 5, 6]. This controller has been recently used in wide range of areas such as in robotics, process control, aerospace applications and in power converters. The main reason to opt for this controller is its acceptable control performance wide range and solves some main challenging topics in control such as resistivity to the external disturbance and uncertainty. However pure sliding mode controller has some disadvantages. First, chattering problem can caused the high frequency oscillation of the controllers output.

Secondly, sensitive where this controller is very sensitive to the noise when the input signals is very close to zero. Equivalent dynamic formulation is another disadvantage where calculation of equivalent control formulation is difficult since it is depending on the nonlinear dynamic equation [7]. Many papers were presented to solve these problems as reported in [8-11].

Since the invention of fuzzy logic theory in 1965 by Zadeh, it has been used in many areas. Fuzzy Logic Controller (FLC) is one of the most important applications of fuzzy logic theory [12]. This controller can be used to control nonlinear, uncertain and noisy systems. Fuzzy logic control systems, do not use complex mathematically models of plant for analysis. This method is free of some model-based techniques as in classical controllers. It must be noted here that the application of fuzzy logic is not limited only to modeling of nonlinear systems [13-17] but also this method can help engineers to design easier controller. However pure FLC works in many engineering applications but, it cannot guarantee two most important challenges in control, namely, stability and acceptable performance [18].

Some researchers had applied fuzzy logic methodology in sliding mode controllers (FSMC) in order to reduce the chattering and to solve the nonlinear dynamic equivalent problems in pure sliding mode controller (FSMC) [19-23, 63-68] and the other researchers applied sliding mode methodology in fuzzy logic controller (SMFC) as to improve the stability of the systems [24-28].

Adaptive control used in systems whose dynamic parameters are varying and need to be trained on line. In general states adaptive control can be classified into two main groups: traditional adaptive method and fuzzy adaptive method, where traditional adaptive method need to have some information about dynamic plant and some dynamic parameters must be known but fuzzy adaptive method can train the variation of parameters by expert knowledge. Adaptive fuzzy inference system provide a good knowledge tools to adjust a complex uncertain nonlinear system with changing dynamics to have an acceptable performance [29] Combined adaptive method to artificial sliding mode controllers can help the controllers to have a better performance by online tuning the nonlinear and time variant parameters [30-35, 61-68].

This paper is organized as follows: In section 2, main subject of proposed methodology is presented. Detail of fuzzy logic controllers and fuzzy rule base, the main subject of sliding mode controller and formulation, the main subject of designing fuzzy sliding mode controller and the design of sliding mode fuzzy artificial chatter free fuzzy sliding mode controller are presented which this method is used to reduce the chattering and estimate the equivalent part. In section 3, modeling robot manipulator and PUMA robot manipulator formulation are presented. This section covers the following details, introducing the dynamic formulation of robot manipulator and calculates the dynamic equation of PUMA robot manipulator. the simulation result is presented in section 4 and finally in section 5, the conclusion is presented.

2. PROPOSED METHODOLOGY

Sliding Mode Controller: This section provides a review of classical sliding mode control and the problem of formulation based on [4]; [37-39, 61-68]. Basically formulation of a nonlinear single input dynamic system is:

$$x^{(n)} = f(\vec{x}) + b(\vec{x})u \tag{1}$$

Where u is the vector of control input, $x^{(n)}$ is the n^{th} derivation of$x$, $x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T$ is the state vector, $f(x)$ is unknown or uncertainty, and $b(x)$ is of known *sign* function. The control input has the following form:

$$\tau_{(q,t)} = \begin{cases} \tau_i^+(q,t) & if\ S_i > 0 \\ \tau_i^-(q,t) & if\ S_i < 0 \end{cases} \tag{2}$$

The control problem is truck to the desired state it means that $x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d^{(n-1)}]^T$, and have an acceptable error which is given by:

$$\tilde{x} = x - x_d = [\tilde{x}, \dots, \tilde{x}^{(n-1)}]^T \tag{3}$$

A time-varying sliding surface $s(x,t)$ is given by the following equation [61-68]:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1} \tilde{x} = 0 \tag{4}$$

where λ is the constant and it is positive. The main derivative of S is

$$\dot{S} = \ddot{q}_d + \lambda \dot{e} \tag{5}$$

The Lyapunov function V is defined as:

$$V = \frac{1}{2} S^T M S \tag{6}$$

Based on the above discussion, the control law for a multi DOF robot manipulator can be written as:

$$U = U_{eq} + U_{dis} \tag{7}$$

Where, the model-based component U_{eq} compensate for the nominal dynamics of the systems. So U_{eq} can be calculated as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{8}$$

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

$$U_{dis} = K(\vec{x},t).\,sgn(s) \qquad sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{9}$$

Where the $K(\vec{x},t)$ is the positive constant. Since the control input U has to be a discontinuous term, the control switching could not to be perfect and this will have chattering. Chattering can caused the high frequency oscillation of the controllers output and fast breakdown of mechanical elements in actuators. Chattering is one of the most important challenging in sliding mode controllers which, many papers have been presented to solve this problems [39]. To reduce chattering many researchers introduced the boundary layer methods, which in this method the basic idea is to replace the discontinuous method by saturation (linear) method with small neighbourhood of the switching surface. Several papers have been presented about reduce the chattering [27]; [18]; [40]. Therefore the saturation function $Sat({}^{S}/_{\emptyset})$ added to the control law:

$$U = K(\vec{x},t).\,Sat\left({}^{S}/_{\emptyset}\right) \qquad sat\left({}^{S}/_{\emptyset}\right) = \begin{cases} 1 & ({}^{S}/_{\emptyset} > 1) \\ -1 & ({}^{S}/_{\emptyset} < 1) \\ {}^{S}/_{\emptyset} & (-1 < {}^{S}/_{\emptyset} < 1) \end{cases} \tag{10}$$

where $\emptyset$ is the width of the boundary layer, therefore the control output can be write as

$$U = U_{eq} + K.\,sat\left({}^{S}/_{\emptyset}\right) = \begin{cases} U_{eq} + K.\,sgn(S) & ,|S| \geq \emptyset \\ U_{eq} + K.\,{}^{S}/_{\emptyset} & ,|S| < \emptyset \end{cases} \tag{11}$$

Suppose that the dynamic formulation of robot manipulate is written by the following equation [61-68]:

$$\tau = M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) \tag{12}$$

the lyapunov formulation can be written as follows,

$$V = \frac{1}{2}S^{T}.M.S \tag{13}$$

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2}S^{T}.\dot{M}.S + S^{T}\,M\dot{S} \tag{14}$$

the dynamic equation of robot manipulator can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \tag{15}$$

it is assumed that

$$S^{T}(\dot{M} - 2V)S = 0 \tag{16}$$

by substituting (15) in (14)

$$\dot{V} = \frac{1}{2}S^{T}\dot{M}S - S^{T}VS + S^{T}\left(M\dot{S} + VS + G - \tau\right) = S^{T}\left(M\dot{S} + VS + G - \tau\right) \tag{17}$$

suppose the control input is written as follows

$$\hat{\tau} = \widehat{\tau_{eq}} + \widehat{\tau_{dis}} = \left[\widehat{M^{-1}}\left(\hat{V} + \hat{G}\right) + \dot{S}\right]\hat{M} + K.\,sgn(S) + K_{v}S \tag{18}$$

by replacing the equation (18) in (17)

$$\dot{V} = S^{T}(M\dot{S} + VS + G - \hat{M}\dot{S} - \hat{V}S - \hat{G} - K_{v}S - Ksgn(S) = S^{T}\left(\tilde{M}\dot{S} + \tilde{V}S + \tilde{G} - K_{v}S - Ksg\right. \tag{19}$$

it is obvious that

$$\left|\tilde{M}\dot{S} + \tilde{V}S + \tilde{G} - K_v S\right| \le \left|\tilde{M}\dot{S}\right| + \left|\tilde{V}S\right| + \left|\tilde{G}\right| + \left|K_v S\right| \tag{20}$$

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

$$K_u = \left[\left|\tilde{M}\dot{S}\right| + \left|VS\right| + \left|G\right| + \left|K_v S\right| + \eta\right]_i, i = 1, 2, 3, 4, \dots \tag{21}$$

the equation (33) can be written as

$$K_u \ge \left|\left[\tilde{M}\dot{S} + VS + G - K_v S\right]_i\right| + \eta_i \tag{22}$$

therefore, it can be shown that [63-68]

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i \left|S_i\right| \tag{23}$$

Consequently the equation (40) guaranties the stability of the Lyapunov equation

Combinations of fuzzy logic systems with sliding mode method have been proposed by several researchers. As mention previously, SMFC is fuzzy controller based on sliding mode method for easy implementation, stability, and robustness. The SMFC initially proposed by Palm to design nonlinear approximation boundary layer instead of linear approximation [27]. The main drawback of SMFC is the value of sliding surface λ which must be pre-defined. The most important advantage of SMFC compare to pure SMC is design a nonlinearity boundary layer. The system performance is sensitive to the sliding surface sloop λ for both sliding mode controller and sliding mode fuzzy controller application. For instance, if large value of λ are chosen the response is very fast but the system is very unstable and conversely, if small value of λ is considered the response of the system is very slow but the system is usually stable. Therefore, calculation the optimum value of λ for a system is one of the most important challenges. Even though most of time the control performance of FLC and SMFC is similar, the SMFC has two most important advantages;

The number of rule base is smaller and
better robustness and stability.

Several papers have been proposed on this method and several researchers' works in this area [41-46]. To compensate the nonlinearity for dynamic equivalent control several researchers used model base fuzzy controller instead of classical equivalent controller that was employed to obtain the desired control behaviour and a fuzzy switching control was applied to reinforce system performance. In the proposed fuzzy sliding mode control fuzzy rule base was designed to estimate the dynamic equivalent part. A block diagram for proposed fuzzy sliding mode controller is shown in Figure 1. In this method fuzzy rule for sliding surface (S) to design fuzzy error base-like equivalent control was obtained the rules whereused instead of nonlinear dynamic equation of equivalent control to reduce the chattering and also to eliminate the nonlinear formulation of dynamic equivalent control term.

$$\begin{gathered} \mathbf{1 > if\ S\ is\ N.B\ then\ \ \tau\ \ is\ N.B} \\ \mathbf{2 > if\ S\ is\ Z\ then\ \tau\ is\ Z} \end{gathered} \tag{24}$$

In FSMC the tracking error is defined as:

$$e = q_d - q_a \tag{25}$$

where $q_d = [q_{1d}, q_{2d}, q_{3d}]^T$ is desired output and $q_a = [q_{1a}, q_{2a}, q_{3a}]^T$ is an actual output. The sliding surface is defined as follows:

$$S = \dot{e} + \lambda e \tag{26}$$

where $\lambda = diag[\lambda_1, \lambda_2, \lambda_3]$ is chosen as the bandwidth of the robot manipulator controller. The time derivative of S can be calculated by the following equation

$$\dot{S} = \ddot{q}_d + \lambda_1 \dot{e} \tag{27}$$

Based on classical SMC the FSMC can be calculated as

$$\hat{U} = U_{fuzzy} + U_r \tag{28}$$

Where, the model-based component $\hat{U}_{eq}$ compensate for the nominal dynamics of systems. So $\hat{\tau}_{eq}$ can be calculated as

$$U_{fuzzy} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{29}$$

and U_r is

$$U_{sat} = K.sat(S) \tag{30}$$

To eliminate the chattering fuzzy inference system is used instead of saturation function to design nonlinear sliding function which as a summary the design of fuzzy logic controller for FSMC has five steps:

1. **Determine inputs and outputs:** This controller has one input (S) and one output (α). The input is sliding function (S) and the output is coefficient which estimate the saturation function (α).
2. **Find membership function and linguistic variable:** The linguistic variables for sliding surface (S) are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and the linguistic variables to find the saturation coefficient (α) are; Large Left (LL), Medium Left (ML), Small Left (SL), Zero (Z), Small Right (SR), Medium Right (MR), Large Right (LR).
3. **Choice of shape of membership function:** In this work triangular membership function was selected.
4. **Design fuzzy rule table:** design the rule base of fuzzy logic controller can play important role to design best performance FSMC, suppose that two fuzzy rules in this controller are

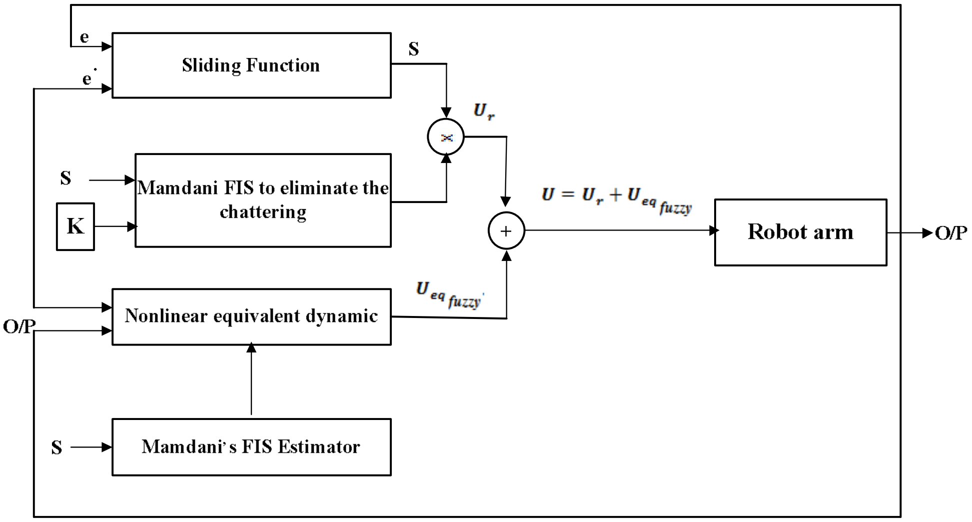


FIGURE 1: Block diagram of proposed artificial chattering free FSMC with minimum rule base

F.R^1**:** ***IF S is Z, THEN*** α ***is Z.*** **(31)**
F.R^2**:** ***IF S is (PB) THEN*** α ***is (LR).***

The complete rule base for this controller is shown in Table 1.

TABLE 1: Rule table for proposed FSMC

S	NB	NM	NS	Z	PS	PM	PB
τ	LL	ML	SL	Z	SR	MR	LR

The control strategy that deduced by Table1 are

- If sliding surface (S) is N.B, the control applied is N.B for moving S to S=0.
- If sliding surface (S) is Z, the control applied is Z for moving S to S=0.

5. **Defuzzification**: The final step to design fuzzy logic controller is deffuzification , there are many deffuzzification methods in the literature, in this controller the COG method will be used, where this is given by

$$COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)} \tag{32}$$

The fuzzy system can be defined as below

$$f(x) = \tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(e, \dot{e}) \tag{33}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)x_i}}{\sum_i \mu_{(xi)}} \tag{34}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (B.1) and $\mu_{(xi)}$ is membership function.

error base fuzzy controller can be defined as

$$\tau_{fuzzy} = \psi(e, \dot{e}) \tag{35}$$

According to the formulation (43)

$$if\ S = 0\ then\ -\dot{e} = \lambda e \tag{36}$$

the fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min \left[Sup_{x \in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \right|\right] \tag{37}$$

Where $\theta^*$is the minimum error, $sup_{x \in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \right|$ is the minimum approximation error. Figure 2 is shown the fuzzy instead of saturation function.

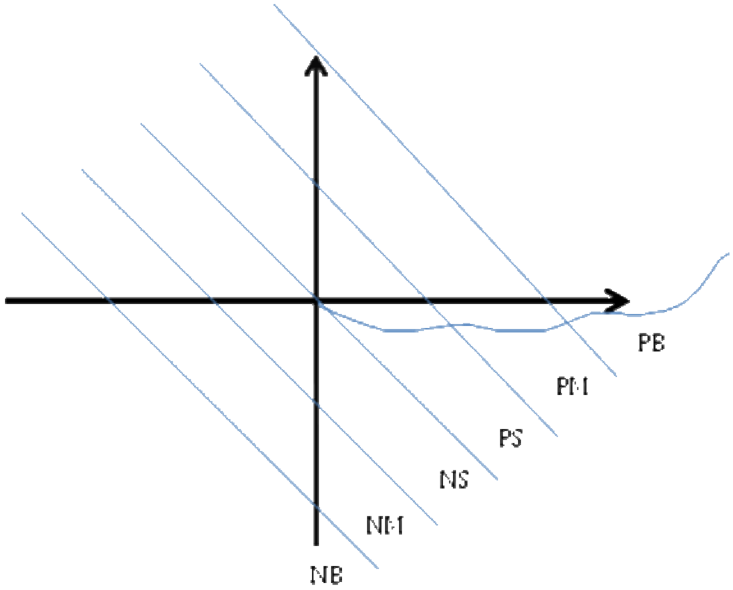


FIGURE 2: Nonlinear fuzzy inference system instead of saturation function

The system performance in FSMC is sensitive to sliding surface slope, λ. Thus, determination of an optimum λ value for a system is an important problem. If the system parameters are unknown or uncertain, the problem becomes more highlighted. This problem may be solved by adjusting the surface slope and boundary layer thickness of the sliding mode controller continuously in real-time. Several researchers' works on adaptive sliding mode control and their applications in robotic manipulator has been investigated in [30-35]; [47-58]. In this way, the performance of the overall system can be improved with respect to the classical sliding mode controller.

This section focuses on, self tuning gain updating factor for two most important factor in FSMC, namely, sliding surface slop (λ) and boundary layer thickness ($\emptyset$). Self tuning-FSMC has strong resistance and can solve the uncertainty problems. Several researchers work on AFSMC in robot manipulator [24-28]; [59-60]. The block diagram for this method is shown in Figure 3. In this controller the actual sliding surface gain (λ) is obtained by multiplying the sliding surface with gain updating factor(α). The gain updating factor(α) is calculated on-line by fuzzy dynamic model independent which has sliding surface (S) as its inputs. The gain updating factor is independent of any dynamic model of robotic manipulator parameters. Assuming that$\alpha = 1$, following steps used to tune the controller: adjust the value of λ, $\emptyset$ and α to have an acceptable performance for any one trajectory by using trial and error. Some researcher design MIMO adaptive fuzzy sliding mode controller [30-31]; [35] and also someone design SISO adaptive fuzzy sliding mode controller [32]; [34].

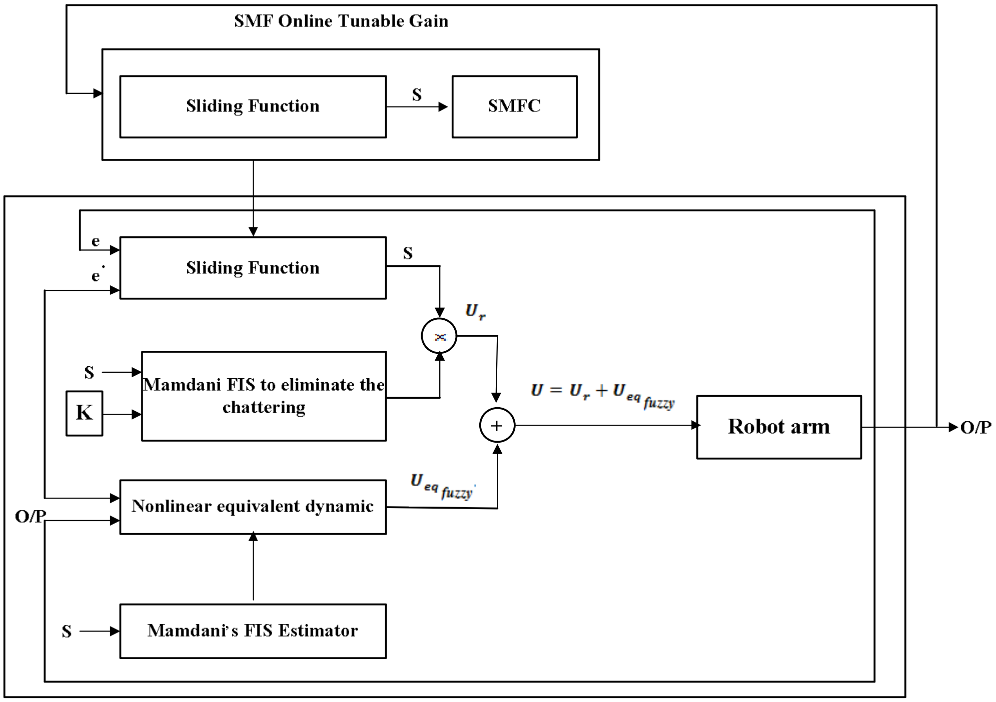


FIGURE 3: Block diagram of proposed artificial chattering free self tuning fuzzy sliding mode controller with minimum rule base in fuzzy equivalent part and fuzzy supervisory.

The adaptive controller is used to find the minimum errors of $\theta - \theta^*$.

suppose K_j is defined as follows

$$K_j = \frac{\sum_{l=1}^{M} \theta_j^l [\mu_A(S_j)]}{\sum_{l=1}^{M} [\mu_A(S_j)]} = \theta_j^T \zeta_j(S_j) \tag{38}$$

Where $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \tag{39}$$

the adaption low is defined as

$$\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j) \tag{40}$$

where the γ_{sj} is the positive constant.
According to the formulation (11) and (12) in addition from (10) and (40)

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) = \sum_{l=1}^{M} \theta^T \zeta(x) - \lambda S - K \tag{41}$$

The dynamic equation of robot manipulator can be written based on the sliding surface as;

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \tag{42}$$

It is supposed that

$$S^T(\dot{M} - 2V)S = 0 \tag{43}$$

it can be shown that

$$M\dot{S} + (V + \lambda)S = \Delta f - K \tag{44}$$

where $\Delta f = [M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q)] - \sum_{l=1}^{M} \theta^T \zeta(x)$
as a result $\dot{V}$ is became

$$\begin{aligned}
\dot{V} &= \frac{1}{2}S^T\dot{M}S - S^T VS + \sum \frac{1}{\gamma_{sj}} \varnothing_j^T \dot{\varnothing}_j \\
&= S^T(-\lambda S + \Delta f - K) + \sum \frac{1}{\gamma_{sj}} \varnothing_j^T \dot{\varnothing}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - K_j)] - S^T\lambda S + \sum \frac{1}{\gamma_{sj}} \varnothing_j^T \dot{\varnothing}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - \theta_j^T \zeta_j(S_j))] - S^T\lambda S + \sum \frac{1}{\gamma_{sj}} \varnothing_j^T \dot{\varnothing}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - (\theta_j^*)^T \zeta_j(S_j) + \varnothing_j^T \zeta_j(S_j))] - S^T\lambda S + \sum \frac{1}{\gamma_{sj}} \varnothing_j^T \dot{\varnothing}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right)] - S^T\lambda S)] + \sum (\frac{1}{\gamma_{sj}} \varnothing_j^T [\gamma_{sj}\zeta_j(S_j)S_j + \dot{\varnothing}_j])
\end{aligned}$$

where $\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j)$ is adaption law, $\dot{\varnothing}_j = -\dot{\theta}_j = -\gamma_{sj} S_j \zeta_j(S_j)$,
consequently $\dot{V}$ can be considered by

$$\dot{V} = \sum_{j=1}^{m} [S_j \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right)] - S^T\lambda S \tag{45}$$

the minimum error can be defined by

$$e_{mj} = \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right) \tag{46}$$

$\dot{V}$ is intended as follows

$$\begin{aligned}
\dot{V} &= \sum_{j=1}^{m} [S_j e_{mj}] - S^T\lambda S \\
&\le \textstyle\sum_{j=1}^{m} |S_j||e_{mj}| - S^T\lambda S \\
&= \sum_{j=1}^{m} |S_j||e_{mj}| - \lambda_j S_j{}^2 \\
&= \sum_{j=1}^{m} |S_j|(|e_{mj}| - \lambda_j S_j)
\end{aligned} \tag{47}$$

For continuous function $g(x)$, and suppose $\varepsilon > 0$ it is defined the fuzzy logic system in form of (36) such that

$$Sup_{x \in U}|f(x) - g(x)| < \epsilon \tag{48}$$

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

$$if\ \lambda_j = \alpha \quad that \quad \alpha|S_j| > e_{mj}\ (S_j \neq 0) \quad then \quad \dot{V} < 0\ for\ (S_j \neq 0) \tag{49}$$

3. APPLICATION: PUMA ROBOT MANIPULATOR

Dynamic modelling of robot manipulators is used to describe the behaviour of robot manipulator, design of model based controller, and simulation results. The dynamic modelling describe 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 behaviour of system. It is well known that the equation of an *n-DOF* robot manipulator governed by the following equation [36]; [2]:

$$M(q)\ddot{q} + N(q,\dot{q}) = \tau \tag{50}$$

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:

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) \tag{51}$$

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 (50) 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 variable q_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [2]:

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \tag{52}$$

This technique is very attractive from a control point of view. The three degrees of freedom PUMA robot has the same configuration space equation general form as its 6-DOF convenient. In this type, the last three joints are blocked, so, only three links of PUMA robot are used in this paper, $q_4 = q_5 = q_6 = 0$. The dynamic equation of PUMA robot manipulator is given by [61-68]

$$M(\theta)\begin{bmatrix}\ddot{\theta}1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix} + \mathrm{B}(\theta)\begin{bmatrix}\dot{\theta}_1\dot{\theta}_2\\ \dot{\theta}_1\dot{\theta}_3\\ \dot{\theta}_2\dot{\theta}_3\end{bmatrix} + C(\theta)\begin{bmatrix}\dot{\theta}_1^2\\ \dot{\theta}_2^2\\ \dot{\theta}_3^2\end{bmatrix} + G(\theta) = \begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} \tag{53}$$

Where

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0\\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0\\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0\\ 0 & 0 & 0 & M_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & M_{55} & 0\\ 0 & 0 & 0 & 0 & 0 & M_{66}\end{bmatrix} \tag{54}$$

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 & 0\\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix} \tag{55}$$

$$C(q) = \begin{bmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0\\ C_{21} & 0 & C_{23} & 0 & 0 & 0\\ C_{31} & C_{32} & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ C_{51} & C_{52} & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix} \tag{56}$$

$$G(q) = \begin{bmatrix} 0\\ g_2\\ g_3\\ 0\\ g_5\\ 0\end{bmatrix} \tag{57}$$

Suppose $\ddot{q}$ is written as follows

$$\ddot{q} = M^{-1}(q).\{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{58}$$

and K is introduced as

$$I = \{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{59}$$

$\ddot{q}$ can be written as

$$\ddot{q} = M^{-1}(q).I \tag{60}$$

Therefore I for PUMA robot manipulator can be calculated by the following equation

$$I_1 = \tau_1 - [\, b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3] - [\, C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2] - g_1 \tag{60}$$

$$I_2 = \tau_2 - [\, b_{223}\dot{q}_2\dot{q}_3] - [\, C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2] - g_2 \tag{61}$$

$$I_3 = \tau_3 - [C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2] - g_3 \tag{62}$$

$$I_4 = \tau_4 - [\, b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3] - g_4 \tag{63}$$

$$I_5 = \tau_5 - [\, C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2] - g_5 \tag{64}$$

$$I_6 = \tau_6 \tag{65}$$

4. RESULTS

Classical sliding mode control (SMC), fuzzy sliding mode control (FSMC) and artificial chattering free adaptive FSMC are implemented in Matlab/Simulink environment. Changing updating factor performance, tracking performance, error, and robustness are compared.

• Changing Sliding Surface Slope Performance

For various value of sliding surface slope (λ) in SMC, AFSMC and ASMFC, the error and trajectory performances are shown in Figures 4 to 7.

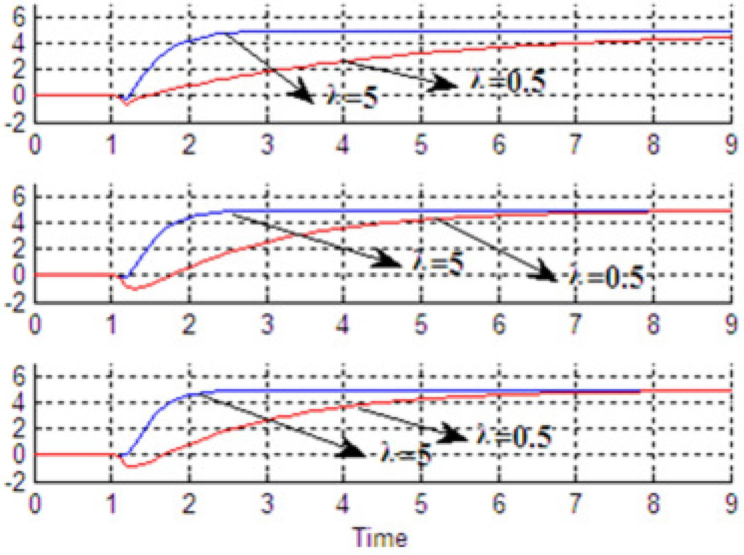


FIGURE 4: SMC trajectory performance, first; second and third link

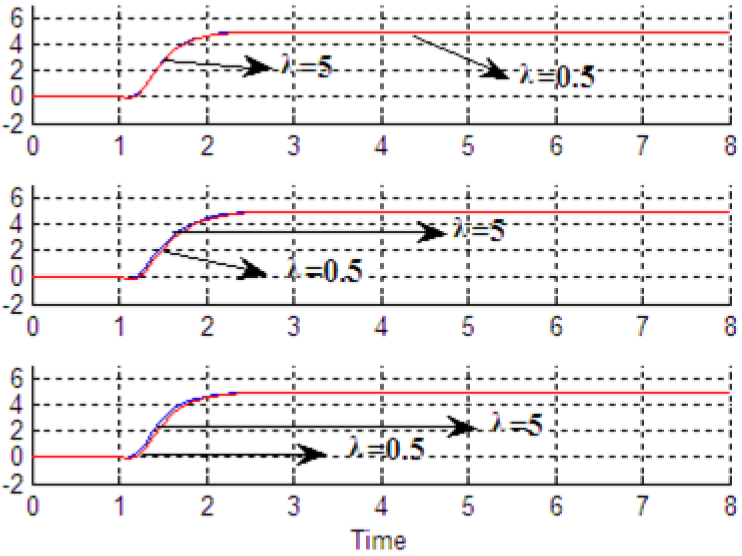


FIGURE 5: Artificial chattering free adaptive FSMC trajectory performance, first; second and third link

Figures 4 and 5 are shown trajectory performance with different sliding function for, Artificial chattering free adaptive FSMC and SMC. It is shown that the sensitivity in Artificial chattering free adaptive FSMC to sliding function is lower than SMC. Figures 6 and 7 are shown the error performance with different sliding surface slope in classical SMC and Artificial chattering free adaptive FSMC. For various sliding surface slope (λ), Artificial chattering free adaptive FSMC, has better error performance compare to classical SMC.

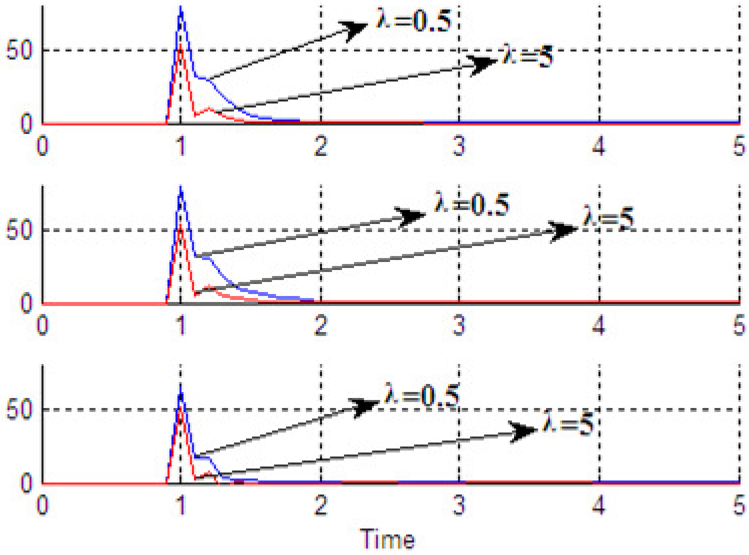


FIGURE 6: Error performance: SMC (first; second and third link)

The new sliding surface slope coefficient is updated by multiplying the error new coefficient (K_c) with predetermined slope value **(λ).**

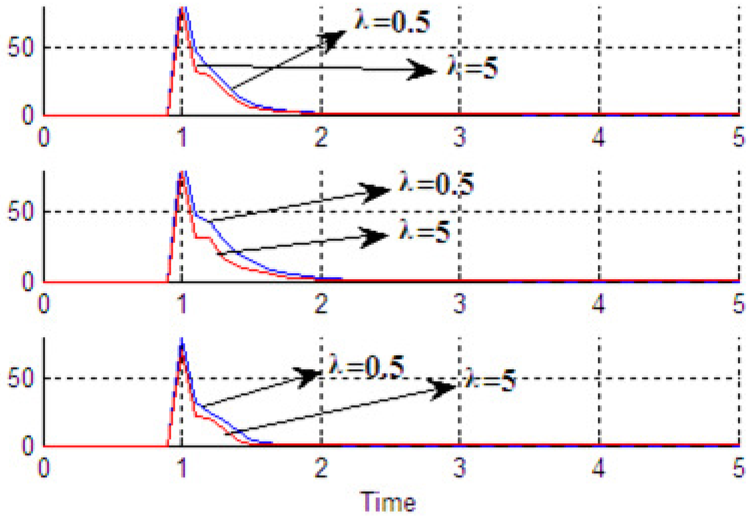


FIGURE 7: Error performance: Artificial chattering free adaptive FSMC (first; second and third link)

- **Tracking Performances**

From the simulation for first, second, and third trajectory without any disturbance, it can be seen that Artificial chattering free adaptive FSMC and classical SMC have same performance. This is primarily due to the constant parameters in simulation. Figure 8 is shown tracking performance in certain system for SMC, FSMC and proposed method.

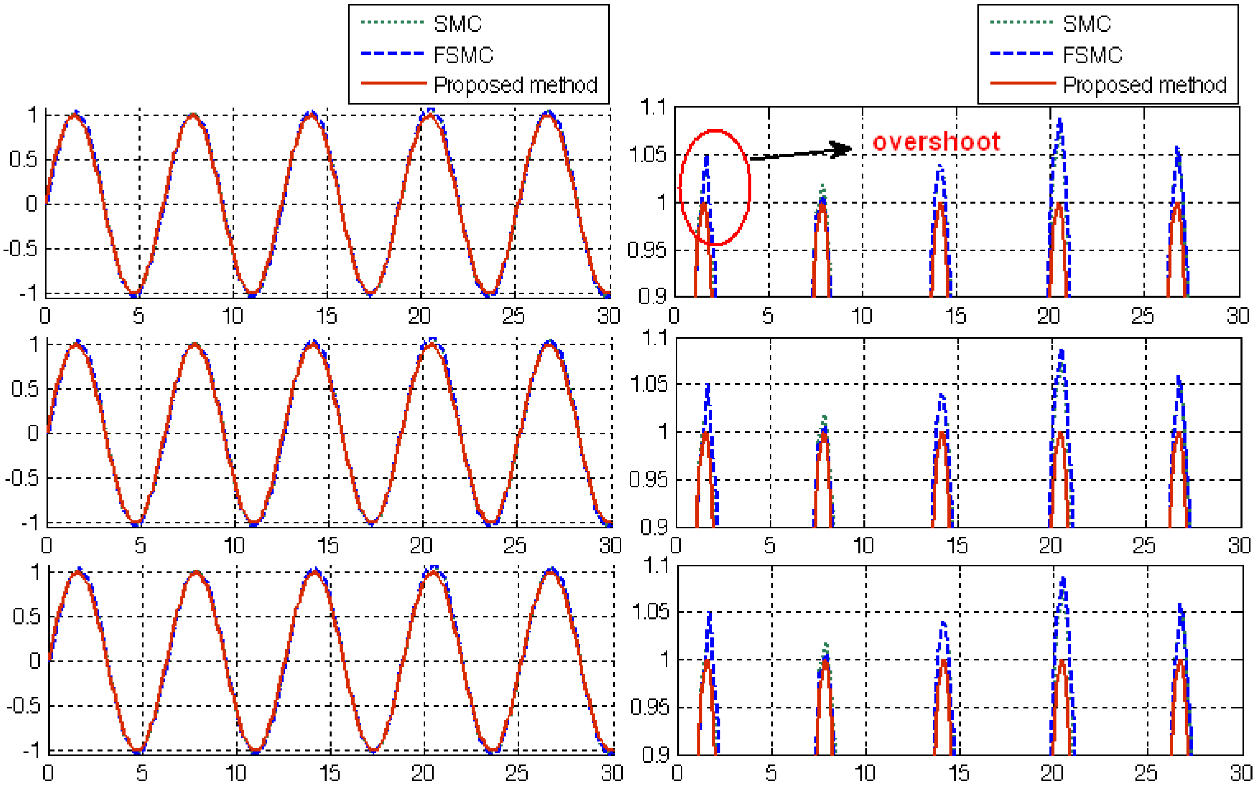


FIGURE 8: Trajectory performance: Artificial chattering free adaptive FSMC, SMC and FSMC (first; second and third link)

- **Disturbance Rejection**

A band limited white noise with predefined of 40% the power of input signal is applied to the Sinuse response. Figure 9 is shown disturbance rejection for Artificial chattering free adaptive FSMC, SMC and FSMC.

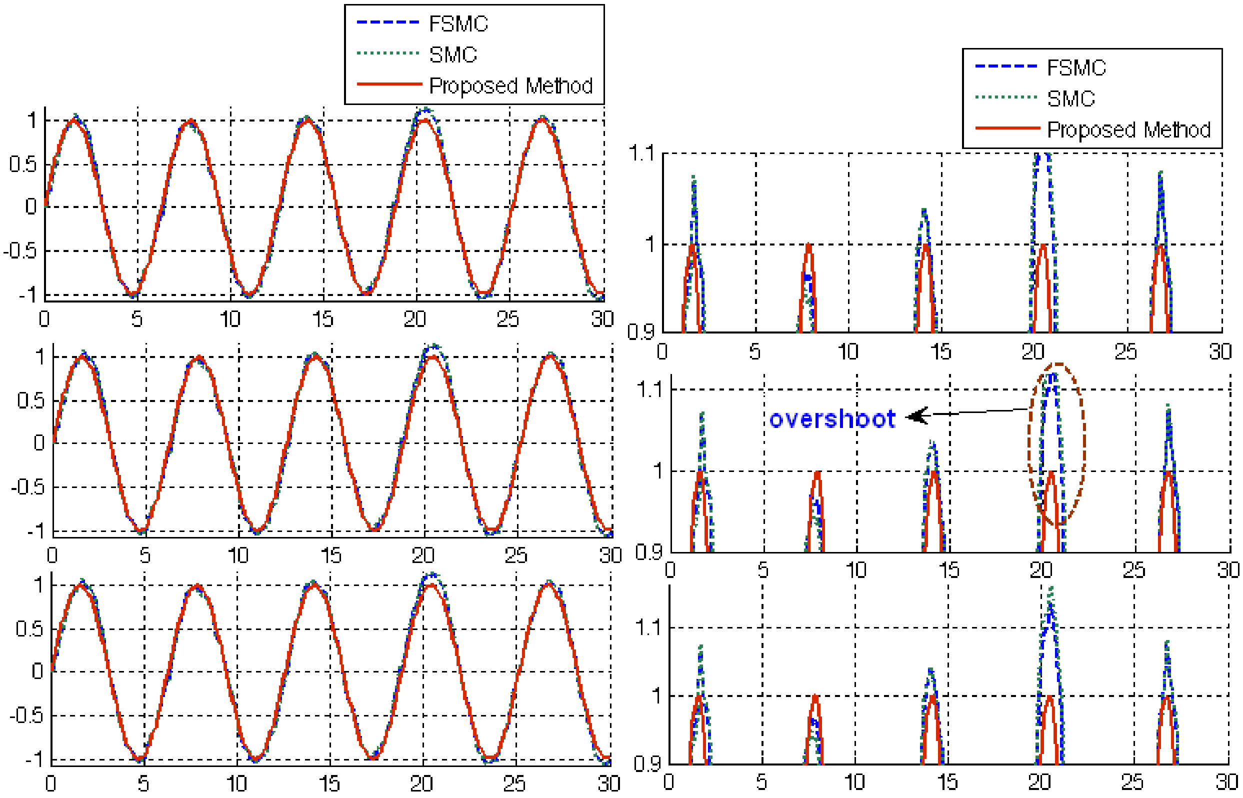


FIGURE 9: Disturbance rejection: Artificial chattering free adaptive FSMC, SMC and FSMC (first; second and third link)

- **Errors in the Model**

Although SMC and FSMC have the same error rate (refer to Table:2), they have oscillation tracking which causes chattering phenomenon at the presence of disturbances. As it is obvious in Table: 2 proposed methods is a FSMC which tuning on-line and FSMC is a SMC which estimate the equivalent part therefore FSMC have acceptable performance with regard to SMC in presence of certain and uncertainty but the best performance is in Artificial chattering free adaptive FSMC.

TABLE 2 : RMS Error Rate of Presented controllers

RMS Error Rate	**SMC**	**FSMC**	**Proposed method**
Without Noise	**1e-3**	**1.2e-3**	**1e-7**
With Noise	**0.012**	**0.013**	**1.12e-6**

- **Chattering Phenomenon**
  As mentioned in previous, chattering is one of the most important challenges in sliding mode controller which one of the major objectives in this research is reduce or remove the chattering in system's output. To reduce the chattering researcher is used fuzzy inference method instead of swithing function. Figure 10 has shown the power of fuzzy boundary layer (fuzzy saturation) method to reduce the chattering in proposed method.

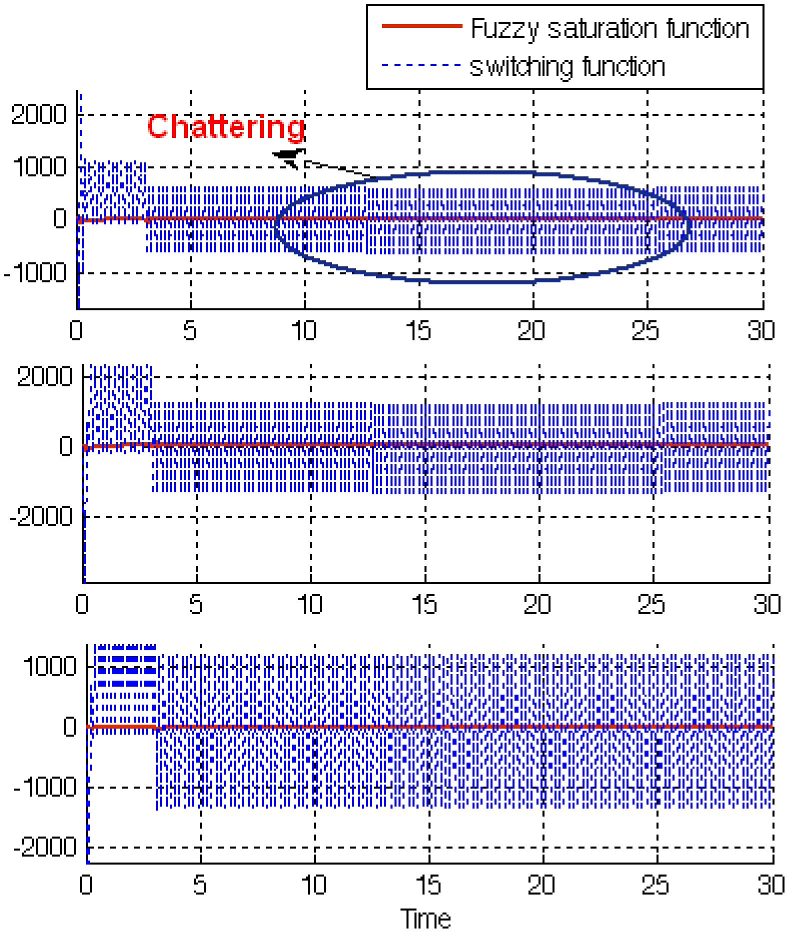


FIGURE 10: chattering phenomenon: Artificial chattering free adaptive FSMC with switching function and fuzzy saturation function (first; second and third link)

5. CONCLUSION

Refer to the research, a 7 rules Mamdani's artificial sliding mode fuzzy chattering free fuzzy sliding mode control and this suitability for use in the control of robot manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties and external disturbances. Sliding mode control methodology is selected as a frame work to construct the control law and address the stability and robustness of the close-loop system. The proposed approach effectively combines the design techniques from sliding mode control, fuzzy logic and adaptive control to improve the performance (e.g., trajectory, disturbance rejection, error and chattering) and enhance the robustness property of the controller. Each method by adding to the previous controller has covered negative points. The system performance in sliding mode controller and fuzzy sliding mode controller are sensitive to the sliding function. Therefore, compute the optimum value of sliding function for a system is the important which this problem has solved by adjusting surface slope of the sliding function continuously in real-time. The chattering phenomenon is eliminate by fuzzy method when estimate the saturation function with 7 rule base. In this way, the overall system performance has improved with respect to the classical sliding mode controller. This controller solved chattering phenomenon as well as mathematical nonlinear equivalent part by applied fuzzy supervisory method in fuzzy sliding mode controller and artificial chattering free adaptive FSMC.

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*, et al.*, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo*, et al.*, *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*, et al.*, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* vol. 52, pp. 2085-2102, 2007.

[8] J. Wang*, et al.*, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* vol. 122, pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng*, et al.*, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* 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," 2005.

[16] D. Nguyen-Tuong*, et al.*, "Computed torque control with nonparametric regression models," 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo*, et al.*, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* vol. 76, pp. 212-232, 2002.

[19] K. D. Young*, et al.*, "A control engineer's guide to sliding mode control," 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* 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,* vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics,* 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,* 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,* vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," 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,* vol. 18, pp. 798-810, 2010.

[30] B. Wu*, et al.*, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," 2006, pp. 5-945.

[31] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems,* vol. 90, pp. 111-127, 1997.

[32] L. Reznik, *Fuzzy controllers*: Butterworth-Heinemann, 1997.

[33] J. Zhou and P. Coiffet, "Fuzzy control of robots," 2002, pp. 1357-1364.

[34] S. Banerjee and P. Y. Woo, "Fuzzy logic control of robot manipulator," 2002, pp. 87-88.

[35] K. Kumbla*, et al.*, "Soft computing for autonomous robotic systems," *Computers and Electrical Engineering,* vol. 26, pp. 5-32, 2000.

[36] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Transactions on systems, man and cybernetics,* vol. 20, pp. 404-418, 1990.

[37] R. J. Wai*, et al.*, "Implementation of artificial intelligent control in single-link flexible robot arm," 2003, pp. 1270-1275.

[38] R. J. Wai and M. C. Lee, "Intelligent optimal control of single-link flexible robot arm," *Industrial Electronics, IEEE Transactions on,* vol. 51, pp. 201-220, 2004.

[39] M. B. Menhaj and M. Rouhani, "A novel neuro-based model reference adaptive control for a two link robot arm," 2002, pp. 47-52.

[40] S. Mohan and S. Bhanot, "Comparative study of some adaptive fuzzy algorithms for manipulator control," *International Journal of Computational Intelligence,* vol. 3, pp. 303–311, 2006.

[41] F. Barrero*, et al.*, "Speed control of induction motors using a novel fuzzy sliding-mode structure," *Fuzzy Systems, IEEE Transactions on,* vol. 10, pp. 375-383, 2002.

[42] Y. C. Hsu and H. A. Malki, "Fuzzy variable structure control for MIMO systems," 2002, pp. 280-285.

[43] Y. C. Hsueh*, et al.*, "Self-tuning sliding mode controller design for a class of nonlinear control systems," 2009, pp. 2337-2342.

[44] R. Shahnazi*, et al.*, "Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control," *Energy Conversion, IEEE Transactions on,* vol. 23, pp. 138-147, 2008.

[45] C. C. Chiang and C. H. Wu, "Observer-Based Adaptive Fuzzy Sliding Mode Control of Uncertain Multiple-Input Multiple-Output Nonlinear Systems," 2007, pp. 1-6.

[46] H. Temeltas, "A fuzzy adaptation technique for sliding mode controllers," 2002, pp. 110-115.

[47] C. L. Hwang and S. F. Chao, "A fuzzy-model-based variable structure control for robot arms: theory and experiments," 2005, pp. 5252-5258.

[48] C. G. Lhee*, et al.*, "Sliding mode-like fuzzy logic control with self-tuning the dead zone parameters," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 343-348, 2002.

[49] "Sliding-Like Fuzzy Logic Control with Self-tuning the Dead Zone Parameters," 1999.

[50] X. Zhang*, et al.*, "Adaptive sliding mode-like fuzzy logic control for high order nonlinear systems," pp. 788-792.

[51] M. R. Emami*, et al.*, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems,* vol. 6, 1998.

[52] *1994,* vol. 4, pp. 212-218, 1994.

[53] Z. Kovacic and S. Bogdan, *Fuzzy controller design: theory and applications*: CRC/Taylor & Francis, 2006.

[54] F. Y. Hsu and L. C. Fu, "Nonlinear control of robot manipulators using adaptive fuzzy sliding mode control," 2002, pp. 156-161.

[55] R. G. Berstecher*, et al.*, "An adaptive fuzzy sliding-mode controller," *Industrial Electronics, IEEE Transactions on,* vol. 48, pp. 18-31, 2002.

[56] V. Kim, "Independent joint adaptive fuzzy control of robot manipulator," 2002, pp. 645-652.

[57] Y. Wang and T. Chai, "Robust adaptive fuzzy observer design in robot arms," 2005, pp. 857-862.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar*, et al.*, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* 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,* vol. 33, pp. 149-159, 2003.

[61] Piltan, F., et al., "Design sliding mode controller for robot manipulator with artificial tunable gain," Canaidian Journal of pure and applied science, 5 (2): 1573-1579, 2011.

[62] 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

[63] F. Piltan*, et al.*, "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," *Australian Journal of Basic and Applied Sciences, 5(6)*, pp. 509-522, 2011.

[64] Piltan, F., et al., "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[65] Piltan, F., et al., "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.

[66] Piltan, F., et al., "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[67] Piltan, F., et al., "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

[68] Piltan, F., et al., "Design Adaptive Fuzzy Robust Controllers for Robot Manipulator," World Applied Science Journal, 12 (12): 2317-2329, 2011.

Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System

Farzin Piltan SSP.ROBOTIC@yahoo.com
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia 43400
Serdang, Selangor, Malaysia

N. Sulaiman nasri@eng.upm.edu.my
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia 43400
Serdang, Selangor, Malaysia

Payman Ferdosali 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

Mehdi Rashidi 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

Zahra Tajpeikar 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

This research is focused on proposed adaptive fuzzy sliding mode algorithms with the adaptation laws derived in the Lyapunov sense. The stability of the closed-loop system is proved mathematically based on the Lyapunov method. Adaptive MIMO fuzzy compensate fuzzy sliding mode method design a MIMO fuzzy system to compensate for the model uncertainties of the system, and chattering also solved by linear saturation method. Since there is no tuning method to adjust the premise part of fuzzy rules so we presented a scheme to online tune consequence part of fuzzy rules. Classical sliding mode control is robust to control model uncertainties and external disturbances. A sliding mode method with a switching control low guarantees the stability of the certain and/or uncertain system, but the addition of the switching control low introduces chattering into the system. One way to reduce or eliminate chattering is to insert a boundary layer method inside of a boundary layer around the sliding surface. Classical sliding mode control method has difficulty in handling unstructured model uncertainties. One can overcome this problem by combining a sliding mode controller and artificial intelligence (e.g. fuzzy logic). To approximate a time-varying nonlinear dynamic system, a fuzzy system requires a large amount of fuzzy rule base. This large number of fuzzy rules will cause a high computation load. The addition of an adaptive law to a fuzzy sliding mode controller to online tune the parameters of the fuzzy rules in use will ensure a moderate computational load. The adaptive laws in this algorithm are designed based on the Lyapunov stability theorem. Asymptotic stability of the closed loop system is also proved in the sense of Lyapunov.

Keywords: Adaptive Fuzzy Sliding Mode Algorithm, Lyapunov Based, Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm, Chattering Phenomenon, Sliding Surface, Fuzzy logic Controller, Adaptive law.

1. INTRODUCTION

The first person who used the word robot was Karel Capek in 1920 in his satirical play, R.U.R (Rossum's Universal Robots). The first person who used the word robotics was the famous author, Issac Asimov along with three fundamental rules. Following World War II, the first industrial robot manipulator have been installation at General Motors in 1962 for the automation. In 1978 the PUMA (Programmable Universal Machine for Assembly) and in 1979 the SCARA (Selective Compliance Assembly Robot Arm) were introduced and they were quickly used in research laboratories and industries. According to the MSN Learning & Research," 700000 robots were in the industrial world in 1995 and over 500000 were used in Japan, about 120000 in Western Europe, and 60000 in the United States [1, 6]." Research about mechanical parts and control methodologies in robotic system is shown; the mechanical design, type of actuators, and type of systems drive play important roles to have the best performance controller. More over types of kinematics chain, i.e., serial Vs. parallel manipulators, and types of connection between link and join actuators, i.e., highly geared systems Vs. direct-drive systems are played important roles to select and design the best acceptable performance controllers[6]. A serial link PUMA 560robot is a sequence of joints and links which begins with a base frame and ends with an end-effector. This type of robot manipulators, comparing with the load capacitance is more weightily because each link must be supported the weights of all next links and actuators between the present link and end-effector[6]. Serial robot manipulators have been used in automotive industry, medical application, and also in research laboratories. One of the most important classifications in controlling the robot manipulator is how the links have connected to the actuators. This classification divides into two main groups: highly geared (e.g., 200 to 1) and direct drive (e.g., 1 to 1). High gear ratios reduce the nonlinear coupling dynamic parameters in robot manipulator. In this case, each joint is modeled the same as SISO systems. In high gear robot manipulators which generally are used in industry, the couplings are modeled as a disturbance for SISO systems. Direct drive increases the coupling of nonlinear dynamic parameters of robot manipulators. This effect should be considered in the design of control systems. As a result some control and robotic researchers' works on nonlinear robust controller design[2]. Although PUMA robot manipulator is high gear but this research focuses on design MIMO controller.

In modern usage, the word of control has many meanings, this word is usually taken to mean regulate, direct or command. The word feedback plays a vital role in the advance engineering and science. The conceptual frame work in Feed-back theory has developed only since world war II. In the twentieth century, there was a rapid growth in the application of feedback controllers in process industries. According to Ogata, to do the first significant work in three-term or PID controllers which Nicholas Minorsky worked on it by automatic controllers in 1922. In 1934, Stefen Black was invention of the feedback amplifiers to develop the negative feedback amplifier[1, 6]. Negative feedback invited communications engineer Harold Black in 1928 and it occurs when the output is subtracted from the input. Automatic control has played an important role in advance science and engineering and its extreme importance in many industrial applications, i.e., aerospace, mechanical engineering and robotic systems. The first significant work in automatic control was James Watt's centrifugal governor for the speed control in motor engine in eighteenth century[2]. There are several methods for controlling a robot manipulator, which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes[1]. Joint space and operational space control are closed loop controllers which they have been used to provide robustness and rejection of disturbance effect. The main target in joint space controller is to design a feedback controller which the actual motion ($q_a(t)$) and desired motion ($q_d(t)$) as closely as possible. This control problem is classified into two main groups. Firstly, transformation the desired motion $X_d(t)$ to joint variable $q_d(t)$ by inverse kinematics of robot manipulators[6]. This control include simple PD control, PID control, inverse dynamic control, Lyapunov-based control, and passivity based control that explained them in the following section. The main target in operational space controller is to design a feedback controller to allow the actual end-effector motion $X_a(t)$ to track the desired endeffector motion $X_d(t)$. This control methodology requires a greater algorithmic complexity and the inverse kinematics used in the feedback control loop. Direct measurement of operational space variables are very expensive that caused to limitation used of this controller in industrial robot manipulators[6]. One of the simplest ways to analysis control of multiple DOF robot manipulators are analyzed each joint separately such as SISO systems and design an independent joint controller for each joint. In this controller, inputs only depends on the velocity and displacement of the corresponding joint and the other parameters between joints such as coupling presented by disturbance input. Joint space controller has many advantages such as one type controllers design for all joints with the same formulation, low cost hardware, and simple structure.

A nonlinear methodology is used for nonlinear uncertain systems (e.g., robot manipulators) to have an acceptable performance. These controllers divided into six groups, namely, feedback linearization (computed-torque control), passivity-based control, sliding mode control (variable structure control), artificial intelligence control, lyapunov-based control and adaptive control[1-20]. Sliding mode controller (SMC) is a powerful nonlinear controller which has been analyzed by many researchers especially in recent years. This theory was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices [1-3, 6, 14]. The main reason to opt for this controller is its acceptable control performance in wide range and solves two most important challenging topics in control which names, stability and robustness [7, 17-20]. Sliding mode controller is divided into two main sub controllers: discontinues controller(τ_{dis}) and equivalent controller(τ_{eq}). Discontinues controller causes an acceptable tracking performance at the expense of very fast switching. In the theory of infinity fast switching can provide a good tracking performance but it also can provide some problems (e.g., system instability and chattering phenomenon). After going toward the sliding surface by discontinues term, equivalent term help to the system dynamics match to the sliding surface[1, 6]. However, this controller used in many applications but, pure sliding mode controller has following challenges: chattering phenomenon, and nonlinear equivalent dynamic formulation [20]. Chattering phenomenon can causes some problems such as saturation and heat the mechanical parts of robot manipulators or drivers. To reduce or eliminate the chattering, various papers have been reported by many researchers which classified into two most important methods: boundary layer saturation method and estimated uncertainties method [1, 10-14]. In boundary layer saturation method, the basic idea is the discontinuous method replacement by saturation (linear) method with small neighborhood of the switching surface. This replacement caused to increase the error performance against with the 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 with boundary layer to improve the industry application [22]. R. 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, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. As mentioned [24]sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most exceptional stability and robustness. Conversely above method has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. To eliminate the above problems control researchers have applied artificial intelligence method (e.g., fuzzy logic) in nonlinear robust controller (e.g., sliding mode controller) besides this technique is very useful in order to implement easily. Estimated uncertainty method used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [25-28]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Their design had three main parts i.e. equivalent, discontinuous and fuzzy estimator tuning part which has reduced the chattering very well. Elmali et al. [27]and Li and Xu [29]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering. This method was tested for the tracking control of the first two links of a SCARA type HITACHI robot. In this technique, digital controller is used to increase the system's response quality. Conversely this method has the following advantages; increasing the controller's response speed and reducing dependence on dynamic system model by on-line control, the main disadvantage are chattering phenomenon and need to improve the performance.

In recent years, artificial intelligence theory has been used in sliding mode control systems. Neural network, fuzzy logic, and neuro-fuzzy are synergically combined with nonlinear classical controller and used in nonlinear, time variant, and uncertainty plant (e.g., robot manipulator). Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control nonlinear, uncertain, and noisy systems. This method is free of some model-based techniques as in classical controllers. As mentioned that fuzzy logic application is not only limited to the modelling of nonlinear systems [31-36]but also this method can help engineers to design easier controller. Control robot arm manipulators using classical controllers are based on manipulator dynamic model. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of robot manipulator, but these models are multi-input, multi-output and non-linear and calculate accurate model can be very difficult. When the system model is unknown or when it is known but complicated, it is difficult or impossible to use classical mathematics to process this model[32]. The main reasons to use fuzzy logic technology are able to give approximate recommended solution for unclear and complicated systems to easy understanding and flexible. Fuzzy logic provides a method which is able to model a controller for nonlinear plant with a set of IF-THEN rules, or it can identify the control actions and describe them by using fuzzy rules. It should be mentioned that application of fuzzy logic is not limited to a system that's difficult for modeling, but it can be used in clear systems that have complicated mathematics models because most of the time it can be shortened in design but there is no high quality design just sometimes we can find design with high quality. Besides using fuzzy logic in the main controller of a control loop, it can be used to design adaptive control, tuning parameters, working in a parallel with the classical and non classical control method [32]. The applications of artificial intelligence such as neural networks and fuzzy logic in modelling and control are significantly growing especially in recent years. For instance, the applications of artificial intelligence, neural networks and fuzzy logic, on robot arm control have reported in [37-39]. Wai et al. [37-38]have proposed a fuzzy neural network (FNN) optimal control system to learn a nonlinear function in the optimal control law. This controller is divided into three main groups: arterial intelligence controller (fuzzy neural network) which it is used to compensate the system's nonlinearity and improves by adaptive method, robust controller to reduce the error and optimal controller which is the main part of this controller. Mohan and Bhanot [40] have addressed comparative study between some adaptive fuzzy, and a new hybrid fuzzy control algorithm for manipulator control. They found that self-organizing fuzzy logic controller and proposed hybrid integrator fuzzy give the best performance as well as simple structure. Research on combinations of fuzzy logic systems with sliding mode method is significantly growing as nonlinear control applications. For instance, the applications of fuzzy logic on sliding mode controller have reported in [24, 41-45]. Research on applied fuzzy logic methodology in sliding mode controller (FSMC) to reduce or eliminate the high frequency oscillation (chattering), to compensate the unknown system dynamics and also to adjust the linear sliding surface slope in pure sliding mode controller considerably improves the robot manipulator control process [42-43]. H.Temeltas [46] has proposed fuzzy adaption techniques for SMC 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. C. L. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode control based on *N* fuzzy based linear state-space to estimate the uncertainties. A multi-input multi-output FSMC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a robot manipulator [42]. Investigation on applied sliding mode methodology in fuzzy logic controller (SMFC) to reduce the fuzzy rules and refine the stability of close loop system in fuzzy logic controller has grown specially in recent years as the robot manipulator control [23]; [48-50, 53]. Lhee et al. [48]have presented a fuzzy logic controller based on sliding mode controller to more formalize and boundary layer thickness. Emami *et al.* [51]have proposed a fuzzy logic approximate inside the boundary layer. H.K.Lee *et al.* [52] have presented self tuning SMFC to reduce the fuzzy rules, increase the stability and to adjust control parameters control automatically. However the application of FSMC and SMFC are growing but the main SMFC drawback compared to FSMC is calculation the value of sliding surface λ pri-defined very carefully. Moreover, the advantages of SMFC compared to FLC reduce the number of fuzzy rule base and increase the robustness and stability. At last FSMC compare to the SMFC is more suitable for implementation action.

In various dynamic parameters systems that need to be training on-line adaptive control methodology is used. Adaptive control methodology can be classified into two main groups, namely, traditional adaptive method and fuzzy adaptive method. Fuzzy adaptive method is used in systems which want to training parameters by expert knowledge. Traditional adaptive method is used in systems which some dynamic parameters are known. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, adaptive method is applied to artificial sliding mode controller. F Y Hsu et al. [54]have presented adaptive fuzzy sliding mode control which can update fuzzy rules to compensate nonlinear parameters and guarantee the stability robot manipulator controller. Y.C. Hsueh et al. [43] have presented self tuning sliding mode controller which can resolve the chattering problem without to using saturation function. For nonlinear dynamic systems (e.g., robot manipulators) with various parameters, adaptive control technique can train the dynamic parameter to have an acceptable controller performance. Calculate several scale factors are common challenge in classical sliding mode controller and fuzzy logic controller, as a result it is used to adjust and tune coefficient. Research on adaptive fuzzy control is significantly growing, for instance, different adaptive fuzzy controllers have been reported in [40, 55-57].

2. PROBLEM STATEMENT AND FORMULATION CHALLENGE

One of the significant challenges in control algorithms is a linear behavior controller design for nonlinear systems. When system works with various parameters and hard nonlinearities this technique is very useful in order to be implemented easily but it has some limitations such as working near the system operating point[2]. 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 nonlinear, uncertain and MIMO[1, 6]. To reduce above challenges the nonlinear robust controllers is used to systems control. One of the powerful nonlinear robust controllers is sliding mode controller (SMC), although this controller has been analyzed by many researchers but the first proposed was in the 1950 [7].This controller is used in wide range areas such as in robotics, in control process, in aerospace applications and in power converters because it has an acceptable control performance and solve some main challenging topics in control such as resistivity to the external disturbance. Even though, this controller is used in wide range areas but, pure sliding mode controller has the following disadvantages: Firstly, chattering problem; which caused the high frequency oscillation in the controllers output. Secondly, equivalent dynamic formulation; calculate the equivalent control formulation is difficult because it depends on the dynamic equation [20]. On the other hand, after the invention of fuzzy logic theory in 1965, this theory was used in wide range applications that fuzzy logic controller (FLC) is one of the most important applications in fuzzy logic theory because the controller has been used for nonlinear and uncertain (e.g., robot manipulator) systems controlling. Conversely pure FLC works in many areas, it cannot guarantee the basic requirement of stability and acceptable performance[8]. Although both SMC and FLC have been applied successfully in many applications but they also have some limitations. The boundary layer method is used to reduce or eliminate the chattering and proposed method focuses on substitution error-base fuzzy logic system instead of dynamic equivalent equation to implement easily and avoid mathematical model base controller. To reduce the effect of uncertainty in proposed method, MIMO adaptive method is applied in fuzzy sliding mode controller in PUMA 560 robot manipulator.

The dynamic formulation of robot manipulate can be written by the following equation

$$\tau = M(q)\ddot{q} + V(q\,.\,\dot{q})\dot{q} + G(q) \tag{1}$$

the lyapunov formulation can be written as follows,

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

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2}S^T.\dot{M}.S + S^T\,M\dot{S} \tag{3}$$

the dynamic equation of robot manipulator can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \tag{4}$$

it is assumed that

$$S^T\left(\dot{M} - 2V\right)S = 0 \tag{5}$$

by substituting (4) in (3)

$$\dot{V}=\frac{1}{2}S^T\dot{M}S-S^TVS+S^T\left(M\dot{S}+VS+G-\tau\right)=S^T\left(M\dot{S}+VS+G-\tau\right) \tag{6}$$

suppose the control input is written as follows

$$\hat{\tau}=\widehat{\tau_{eq}}+\widehat{\tau_{dis}}=\left[\hat{M}^{-1}\left(\hat{V}+\hat{G}\right)+\dot{S}\right]\hat{M}+K.sgn(S)+K_vS \tag{7}$$

by replacing the equation (7) in (6)

$$\dot{V}=S^T(M\dot{S}+VS+G-\hat{M}\dot{S}-\hat{V}S-\hat{G}-K_vS-Ksgn(S)=S^T\left(\tilde{M}\dot{S}+\tilde{V}S+\tilde{G}-K_vS-Ksg\right. \tag{8}$$

it is obvious that

$$\left|\tilde{M}\dot{S}+\tilde{V}S+\tilde{G}-K_vS\right|\le\left|\tilde{M}\dot{S}\right|+\left|\tilde{V}S\right|+\left|\tilde{G}\right|+\left|K_vS\right| \tag{9}$$

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

$$K_u=\left[\left|\tilde{M}\dot{S}\right|+|VS|+|G|+|K_vS|+\eta\right]_i, i=1,2,3,4,\dots \tag{10}$$

the equation (5) can be written as

$$K_u\ge\left|\left[\tilde{M}\dot{S}+VS+G-K_vS\right]_i\right|+\eta_i \tag{11}$$

therefore, it can be shown that

$$\dot{V}\le-\sum_{i=1}^{n}\eta_i|S_i| \tag{12}$$

Based on above discussion, the control law for a multi degrees of freedom robot manipulator is written as:

$$U=U_{eq}+U_r \tag{13}$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and U_{eq} can be calculate as follows:

$$U_{eq}=\left[M^{-1}(B+C+G)+\dot{S}\right]M \tag{14}$$

and U_{sat} is computed as;

$$U_{sat}=K\cdot\mathrm{sat}\left({}^{S}/_{\emptyset}\right) \tag{15}$$

by replace the formulation (15) in (13) the control output can be written as;

$$U=U_{eq}+K.\mathrm{sat}\left({}^{S}/_{\emptyset}\right)=\begin{cases}U_{eq}+K.\mathrm{sgn}(S) & ,|S|\ge\emptyset\\ U_{eq}+K.{}^{S}/_{\emptyset} & ,|S|<\emptyset\end{cases} \tag{16}$$

By (16) and (14) the sliding mode control of PUMA 560 robot manipulator is calculated as;

$$U=\left[M^{-1}(B+C+G)+\dot{S}\right]M+K\cdot\mathrm{sat}\left({}^{S}/_{\emptyset}\right) \tag{17}$$

3. DESIGN ADAPTIVE MIMO FUZZY COMPENSATE FUZZY SLIDING MODE ALGORITHM

Zadeh introduced fuzzy sets in 1965. After 40 years, fuzzy systems have been widely used in different fields, especially on control problems. Fuzzy systems transfer expert knowledge to mathematical models. Fuzzy systems used fuzzy logic to estimate dynamics of our systems. Fuzzy controllers including fuzzy if-then rules are used to control our systems. However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of;

- Input fuzzification (binary-to-fuzzy[B/F]conversion)
- Fuzzy rule base (knowledge base)
- Inference engine
- Output defuzzification (fuzzy-to-binary[F/B]conversion) [30-40].

The basic structure of a fuzzy controller is shown in Figure 1.

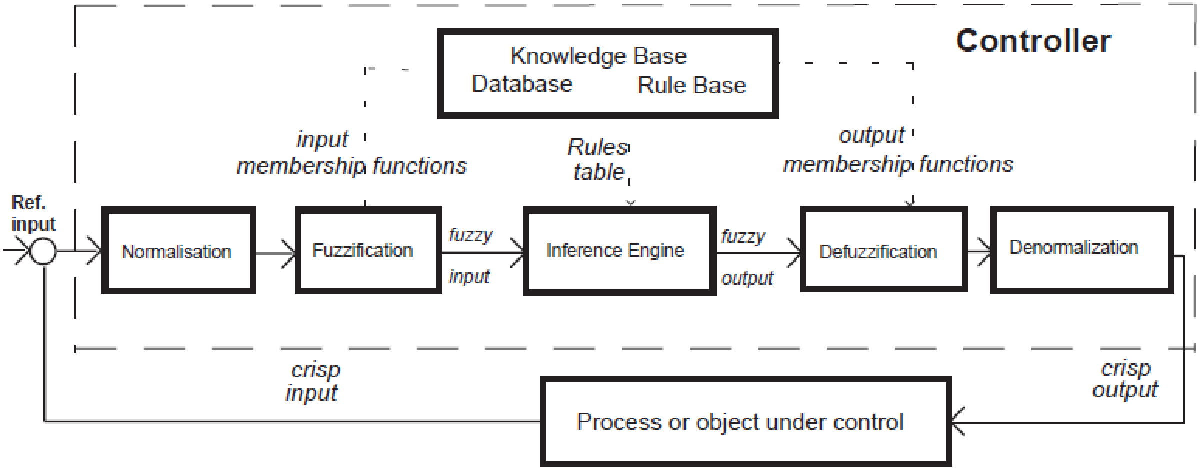


FIGURE 1: Block diagram of a fuzzy controller with details.

Conventional control methods use mathematical models to controls systems. Fuzzy control methods replace the mathematical models with fuzzy if then-rules and fuzzy membership function to controls systems. Both fuzzy and conventional control methods are designed to meet system requirements of stability and convergence. When mathematical models are unknown or partially unknown, fuzzy control models can used fuzzy systems to estimate the unknown models. This is called the model-free approach [31, 35]. Conventional control models can use adaptive control methods to achieve the model-free approach. When system dynamics become more complex, nonlinear systems are difficult to handle by conventional control methods. Fuzzy systems can approximate arbitrary nonlinear systems. In practical problems, systems can be controlled perfectly by expert. Experts provide linguistic description about systems. Conventional control methods cannot design controllers combined with linguistic information. When linguistic information is important for designing controllers, we need to design fuzzy controllers for our systems. Fuzzy control methods are easy to understand for designers. The design process of fuzzy controllers can be simplified with simple mathematical models. Adaptive control uses a learning method to self-learn the parameters of systems. For system whose dynamics are varying, adaptive control can learn the parameters of system dynamics. In traditional adaptive control, we need some information about our system such as the structure of system or the order of the system. In adaptive fuzzy control we can deal with uncertain systems. Due to the linguistic characteristic, adaptive fuzzy controllers behave like operators: adaptively controlling the system under various conditions. Adaptive fuzzy control provides a good tool for making use of expert knowledge to adjust systems. This is important for a complex unknown system with changing dynamics. We divide adaptive fuzzy control into two categories: direct adaptive fuzzy control and indirect adaptive fuzzy control. A direct adaptive fuzzy controller adjusts the parameters of the control input. An indirect adaptive fuzzy controller adjusts the parameters of the control system based on the estimated dynamics of the plant.

We define fuzzy systems as two different types. The firs type of fuzzy systems is given by

$$f(x)=\sum_{l=1}^{M}\theta^{l}\mathcal{E}^{l}(x)=\theta^{T}\mathcal{E}(x) \tag{18}$$

Where $\theta=(\theta^{1},\dots,\theta^{M})^{T},\mathcal{E}(x)=(\mathcal{E}^{1}(x),\dots,\mathcal{E}^{M}(x))^{T}, and\ \mathcal{E}^{l}(x)=:\prod_{i=1}^{n}\frac{\mu_{A_i^l}(x_i)}{}\sum_{l=1}^{M}(\prod_{i=1}^{n}\mu_{A_i^l}(x_i)).$ $\theta^{1},\dots,\theta^{M}$ are adjustable parameters in (18) . $\mu_{A_1^1}(x_1),\dots,\mu_{A_n^m}(x_n)$ are given membership functions whose parameters will not change over time.

The second type of fuzzy systems is given by

$$f(x) = \frac{\sum_{l=1}^{M} \theta^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]} \quad \textbf{(19)}$$

Where θ^l , α_i^l *and* δ_i^l are all adjustable parameters.
From the universal approximation theorem, we know that we can find a fuzzy system to estimate any continuous function. For the first type of fuzzy systems, we can only adjust θ^l in (18). We define $f^{\wedge}(x|\theta)$ as the approximator of the real function $f(x)$.

$$f^{\wedge}(x|\theta) = \theta^T \varepsilon(x) \quad \textbf{(20)}$$

We define θ^* as the values for the minimum error:

$$\theta^* = \arg\min_{\theta \in \Omega} \left[\sup_{x \in U} |f^{\wedge}(x|\theta) - g(x)|\right] \quad \textbf{(21)}$$

Where Ω is a constraint set for θ. For specific x , $sup_{x \in U} |f^{\wedge}(x|\theta^*) - f(x)|$ is the minimum approximation error we can get.

We used the first type of fuzzy systems (18) to estimate the nonlinear system (23) the fuzzy formulation can be write as below;

$$f(x|\theta) = \theta^T \varepsilon(x) = \frac{\sum_{l=1}^{n} \theta^l [\mu_{A^l}(x)]}{\sum_{l=1}^{n} [\mu_{A^l}(x)]} \quad \textbf{(22)}$$

Where $\theta^1, \dots, \theta^n$ are adjusted by an adaptation law. The adaptation law is designed to minimize the parameter errors of $\theta - \theta^*$.
If the dynamic equation of an m-link robotic manipulator is [piltan reference]

$$M(q)\ddot{q} + c(q, \dot{q}) + G(q) = \tau \quad \textbf{(23)}$$

Where $q = [q_1, \dots, q_m]^T$ is an $m \times 1$ vector of joint position, $M(q)$ is an $m \times m$ inertial matrix, $c(q, \dot{q})$ is an $m \times 1$ matrix of Coriolis and centrifugal forces, $G(q)$ is an $m \times 1$ gravity vector and $\tau = [\tau_1, \dots, \tau_m]^T$ is an $m \times 1$ vector of joint torques. This paper proposed an adaptive fuzzy sliding mode control scheme applied to a robotic manipulator. A MIMO (multi-input multi-output) fuzzy system is designed to compensate the uncertainties of the robotic manipulator. The parameters of the fuzzy system are adjusted by adaptation laws.
The tracking error and the sliding surface state are defined as (58-64)

$$e = q - q_d \quad \textbf{(24)}$$

$$s = \dot{e} + \lambda_e \quad \textbf{(25)}$$

We define the reference state as

$$\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e \quad \textbf{(26)}$$

$$\ddot{q}_r = \ddot{q} - \dot{s} = \ddot{q}_d - \lambda \dot{e} \quad \textbf{(27)}$$

The general MIMO if-then rules are given by

$$R^l: if\ x_1\ is\ A_1^l\ , x_2\ is\ A_2^l\ , \dots, x_n\ is\ A_n^l\ , then\ y_1 is\ B_1^l\ , \dots, y_m\ is\ B_m^l \quad \textbf{(28)}$$

Where $l = 1, 2, \dots, M$ are fuzzy if-then rules; $x = (x_1, \dots, x_n)^T$ and $y = (y_1, \dots, y_n)^T$ are the input and output vectors of the fuzzy system. The MIMO fuzzy system is define as

$$f(x) = \ominus^T \varepsilon(x) \quad \textbf{(29)}$$

Where

$$\ominus^T = (\theta_1, \dots, \theta_m)^T = \begin{bmatrix} \theta_1^1, \theta_1^2, \dots, \theta_1^M \\ \theta_2^1, \theta_2^2, \dots, \theta_2^M \\ \vdots \\ \theta_m^1, \theta_m^2, \dots, \theta_m^M \end{bmatrix} \tag{30}$$

$\varepsilon(x) = (\varepsilon^1(x), \dots, \varepsilon^M(x))^T$, $\varepsilon^1(x) = \prod_{i=1}^{n} \mu_{A_i^l}(x_i) / \sum_{l=1}^{M} (\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$, and $\mu_{A_i^l}(x_i)$ is defined in (22). To reduce the number of fuzzy rules, we divide the fuzzy system in to three parts:

$$\begin{aligned} F^1(q, \dot{q}) &= \ominus^{1^T} \varepsilon\,(q, \dot{q}) \\ &= \left[\theta_1^{1^T} \varepsilon\,(q, \dot{q}), \dots, \theta_m^{1^T} \varepsilon\,(q, \dot{q})\right]^T \end{aligned} \tag{31}$$

$$\begin{aligned} F^2(q, \ddot{q}_r) &= \ominus^{2^T} \varepsilon\,(q, \ddot{q}_r) \\ &= \left[\theta_1^{2^T} \varepsilon\,(q, \ddot{q}_r), \dots, \theta_m^{2^T} \varepsilon\,(q, \ddot{q}_r)\right]^T \end{aligned} \tag{32}$$

$$\begin{aligned} F^3(q, \ddot{q}) &= \ominus^{3^T} \varepsilon\,(q, \ddot{q}) \\ &= \left[\theta_1^{3^T} \varepsilon\,(q, \dot{q}), \dots, \theta_m^{3^T} \varepsilon\,(q, \ddot{q})\right]^T \end{aligned} \tag{33}$$

The control input is given by

$$\tau = M^{\wedge} \ddot{q}_r + C_1^{\wedge} \dot{q}_r + G^{\wedge} + F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q}) - K_D s - W sgn(s) \tag{34}$$

Where $M^{\wedge}$, $C_1^{\wedge}$ are the estimations of $M(q)$ and $C_1(q, \dot{q})$; $K_D = diag\,[K_{D1}, \dots, K_{Dm}]$ and $K_{D1}, \dots, K_{Dm}$ are positive constants; $W = diag\,[W_1, \dots, W_m]$ and $W_1, \dots, W_m$ are positive constants. The adaptation law is given by

$$\begin{aligned} \dot{\theta}_j^1 &= -\Gamma_{1j} s_j\, \varepsilon(q, \dot{q}) \\ \dot{\theta}_j^2 &= -\Gamma_{2j} s_j\, \varepsilon(q, \ddot{q}_r) \\ \dot{\theta}_j^3 &= -\Gamma_{3j} s_j\, \varepsilon(q, \ddot{q}) \end{aligned} \tag{35}$$

Where $j = 1, \dots, m$ and $\Gamma_{1j} - \Gamma_{3j}$ are positive diagonal matrices.
The Lyapunov function candidate is presented as

$$V = \frac{1}{2} s^T M s + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \emptyset_j^1 + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \emptyset_j^2 + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{13^T} \emptyset_j^3 \tag{36}$$

Where $\emptyset_j^1 = \emptyset_j^{1^*} - \emptyset_j^1$, $\emptyset_j^2 = \emptyset_j^{2^*} - \emptyset_j^2$ and $\emptyset_j^3 = \emptyset_j^{3^*} - \emptyset_j^3$ we define

$$F(q, \dot{q}, \ddot{q}_r, \ddot{q}) = F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q}) \tag{37}$$

From (23) and (22), we get

$$M(q)\ddot{q} + C_1(q, \dot{q})\dot{q} + G(q) = M^{\wedge} \ddot{q}_r + C_1^{\wedge} \dot{q}_r + G^{\wedge} + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) - K_D s - W sgn(s) \tag{38}$$

Since $\dot{q}_r = \dot{q} - s$ and $\ddot{q}_r = \ddot{q} - \dot{s}$, we get

$$M\dot{s} + (C_1 + K_D)s + W sgn(s) = -\Delta F + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) \tag{39}$$

Then $M\dot{s} + C_1 s$ can be written as

$$M\dot{s} + C_1 s = -\Delta F + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) - K_D s - W sgn(s) \tag{40}$$

Where $\Delta F = \tilde{M}\ddot{q}_r + \tilde{C}_1 \dot{q}_r + \tilde{G}$,$\tilde{M} = M - M^{\wedge}$,$\tilde{C}_1 = C_1 - \tilde{C}_1$ and $\tilde{G} = G - G^{\wedge}$.
The derivative of V is

$$\dot{V} = s^T M\dot{s} + \frac{1}{2}s^T \dot{M}s + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\dot{\emptyset}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\dot{\emptyset}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^{13^T}\dot{\emptyset}_j^3 \quad \textbf{(41)}$$

We know that $s^T M\dot{s} + \frac{1}{2}s^T \dot{M}s = s^T(M\dot{s} + C_1 s)$ from (2.38). Then

$$\dot{V} = -s^T[-K_D s + W sgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\dot{\emptyset}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\dot{\emptyset}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^1 \quad \textbf{(42)}$$

We define the minimum approximation error as

$$\omega = \Delta F - \left[F^1\left(q, \dot{q} \mid \ominus^{1^*}\right) + F^2\left(q, \ddot{q}_r \mid \ominus^{2^*}\right) + F^3\left(q, \ddot{q} \mid \ominus^{3^*}\right)\right] \quad \textbf{(43)}$$

We plug (43) in to (42)

$$\dot{V} = -s^T[-K_D s + W sgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\dot{\emptyset}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\dot{\emptyset}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^{13^T}\dot{\emptyset}_j^3$$

$$= -s^T\left[-K_D s + W sgn(s) + \omega + F^1\left(q, \dot{q} \mid \ominus^{1^*}\right) + F^2\left(q, \ddot{q}_r \mid \ominus^{2^*}\right) + F^3\left(q, \ddot{q} \mid \ominus^{3^*}\right) - F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q})\right] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\dot{\emptyset}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\dot{\emptyset}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^{13^T}\dot{\emptyset}_j^3$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m} s_j \emptyset_j^{1^T}\varepsilon(q, \dot{q}) - \sum_{j=1}^{m} s_j \emptyset_j^{2^T}\varepsilon(q, \ddot{q}_r) - \sum_{j=1}^{m} s_j \emptyset_j^{3^T}\varepsilon(q, \ddot{q}) \cdot \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\dot{\emptyset}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\dot{\emptyset}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^{3^T}\dot{\emptyset}_j^3$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m}\emptyset_j^{1^T}\left(s_j\varepsilon(q, \dot{q}) - \frac{1}{\Gamma_{1j}}\dot{\emptyset}_j^1\right) - \sum_{j=1}^{m}\emptyset_j^{2^T}\left(s_j\varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{2j}}\dot{\emptyset}_j^2\right) - \sum_{j=1}^{m}\emptyset_j^{3^T}\left(s_j\varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{3j}}\dot{\emptyset}_j^3\right)$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m}\emptyset_j^{1^T}\left(s_j\varepsilon(q, \dot{q}) + \frac{1}{\Gamma_{1j}}\dot{\emptyset}_j^1\right) - \sum_{j=1}^{m}\emptyset_j^{2^T}\left(s_j\varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{2j}}\dot{\emptyset}_j^2\right) - \sum_{j=1}^{m}\emptyset_j^{3^T}\left(s_j\varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{3j}}\dot{\emptyset}_j^3\right)$$

The adaptation laws are chosen as (20). Then $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= -s^T K_D s - s^T W sgn(s) - s^T\omega \\ &= -\sum_{j=1}^{m}\left(s_j^2 K_{Dj} + W_j|s_j| + s_j\omega_j\right) \\ &= -\sum_{j=1}^{m}\left[s_j(s_j K_{Dj} + \omega_j) + W_j|s_j|\right] \end{aligned} \quad \textbf{(44)}$$

Since ω_j can be as small as possible, we can find K_{Dj} that $|s_j^2 K_{Dj}| > |\omega_j|(s_j \neq 0)$.
Therefore, we can get $s_j(s_j K_{Dj} + \omega_j) > 0$ for $s_j \neq 0$ and $\dot{V} < 0\ (s \neq 0)$.
Figure 2 is shown the proposed method.

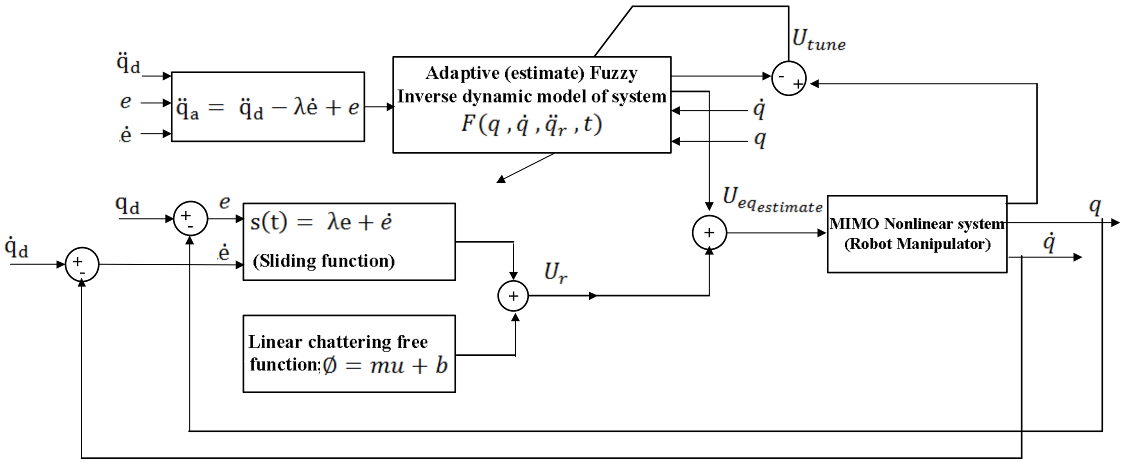


FIGURE 2: Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm

4. APPLICATION: ROBOT MANIPULATOR

Dynamic modelling of robot manipulators is used to describe the behaviour of robot manipulator, design of model based controller, and simulation results. The dynamic modelling describe 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 behaviour of system. It is well known that the equation of an *n-DOF* robot manipulator governed by the following equation [36]; [58-64]:

$$M(q)\ddot{q} + N(q, \dot{q}) + g(x) = \tau \tag{45}$$

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 and $g(x)$ is uncertainty input. This robot manipulator dynamic equation can also be written in a following form:

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) + g(x) \tag{46}$$

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 (45) 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 [6]:

$$\ddot{q} = M^{-1}(q).\{\tau - N(q, \dot{q})\} \tag{47}$$

In this research proposed method is applied to 2 DOF's robot manipulator with the following Where

$$M(q) = \begin{bmatrix} m_1 l^2 + 2m_2 l^2 + 2m_2 l^2 \cos q_2 & m_2 l^2 + m_2 l^2 \cos q_2 \\ m_2 l^2 + m_2 l^2 \cos q_2 & m_2 l^2 \end{bmatrix} \tag{48}$$

$$C(q.\dot{q}) = \begin{bmatrix} -2m_2 l^2 \dot{q}_1 \dot{q}_2 \sin q_2 - m_2 l^2 \dot{q}_2^2 \sin q_2 \\ m_2 l^2 \dot{q}_1^2 \sin q_2 \end{bmatrix} \tag{49}$$

Take the derivative of M with respect to time in (48) and we get

$$\dot{M} = \begin{bmatrix} -2] m_2 l^2 \dot{q}_2 \sin q_2 - m_2 l^2 \dot{q}_2 \sin q_2 \\ -m_2 l^2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \tag{50}$$

From (50) and (48) we get

$$\dot{M} - 2C_1 = \begin{bmatrix} 0 & 2m_2 l^2 \dot{q}_1 \sin q_2 + m_2 l^2 \dot{q}_2 \sin q_2 \\ -2m_2 l^2 \dot{q}_1 \sin q_2 - m_2 l^2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \tag{51}$$

Which is a skew-systemmetric matrix satisfying

$$s^T(\dot{M}-2C_1)s=0 \tag{52}$$

Then $\dot{V}$ becomes

$$\begin{aligned}\dot{V} &= s^T M\dot{s}+\frac{1}{2}s^T\dot{M}s\\ &= s^T(M\dot{s}+C_1 s)\\ &= s^T[-As+\Delta f-K sgn(s)]\\ &= \sum_{i=1}^{2}(s_i[\Delta f_i-K_i sgn(s_i)])-s^T As\end{aligned} \tag{53}$$

For $K_i \geq |\Delta f_i|$, we always get $s_i[\Delta f_i - K_i sgn(s_i)] \leq 0$. We can describe $\dot{V}$ as

$$\dot{V}=\sum_{i=1}^{2}(s_i[\Delta f_i-K_i sgn(s_i)])-s^T As \leq -s^T\ As<0\ \ (s\neq 0) \tag{54}$$

Figure 3 is shown 2 DOF robot manipulator which used in this research.

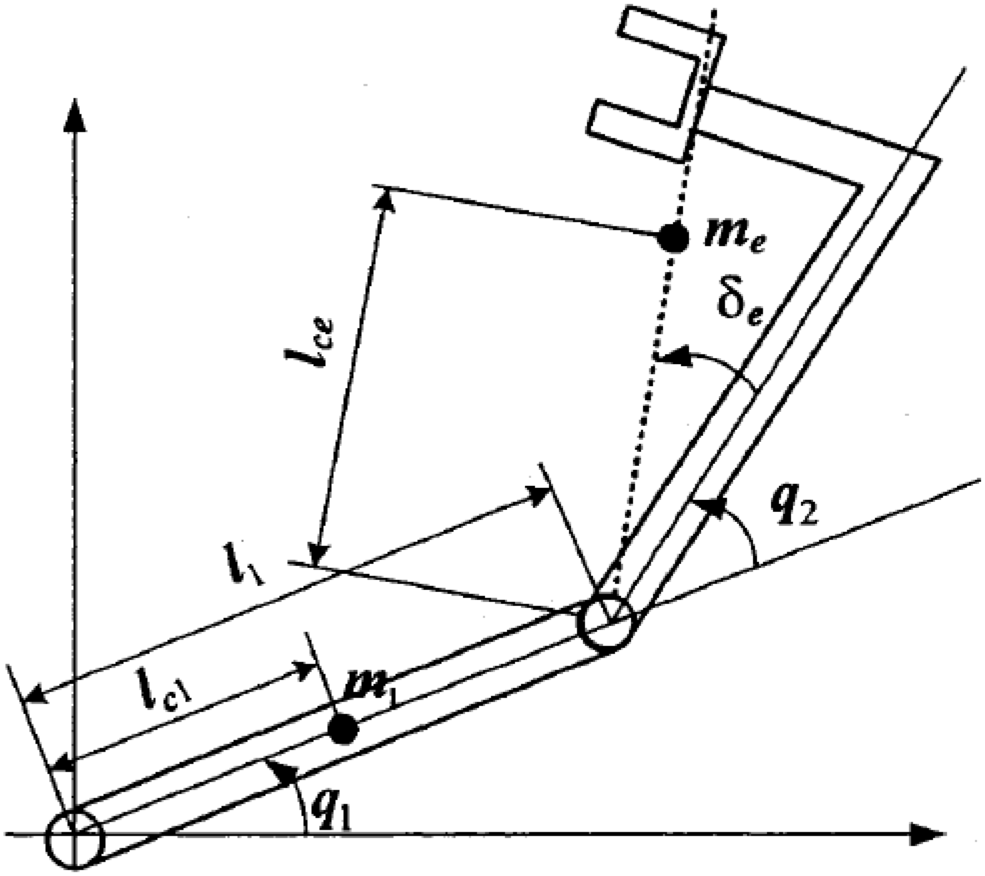


FIGURE 3: A 2-DOF serial robot manipulator

5. SIMULATION RESULT

Sliding mode controller (SMC) and adaptive MIMO fuzzy compensate fuzzy sliding mode controller (AFCFSMC) are implemented in Matlab/Simulink environment. Tracking performance, disturbance rejection and error are compared.

Tracking Performances

From the simulation for first and second trajectory without any disturbance, it was seen that both of controllers have the same performance, because these controllers are adjusted and worked on certain environment. Figure 4 is shown tracking performance in certain system and without external disturbance these two controllers.

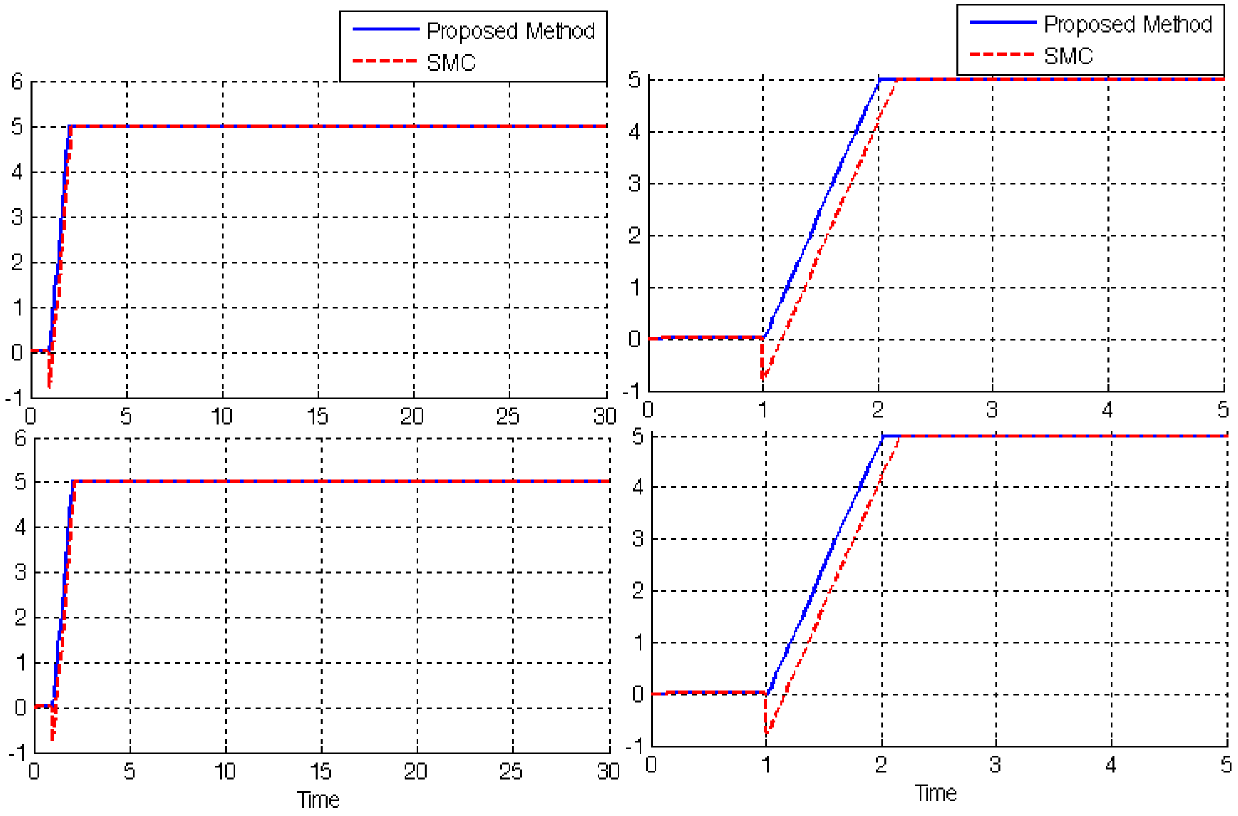


FIGURE 4: SMC Vs. AFCFSMC: applied to 2-DOF serial robot manipulator

By comparing trajectory response in above graph it is found that the AFCFSMC undershoot **(0%)** is lower than SMC **(13.8%)**, although both of them have about the same overshoot.

Disturbance Rejection

Figure 4 has shown the power disturbance elimination in above controllers. The main targets in these controllers are disturbance rejection as well as the other responses. A band limited white noise with predefined of 40% the power of input signal is applied to controllers. It found fairly fluctuations in SMC trajectory responses.

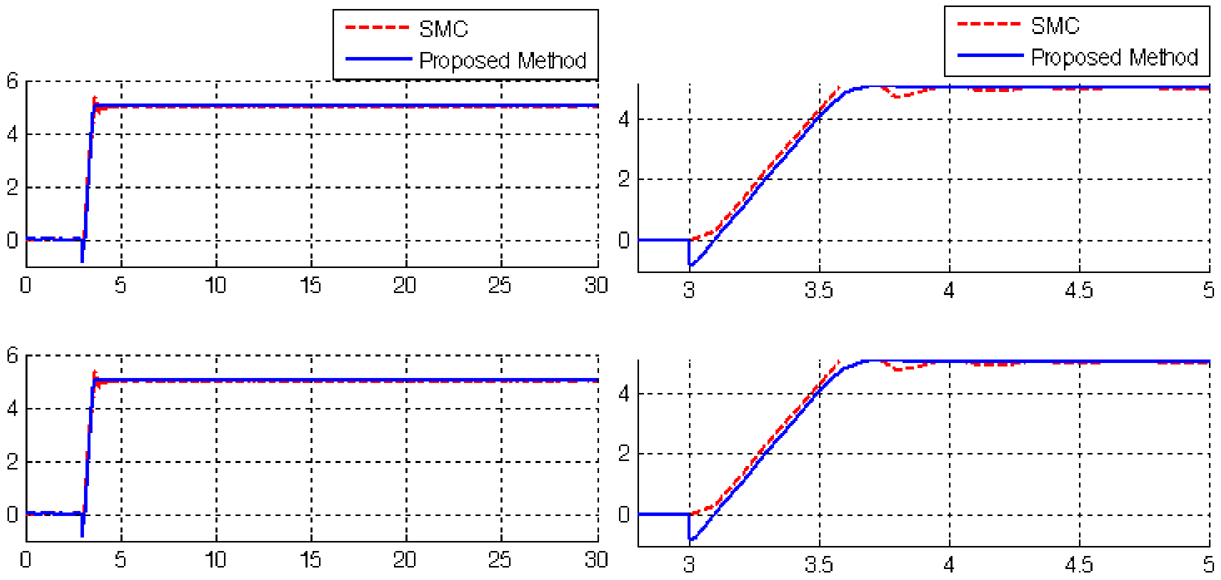


FIGURE 5: SMC Vs. AFCFSMC in presence of uncertainty and external disturbance: applied to 2-DOF serial robot manipulator

Among above graph relating to trajectory following with external disturbance, SMC has fairly fluctuations. By comparing some control parameters such as overshoot and rise time it found that the AFCFSMC's overshoot **(0%)** is lower than SMC's **(6%)**, although both of them have about the same rise time**.**

Calculate Errors

Figure 6 has shown the error disturbance in above controllers. The controllers with no external disturbances have the same error response. By comparing the steady state error and RMS error it found that the AFCFSMC's errors (Steady State error = -0.000007 and RMS error=0.000008) are fairly less than FLC's (Steady State error ≅ **0.0012** and RMS error=**0.0018**), When disturbance is applied to the SMC error is about 23% growth.

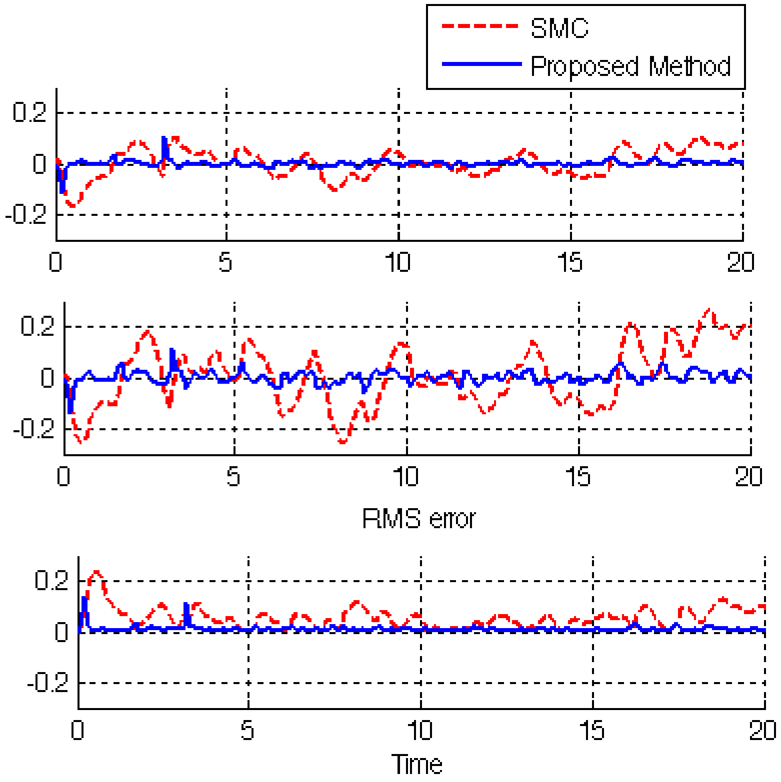


FIGURE 6: SMC Vs. AFCFSMC (error performance): applied to 2-DOF serial robot manipulator

6. CONCLUSIONS

Adaptive fuzzy sliding mode control algorithm for robot manipulators is investigated in this paper. Proposed algorithm utilizes MIMO fuzzy system to estimate the cross-coupling effects in robotic manipulator and gets perfect tracking accuracy. However, the switching control term in the control law causes chattering and there is no methodology to tune the premise part of the fuzzy rules. Proposed algorithm attenuated the chattering problem very well by substituting a fuzzy compensator and saturation function for the switching control term. The number of fuzzy rules is also reduced by abandoning MIMO fuzzy systems and SISO fuzzy systems instead. But we still need to predefine the premise part of the fuzzy rules. The stability and the convergence of this algorithms for the m-link robotic manipulator is proved theoretically using Lyapunov stability theory. Proposed algorithm has predefined adaptation gains in the adaptation laws which are highly related to the performance of our controllers. In this method the tuning part is applied to consequence part, in the case of the m-link robotic manipulator, if we define k_1 membership functions for each input variable, the number of fuzzy rules applied for each joint is $3k_1^{2m}$ and eliminate the chattering.

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] Piltan, F., et al., “Design sliding mode controller for robot manipulator with artificial tunable gain,” Canaidian Journal of pure and applied science, 5 (2): 1573-1579, 2011.

[4] L. Cheng*, et al.*, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo*, et al.*, *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*, et al.*, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* vol. 52, pp. 2085-2102, 2007.

[8] J. Wang*, et al.*, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* vol. 122, pp. 21-30, 2001.

[9] 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

[10] F. Piltan*, et al.*, "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," *Australian Journal of Basic and Applied Sciences, 5(6)*, pp. 509-522, 2011.

[11] Piltan, F., et al., "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[12] Piltan, F., et al., "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.

[13] Piltan, F., et al., "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[14] Piltan, F., et al., "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

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

[16] D. Nguyen-Tuong*, et al.*, "Computed torque control with nonparametric regression models," 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo*, et al.*, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* vol. 76, pp. 212-232, 2002.

[19] K. D. Young*, et al.*, "A control engineer's guide to sliding mode control," 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* 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,* vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics,* 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,* 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,* vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," 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,* vol. 18, pp. 798-810, 2010.

[30] B. Wu*, et al.*, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," 2006, pp. 5-945.

[31] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems,* vol. 90, pp. 111-127, 1997.

[32] L. Reznik, *Fuzzy controllers*: Butterworth-Heinemann, 1997.

[33] J. Zhou and P. Coiffet, "Fuzzy control of robots," 2002, pp. 1357-1364.

[34] S. Banerjee and P. Y. Woo, "Fuzzy logic control of robot manipulator," 2002, pp. 87-88.

[35] K. Kumbla*, et al.*, "Soft computing for autonomous robotic systems," *Computers and Electrical Engineering,* vol. 26, pp. 5-32, 2000.

[36] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Transactions on systems, man and cybernetics,* vol. 20, pp. 404-418, 1990.

[37] R. J. Wai*, et al.*, "Implementation of artificial intelligent control in single-link flexible robot arm," 2003, pp. 1270-1275.

[38] R. J. Wai and M. C. Lee, "Intelligent optimal control of single-link flexible robot arm," *Industrial Electronics, IEEE Transactions on,* vol. 51, pp. 201-220, 2004.

[39] M. B. Menhaj and M. Rouhani, "A novel neuro-based model reference adaptive control for a two link robot arm," 2002, pp. 47-52.

[40] S. Mohan and S. Bhanot, "Comparative study of some adaptive fuzzy algorithms for manipulator control," *International Journal of Computational Intelligence,* vol. 3, pp. 303–311, 2006.

[41] F. Barrero*, et al.*, "Speed control of induction motors using a novel fuzzy sliding-mode structure," *Fuzzy Systems, IEEE Transactions on,* vol. 10, pp. 375-383, 2002.

[42] Y. C. Hsu and H. A. Malki, "Fuzzy variable structure control for MIMO systems," 2002, pp. 280-285.

[43] Y. C. Hsueh*, et al.*, "Self-tuning sliding mode controller design for a class of nonlinear control systems," 2009, pp. 2337-2342.

[44] R. Shahnazi*, et al.*, "Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control," *Energy Conversion, IEEE Transactions on,* vol. 23, pp. 138-147, 2008.

[45] C. C. Chiang and C. H. Wu, "Observer-Based Adaptive Fuzzy Sliding Mode Control of Uncertain Multiple-Input Multiple-Output Nonlinear Systems," 2007, pp. 1-6.

[46] H. Temeltas, "A fuzzy adaptation technique for sliding mode controllers," 2002, pp. 110-115.

[47] C. L. Hwang and S. F. Chao, "A fuzzy-model-based variable structure control for robot arms: theory and experiments," 2005, pp. 5252-5258.

[48] C. G. Lhee*, et al.*, "Sliding mode-like fuzzy logic control with self-tuning the dead zone parameters," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 343-348, 2002.

[49] Lhee. C. G., J. S. Park, H. S. Ahn, and D. H. Kim, "Sliding-Like Fuzzy Logic Control with Self-tuning the Dead Zone Parameters," *IEEE International fuzzy systems conference proceeding*, 1999,pp.544-549.

[50] X. Zhang*, et al.*, "Adaptive sliding mode-like fuzzy logic control for high order nonlinear systems," pp. 788-792.

[51] M. R. Emami*, et al.*, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems,* vol. 6, 1998.

[52] H.K.Lee, K.Fms, "A Study on the Design of Self-Tuning Sliding Mode Fuzzy Controller. Domestic conference," *IEEE Conference*, *1994,* vol. 4, pp. 212-218.

[53] Z. Kovacic and S. Bogdan, *Fuzzy controller design: theory and applications*: CRC/Taylor & Francis, 2006.

[54] F. Y. Hsu and L. C. Fu, "Nonlinear control of robot manipulators using adaptive fuzzy sliding mode control," 2002, pp. 156-161.

[55] R. G. Berstecher*, et al.*, "An adaptive fuzzy sliding-mode controller," *Industrial Electronics, IEEE Transactions on,* vol. 48, pp. 18-31, 2002.

[56] V. Kim, "Independent joint adaptive fuzzy control of robot manipulator," 2002, pp. 645-652.

[57] Y. Wang and T. Chai, "Robust adaptive fuzzy observer design in robot arms," 2005, pp. 857-862.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar*, et al.*, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* 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,* 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,* vol. 19, pp. 362-368, 2004.

[62] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

[64] Piltan, F., et al., “Design Adaptive Fuzzy Robust Controllers for Robot Manipulator,” World Applied Science Journal, 12 (12): 2317-2329, 2011.

Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method

Farzin Piltan SSP.ROBOTIC@yahoo.com
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia
43400 Serdang, Selangor, Malaysia

N. Sulaiman nasri@eng.upm.edu.my
Department of Electrical and Electronic Engineering,
Faculty of Engineering, Universiti Putra Malaysia
43400 Serdang, Selangor, Malaysia

Hajar Nasiri 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

Sadeq Allahdadi 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

Mohammad A. Bairami 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

Refer to the research, design a novel SISO adaptive fuzzy sliding algorithm inverse dynamic like method (NAIDLC) and application to robot manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Regarding to the positive points in inverse dynamic controller, fuzzy logic controller and self tuning fuzzy sliding method, the output has improved. The main objective in this research is analyses and design of the adaptive robust controller based on artificial intelligence and nonlinear control. Robot manipulator is nonlinear, time variant and a number of parameters are uncertain, so design the best controller for this plant is the main target. Although inverse dynamic controller have acceptable performance with known dynamic parameters but regarding to uncertainty, this controller's output has fairly fluctuations. In order to solve this problem this research is focoused on two methodology the first one is design a fuzzy inference system as a estimate nonlinear part of main controller but this method caused to high computation load in fuzzy rule base and the second method is focused on design novel adaptive method to reduce the computation in fuzzy algorithm.

Keywords: Inverse Dynamic Control, Sliding Mode Algorithm, Fuzzy Estimator Sliding Mode Control, Adaptive Method, Adaptive Fuzzy Sliding Mode Inverse Dynamic like Method, Fuzzy Inference System, Robot Manipulator

1. INTRODUCTION

Robot manipulator is 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). Parallel robot manipulators have many legs with some links and joints, where in these robot manipulators base frame has connected to the final frame. Most of industrial robots are serial links, which in serial 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, at last the axis number seven to n use to avoid the bad situation. Dynamic modeling of robot manipulators is used to describe the behavior of robot manipulator, design of model based controller, 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]. The Unimation PUMA 560 serially links robot manipulator was used as a basis, because this robot manipulator widely used in industry and academic. It has a nonlinear and uncertain dynamic parameters serial link 6 degrees of freedom (DOF) robot manipulator. A non linear robust controller design is major subject in this work [1-3].

In modern usage, the word of control has many meanings, this word is usually taken to mean regulate, direct or command. The word feedback plays a vital role in the advance engineering and science. The conceptual frame work in Feed-back theory has developed only since world war II. In the twentieth century, there was a rapid growth in the application of feedback controllers in process industries. According to Ogata, to do the first significant work in three-term or PID controllers which Nicholas Minorsky worked on it by automatic controllers in 1922. In 1934, Stefen Black was invention of the feedback amplifiers to develop the negative feedback amplifier[2]. Negative feedback invited communications engineer Harold Black in 1928 and it occurs when the output is subtracted from the input. Automatic control has played an important role in advance science and engineering and its extreme importance in many industrial applications, i.e., aerospace, mechanical engineering and robotic systems. The first significant work in automatic control was James Watt's centrifugal governor for the speed control in motor engine in eighteenth century[2]. There are several methods for controlling a robot manipulator, which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes[1]. Joint space and operational space control are closed loop controllers which they have been used to provide robustness and rejection of disturbance effect. The main target in joint space controller is to design a feedback controller which the actual motion ($q_a(t)$) and desired motion ($q_d(t)$) as closely as possible. This control problem is classified into two main groups. Firstly, transformation the desired motion $X_d(t)$ to joint variable $q_d(t)$ by inverse kinematics of robot manipulators[6]. This control include simple PD control, PID control, inverse dynamic control, Lyapunov-based control, and passivity based control that explained them in the following section. The main target in operational space controller is to design a feedback controller to allow the actual end-effector motion $X_a(t)$ to track the desired endeffector motion $X_d(t)$. This control methodology requires a greater algorithmic complexity and the inverse kinematics used in the feedback control loop. Direct measurement of operational space variables are very expensive that caused to limitation used of this controller in industrial robot manipulators[4-8]. One of the simplest ways to analysis control of multiple DOF robot manipulators are analyzed each joint separately such as SISO systems and design an independent joint controller for each joint. In this controller, inputs only depends on the velocity and displacement of the corresponding joint and the other parameters between joints such as coupling presented by disturbance input. Joint space controller has many advantages such as one type controllers design for all joints with the same formulation, low cost hardware, and simple structure. Nonlinear control provides a methodology of nonlinear methodologies for nonlinear uncertain systems (e.g., robot manipulators) to have an acceptable performance. These controllers divided into seven groups, namely, inverse dynamic control, computed-torque control, passivity-based control, sliding mode control (variable structure control), artificial intelligence control, lyapunov-based control and adaptive control[9-14]. Inverse dynamic controller (IDC) is a powerful nonlinear controller which it widely used in control robot manipulator. It is based on Feed-back linearization and computes the required arm torques using the nonlinear feedback control law. This controller works very well when all dynamic and physical parameters are known but when the robot manipulator has variation in dynamic parameters, the controller has no acceptable performance[14]. In practice, most of physical systems (e.g., robot manipulators) parameters are unknown or time variant, therefore, inverse dynamic like controller used to compensate dynamic equation of robot manipulator[1, 6]. Research on inverse dynamic controller is significantly growing on robot manipulator application which has been reported in [1, 6, 9, 11, 63-65]. Vivas and Mosquera [63]have proposed a predictive functional controller and compare to inverse dynamic controller for tracking response in uncertain environment. However both controllers have been used in Feed-back linearization, but predictive strategy gives better result as a performance. An inverse dynamic control with non parametric regression models have been presented for a robot arm[64]. This controller also has been problem in uncertain dynamic models. Based on [1, 6]and [63-65] inverse dynamic controller is a significant nonlinear controller to certain systems which it is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. When all dynamic and physical parameters are known the controller works fantastically; practically a large amount of systems have uncertainties and complicated by artificial intelligence or applied on line tuning in this controller decrease this kind of challenge.

Zadeh [31] introduced fuzzy sets in 1965. After 40 years, fuzzy systems have been widely used in different fields, especially on control problems. Fuzzy systems transfer expert knowledge to mathematical models. Fuzzy systems used fuzzy logic to estimate dynamics of proposed systems. Fuzzy controllers including fuzzy if-then rules are used to control proposed systems. Conventional control methods use mathematical models to controls systems [31-40]. Fuzzy control methods replace the mathematical models with fuzzy if then-rules and fuzzy membership function to controls systems. Both fuzzy and conventional control methods are designed to meet system requirements of stability and convergence. When mathematical models are unknown or partially unknown, fuzzy control models can used fuzzy systems to estimate the unknown models. This is called the model-free approach [31-40]. Conventional control models can use adaptive control methods to achieve the model-free approach. When system dynamics become more complex, nonlinear systems are difficult to handle by conventional control methods. From the universal approximation theorem, fuzzy systems can approximate arbitrary nonlinear systems. In practical problems, systems can be controlled perfectly by expert. Experts provide linguistic description about systems. Conventional control methods cannot design controllers combined with linguistic information. When linguistic information is important for designing controllers, we need to design fuzzy controllers for our systems. Fuzzy control methods are easy to understand for designers. The design process of fuzzy controllers can be simplified with simple mathematical models. Research on applied fuzzy logic methodology in inverse dynamic controller (FIDLC) to compensate the unknown system dynamics considerably improves the robot manipulator control process [15-30, 41-47].

Adaptive control uses a learning method to self-learn the parameters of systems. For system whose dynamics are varying, adaptive control can learn the parameters of system dynamics. In traditional adaptive control, we need some information about our system such as the structure of system or the order of the system. In adaptive fuzzy control we can deal with uncertain systems. Due to the linguistic characteristic, adaptive fuzzy controllers behave like operators: adaptively controlling the system under various conditions. Adaptive fuzzy control provides a good tool for making use of expert knowledge to adjust systems. This is important for a complex unknown system with changing dynamics. We divide adaptive fuzzy control into two categories: direct adaptive fuzzy control and indirect adaptive fuzzy control. A direct adaptive fuzzy controller adjusts the parameters of the control input. An indirect adaptive fuzzy controller adjusts the parameters of the control system based on the estimated dynamics of the plant. This research is used fuzzy indirect method to estimate the nonlinear equivalent part in order to used sliding mode fuzzy algorithm to tune and adjust the sliding function (direct adaptive). Research on applied fuzzy logic methodology in sliding mode controller (FSMC) to reduce or eliminate the high frequency oscillation (chattering), to compensate the unknown system dynamics and also to adjust the linear sliding surface slope in pure sliding mode controller considerably improves the robot manipulator control process [41-62]. H.Temeltas [46] has proposed fuzzy adaption techniques for SMC 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. C. L. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode control based on N fuzzy based linear state-space to estimate the uncertainties. A multi-input multi-output FSMC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a robot manipulator [42].

In this research we will highlight the SISO adaptive fuzzy sliding algorithm to on line tuning inverse dynamic like controller with estimates the nonlinear dynamic part derived in the Lyapunov sense. This algorithm will be analyzed and evaluated on robotic manipulators. Section 2, serves as an introduction to the classical inverse dynamic control algorithm and its application to a 3 degree of-freedom robot manipulator, introduced sliding mode controller to design adaptive part, describe the objectives and problem statements. Part 3, introduces and describes the methodology algorithms and proves Lyapunov stability. Section 4 presents the simulation results of this algorithm applied to a 2 degree-of-freedom robot manipulator and the final section is describe the conclusion.

2. OBJECTIVES, PROBLEM STATEMENTS, INVERSE DYNAMIC METHODOLOGY AND SLIDING MODE ALGORITM

When system works with various parameters and hard nonlinearities design linear controller technique is very useful in order to be implemented easily but it has some limitations such as working near the system operating point[2-20]. Inverse dynamic controller is used in wide range areas such as in robotics, in control process, in aerospace applications and in power converters because it has an acceptable control performance and solve some main challenging topics in control such as resistivity to the external disturbance. Even though, this controller is used in wide range areas but, classical inverse dynamic controller has nonlinear part disadvantage which this challenge must be estimated by fuzzy method [20]. Conversely pure FLC works in many areas, it cannot guarantee the basic requirement of stability and acceptable performance[31-40]. Although both inverse dynamic controller and FLC have been applied successfully in many applications but they also have some limitations. Fuzzy estimator is used instead of dynamic uncertaint equation to implement easily and avoid mathematical model base controller. To reduce the effect of uncertainty in proposed method, adaptive fuzzy sliding mode method is applied in inverse dynamic like controller in robot manipulator in order to solve above limitation.

Robot Manipulator Formulation

The equation of a multi degrees of freedom (DOF) robot manipulator is calculated by the following equation[6]:

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

Where τ is $n \times 1$ vector of actuation torque, M (q) is $n \times n$ symmetric and positive define inertia matrix, $N(q,\dot{q})$ is the vector of nonlinearity term, and q is $n \times 1$ position vector. In equation 1 if vector of nonlinearity term derive as Centrifugal, Coriolis and Gravity terms, as a result robot manipulator dynamic equation can also be written as [9-14]:

$$N(q,\dot{q}) = V(q,\dot{q}) + G(q) \tag{2}$$
$$V(q,\dot{q}) = B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 \tag{3}$$
$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) \tag{4}$$

Where,

$B(q)$ is matrix of coriolis torques, $C(q)$ is matrix of centrifugal torque, $[\dot{q}\,\dot{q}]$ is vector of joint velocity that it can give by: $[\dot{q}_1.\dot{q}_2,\dot{q}_1.\dot{q}_3,\dots,\dot{q}_1.\dot{q}_n,\dot{q}_2.\dot{q}_3,\dots]^T$, and $[\dot{q}]^2$ is vector, that it can given by: $[\dot{q}_1^{\,2},\dot{q}_2^{\,2},\dot{q}_3^{\,2},\dots]^T$. In robot manipulator dynamic part the inputs are torques and the outputs are actual displacements, as a result in (4) it can be written as [1, 6, 80-81];

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \tag{5}$$

To implementation (5) the first step is implement the kinetic energy matrix (M) parameters by used of Lagrange's formulation. The second step is implementing the Coriolis and Centrifugal matrix which they can calculate by partial derivatives of kinetic energy. The last step to implement the dynamic equation of robot manipulator is to find the gravity vector by performing the summation of Lagrange's formulation.

The kinetic energy equation (M) is a $n \times n$ symmetric matrix that can be calculated by the following equation;

$$M(\theta) = m_1 J_{v1}^T J_{v1} + J_{\omega 1}^{TC1} I_1 J_{\omega 1} + m_2 J_{v2}^T J_{v2} + J_{\omega 2}^{TC2} I_2 J_{\omega 2} + m_3 J_{v3}^T J_{v3} + J_{\omega 3}^{TC3} I_3 J_{\omega 3} + m_4 J_{v4}^T J_{v4} \\ m_5 J_{v5}^T J_{v5} + J_{\omega 5}^{TC5} I_5 J_{\omega 5} + m_6 J_{v6}^T J_{v6} + J_{\omega 6}^{TC6} I_6 J_{\omega 6} \tag{6}$$

As mentioned above the kinetic energy matrix in n DOF is a $n \times n$ matrix that can be calculated by the following matrix [1, 6]

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & \dots & \dots & \dots & M_{1n} \\ M_{21} & \dots & \dots & \dots & \dots & M_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ M_{n.1} & \dots & \dots & \dots & \dots & M_{n.n} \end{bmatrix} \tag{7}$$

The Coriolis matrix (B) is a $n \times \frac{n(n-1)}{2}$ matrix which calculated as follows;

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & \dots & b_{11n} & b_{123} & \dots & b_{12n} & \dots & \dots & b_{1n-1.n} \\ b_{212} & \dots & \dots & b_{21n} & b_{223} & \dots & \dots & \dots & \dots & b_{2n-1.n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{n.1.2} & \dots & \dots & b_{n.1.n} & \dots & \dots & \dots & \dots & \dots & b_{n.n-1.n} \end{bmatrix} \tag{8}$$

and the Centrifugal matrix (C) is a $n \times n$ matrix;

$$C(q) = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \tag{9}$$

And last the Gravity vector (G) is a $n \times 1$ vector;

$$G(q) = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \tag{10}$$

Inverse Dynamic Control Formulation

Inverse dynamics control is based on cancelling decoupling and nonlinear terms of dynamics of each link. Inverse dynamics control has the form:

$$\tau = M(q).V + B(q)[\dot{q}\dot{q}] + C(q)\,[\dot{q}]^2 + G(q) \tag{11}$$

where typical choices for V are:

$$V = \ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a) \tag{12}$$

or with an integral term

$$v = \ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a) + K_I \int (q_d - q_a)dt \tag{13}$$

where $e = (q_d - q_a)$, the resulting error dynamics is [9, 11, 63-65]

$$\ddot{q}_d + K_v\ddot{e} + K_p\dot{e} + K_I e = 0 \tag{14}$$

where K_p , K_v and K_I are the controller gains. The result schemes is shown in Figure 1, in which two feedback loops, namely, inner loop and outer loop, which an inner loop is a compensate loop and an outer loop is a tracking error loop. However, mostly parameter $N(q,\dot{q})$ is all unknown. So the control cannot be implementation because non linear parameters cannot be determined. In the following section computed torque like controller will be introduced to overcome the problems.

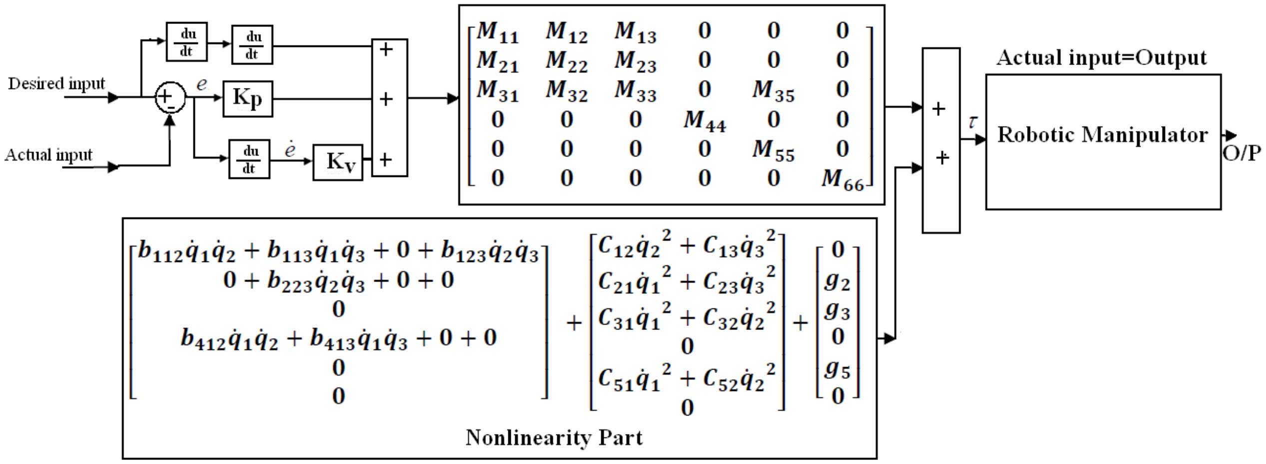


FIGURE 1: Classical inverse dynamic controller: applied to three-link robotic manipulator

The application of proportional-plus-derivative (PD) inverse dynamic controller to control of PUMA robot manipulator introduced in this part.
Suppose that in (13) the nonlinearity term defined by the following term

$$N(q,\dot{q}) = B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q) = \begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2 \\ C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2 \\ C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2 \\ 0 \\ C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{15}$$

Therefore the equation of PD-inverse dynamic controller for control of PUMA robot manipulator is written as the equation of (16);

$$\begin{bmatrix} \widehat{\tau_1} \\ \widehat{\tau_2} \\ \widehat{\tau_3} \\ \widehat{\tau_4} \\ \widehat{\tau_5} \\ \widehat{\tau_6} \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix} \begin{bmatrix} \ddot{q}_{d1} + K_{v1}\dot{e}_1 + K_{p1}e_1 \\ \ddot{q}_{d2} + K_{v2}\dot{e}_2 + K_{p2}e_2 \\ \ddot{q}_{d3} + K_{v3}\dot{e}_3 + K_{p3}e_3 \\ \ddot{q}_{d4} + K_{v4}\dot{e}_4 + K_{p4}e_4 \\ \ddot{q}_{d5} + K_{v5}\dot{e}_5 + K_{p5}e_5 \\ \ddot{q}_{d6} + K_{v6}\dot{e}_6 + K_{p6}e_6 \end{bmatrix} + \begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2 \\ C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2 \\ C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2 \\ 0 \\ C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{16}$$

The controller based on a formulation (16) is related to robot dynamics therefore it has problems in uncertain conditions.

Sliding Mode Control Formulation

We define the tracking error as

$$e = q - q_d \tag{17}$$

Where $q = [q_1, q_2]^T$, $q_d = [q_{1d}, q_{2d}]^T$. The sliding surface is expressed as

$$s = \dot{e} + \lambda e \tag{18}$$

Where $\lambda = diag[\lambda_1, \lambda_2]$, λ_1 and λ_2 are chosen as the bandwidth of the robot controller.
We need to choose τ to satisfy the sufficient condition (9). We define the reference state as

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S}\cdot S = [f - \hat{f} - K\mathrm{sgn}(s)]\cdot S = (f - \hat{f})\cdot S - K|S| \tag{19}$$

$$\dot{q}_e = \dot{q} - s = \dot{q}_d - \lambda e \tag{20}$$

Now we pick the control input τ as

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r - As - Ksgn(s) \tag{21}$$

Where $M^{\wedge}$ and $C_1^{\wedge}$ are the estimations of $M(q)$ and $C_1(q, \dot{q})$; $A = diag[a_1, a_2]$ and $K = diag[k_1, k_2]$ are diagonal positive definite matrices. From (17) and (21), we can get

$$M\dot{s} + (C_1 + A)s = \Delta f - Ksgn(s) \tag{22}$$

Where $\Delta f = \Delta M\ddot{q}_r + \Delta C_1\dot{q}_r$, $\Delta M = M^{\wedge} - M$ and $\Delta C_1 = C_1^{\wedge} - C_1$. We assume that the bound $|\Delta f_i|_{bound}$ *of* $\Delta f_i (i = 1,2)$ is known. We choose K as

$$K_i \geq |\Delta f_i|_{bound} \tag{23}$$

We pick the Lyapunov function candidate to be

$$V = \frac{1}{2}s^T M s \tag{24}$$

Which is a skew-systemmetric matrix satisfying

$$s^T(\dot{M} - 2C_1)s = 0 \tag{25}$$

Then $\dot{V}$ becomes

$$\begin{aligned}\dot{V} &= s^T M\dot{s} + \frac{1}{2}s^T\dot{M}s \\ &= s^T(M\dot{s} + C_1 s) \\ &= s^T[-As + \Delta f - Ksgn(s)] \\ &= \sum_{i=1}^{2}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T As\end{aligned} \tag{26}$$

For $K_i \geq |\Delta f_i|$, we always get $s_i[\Delta f_i - K_i sgn(s_i)] \leq 0$. We can describe $\dot{V}$ as

$$\dot{V} = \sum_{i=1}^{2}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T As \leq -s^T As < 0 \quad (s \neq 0) \tag{27}$$

To attenuate chattering problem, we introduce a saturation function in the control law instead of the sign function in (22). The control law becomes

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r - As - Ksat(s/\Phi) \tag{28}$$

In this classical sliding mode control method, the model of the robotic manipulator is partly unknown. To attenuate chattering, we use the saturation function described in (20). Our control law changes to

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r - As - Ksat(s) \tag{29}$$

The main goal is to design a position controller for robot manipulator with acceptable performances (e.g., trajectory performance, torque performance, disturbance rejection, steady state error and RMS error). Robot manipulator has nonlinear dynamic and uncertain parameters consequently; following objectives have been pursuit in this research:

- To develop an inverse dynamic control and applied to robot manipulator.
- To design and implement a position fuzzy estimator inverse dynamic like controller in order to solve the uncertain nonlinear problems in the pure inverse dynamic control.

- To develop a position adaptive fuzzy sliding mode fuzzy estimator inverse dynamic like controller in order to solve the disturbance rejection and reduce the fuzzy load computation.

Figure 2 is shown the classical sliding mode methodology with linear saturation function to eliminate the chattering.

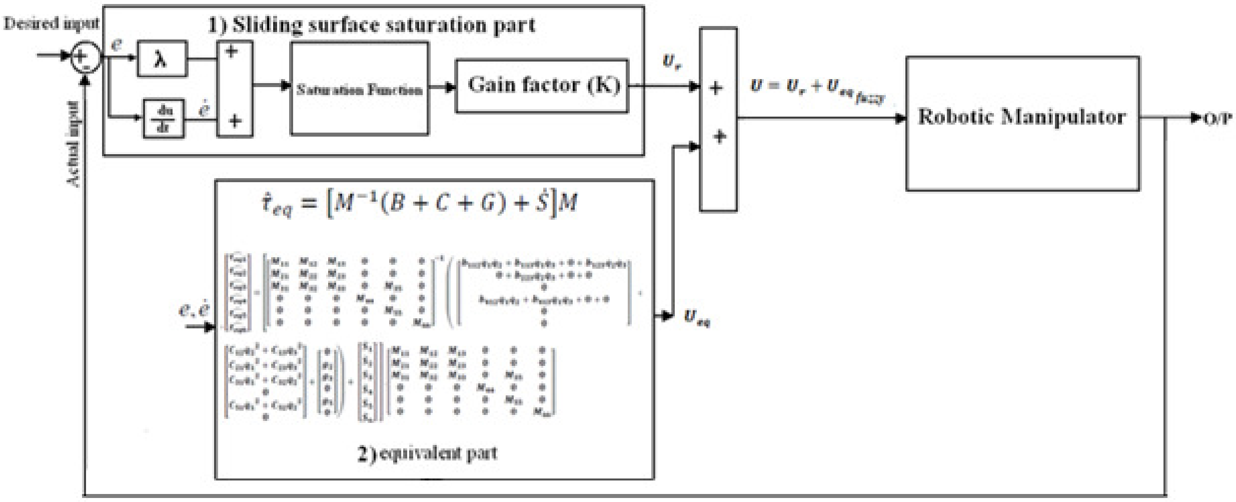


FIGURE 2: Classical sliding mode controller: applied to two-link robotic manipulator

3. METHODOLOGY: DESIGN A NOVEL SISO ADAPTIVE FUZZY SLIDING ALGORITHM INVERSE DYNAMIC LIKE METHOD

First parts are focused on design inverse dynamic like method using fuzzy inference system and estimate or compensate the nonlinear uncertain part. Inverse dynamics control has the form:

$$U = M(q).[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a)] + B(q)[\dot{q}\dot{q}] + C(q)[\dot{q}]^2 + G(q) \quad \textbf{(30)}$$

If nonlinear part is introduced by (31)

$$U_{nonlinear} = B(q)[\dot{q}\dot{q}] + C(q)[\dot{q}]^2 + G(q) \quad \textbf{(31)}$$

However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of;

- Input fuzzification (binary-to-fuzzy[B/F]conversion)
- Fuzzy rule base (knowledge base)
- Inference engine
- Output defuzzification (fuzzy-to-binary[F/B]conversion) [30-40].

The basic structure of a fuzzy controller is shown in Figure 3.

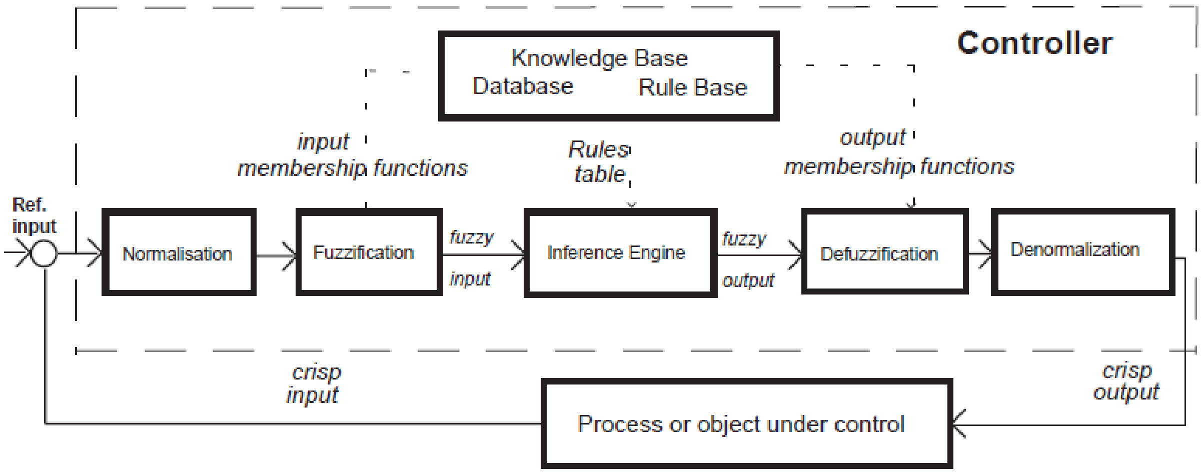


FIGURE 3: Block diagram of a fuzzy controller with details.

The fuzzy system can be defined as below [38-40]

$$f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(S) \tag{32}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{33}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (8) and $\mu_{(xi)}$ is membership function.

error base fuzzy controller can be defined as

$$U_{fuzzy} = \psi(S) \tag{34}$$

Proposed method is used to a SISO fuzzy system which can approximate the residual coupling effect and alleviate the nonlinear part. The robotic manipulator used in this algorithm is defined as below: the tracking error is defined as:

$$e = q - q_d \tag{35}$$

The control input is given by

$$U_{fuzzy} = \hat{B}[\dot{q}_r \dot{q}_r] + \hat{C}\dot{q}_r + \hat{G} - M(q).[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a)]$$

The fuzzy if-then rules for the jth joint of the robotic manipulator are defined as

$$R^{(l)}: if\ e_j\ is\ A_1^l\ ,then\ y\ is\ B_j^l \tag{36}$$

Where $j = 1, \dots, m$ and $l = 1, \dots, M$.
We define K_j by

$$K_j = \frac{\sum_{l=1}^{M} \theta_j^l \left[\mu_{A_j^l}(e_j)\right]}{\sum_{l=1}^{M} \left[\mu_{A_j^l}(e_j)\right]} = \theta_j^T \varepsilon_j(e_j) \tag{37}$$

Where

$$\varepsilon_j(e_j) = \left[\varepsilon_j^1(e_j), \varepsilon_j^2(e_j), \dots, \varepsilon_j^M(e_j)\right]^T, \tag{38}$$

$$\varepsilon_j^l(e_j) = \frac{\sum_{l=1}^{M} \mu_{A_j^l}(e_j)}{\sum_{l=1}^{M} \left[\mu_{A_j^l}(e_j)\right]} \tag{39}$$

The membership function $\mu_{A_j^l}(s_j)$ is a Gaussian membership function defined in bellows:

$$\mu_{A_j^l}(e_j) = \exp\left[-\left(\frac{e_j - \alpha_j^l}{\delta_j^l}\right)^2\right] (j = 1, \dots, m). \tag{40}$$

The fuzzy estimator can be written as follow;

$$\begin{aligned} U &= M(q).[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a)] + B(q)[\dot{q}\dot{q}] + C(q)[\dot{q}]^2 + G(q) \\ &= \hat{B}[\dot{q}_r\dot{q}_r] + \hat{C}\dot{q}_r + \hat{G} - M(q).[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a)] \end{aligned} \tag{41}$$

Since $\dot{q}_r = \dot{q} - e$ and $\ddot{q}_r = \ddot{q} - \dot{e}$ in (41) and (40), we get

$$M\dot{e} + (C_1 + A)s = \Delta F - K \tag{42}$$

Where $\Delta F = \Delta B[\dot{q}_r\dot{q}_r] + \Delta C_1 \dot{q}_r^{\ 2} + \Delta G$, $\Delta M = M^{\wedge} - M$, $\Delta C_1 = C_1^{\wedge} - C_1$ *and* $G = G^{\wedge} - G$, then $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= e^T(M\dot{e} + C_1 e) + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \dot{\emptyset}_j^1 \\ &= -e^T(-Ae + \Delta f - K) + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \dot{\emptyset}_j^1 \\ &= \sum_{j=1}^{m} [e_j(\Delta f_j - K_j)] - e^T A e + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \dot{\emptyset}_j^1 \\ &= \sum_{j=1}^{m} [e_j(\Delta f_j - \theta_j^T \varepsilon_j(e_j))] - e^T A e + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \dot{\emptyset}_j^1 \\ &= \sum_{j=1}^{m} (e_j [\Delta f_j - (\theta_j^*)\varepsilon_j(e_j) + \emptyset_j^T \varepsilon_j(e_j)]) - e^T A e + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \dot{\emptyset}_j^1 \\ &= \sum_{j=1}^{m} (e_j [\Delta f_j - (\theta_j^*)\varepsilon_j(e_j)]) - e^T A e + \sum_{j=1}^{m} \left(\frac{1}{\gamma_{sj}} \emptyset_j^T [\gamma_{sj} e_j \varepsilon_j(e_j) + \dot{\emptyset}_j]\right) \end{aligned}$$

Based on (3) the formulation of proposed fuzzy sliding mode controller can be written as;

$$U = U_{nonlinear_{fuzzy}} + U_r \tag{43}$$

Where $U_{nonlinear_{fuzzy}} = \hat{B}[\dot{q}_r\dot{q}_r] + \hat{C}\dot{q}_r + \hat{G} - M(q).[\ddot{q}_d + K_v(\dot{q}_d - \dot{q}_a) + K_p(q_d - q_a)] + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

Figure 4 is shown the proposed fuzzy inverse dynamic controller.

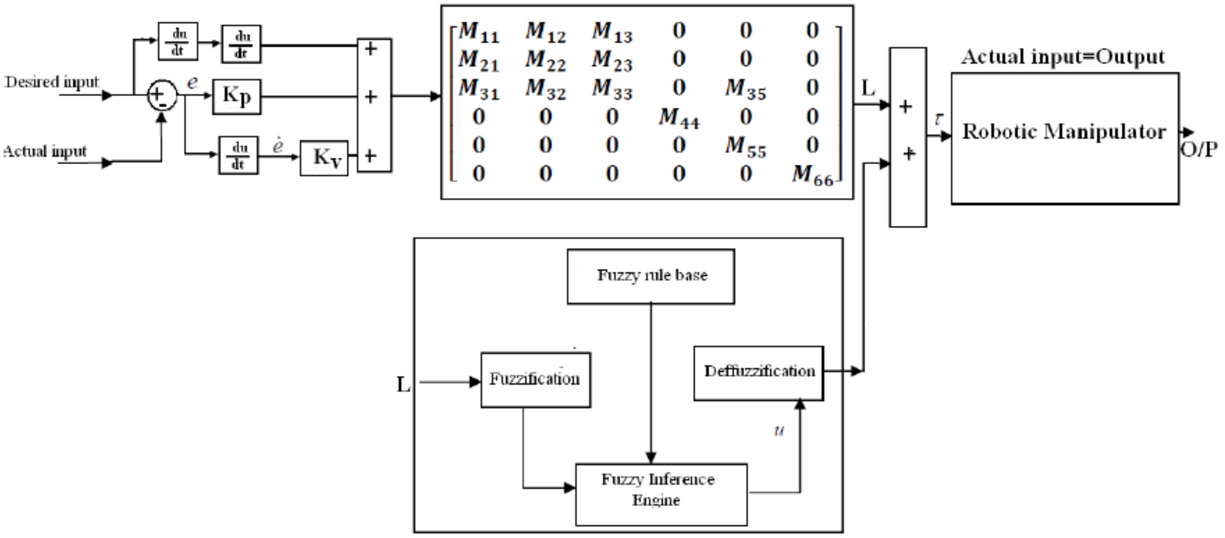


FIGURE 4: Proposed fuzzy estimator inverse dynamic algorithm: applied to robot manipulator

Second part is focuses on design fuzzy sliding mode fuzzy adaptive algorithm, fuzzy algorithm is compensator to estimate nonlinear equivalent part. Adaptive control uses a learning method to self-learn the parameters of systems. For system whose dynamics are varying, adaptive control can learn the parameters of system dynamics. In traditional adaptive control, we need some information about our system such as the structure of system or the order of the system. In adaptive fuzzy control we can deal with uncertain systems. Due to the linguistic characteristic, adaptive fuzzy controllers behave like operators: adaptively controlling the system under various conditions. Adaptive fuzzy control provides a good tool for making use of expert knowledge to adjust systems. This is important for a complex unknown system with changing dynamics. The adaptive fuzzy systems is defined by

$$f(x) = \sum_{l=1}^{M} \theta^l \mathcal{E}^l(x) = \theta^T \mathcal{E}(x) \tag{44}$$

Where $\theta = (\theta^1, \dots, \theta^M)^T, \mathcal{E}(x) = (\mathcal{E}^1(x), \dots, \mathcal{E}^M(x))^T, and\ \mathcal{E}^l(x) =: \prod_{i=1}^{n} \mu_{A_i^l}(x_i) / \sum_{l=1}^{M}(\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$ define in the previous part. $\theta^1, \dots, \theta^M$ are adjustable parameters in (40) . $\mu_{A_1^1}(x_1), \dots, \mu_{A_n^m}(x_n)$ are given membership functions whose parameters will not change over time.
The second type of fuzzy systems is given by

$$f(x) = \frac{\sum_{l=1}^{M} \theta^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l} \right)^2 \right) \right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l} \right)^2 \right) \right]} \tag{45}$$

Where θ^l , α_i^l and δ_i^l are all adjustable parameters.
From the universal approximation theorem, we know that we can find a fuzzy system to estimate any continuous function. For the first type of fuzzy systems, we can only adjust θ^l in (42). We define $f^{\wedge}(x|\theta)$ as the approximator of the real function $f(x)$.

$$f^{\wedge}(x|\theta) = \theta^T \varepsilon(x) \tag{46}$$

We define θ^* as the values for the minimum error:

$$\theta^* = \arg\min_{\theta \in \Omega} \left[\sup_{x \in U} | f^{\wedge}(x|\theta) - g(x) | \right] \tag{47}$$

Where Ω is a constraint set for θ. For specific x ,$sup_{x \in U} | f^{\wedge}(x|\theta^*) - f(x) |$ is the minimum approximation error we can get.
The fuzzy system can be defined as below

$$f(x) = \tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(e, \dot{e}) \tag{48}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{49}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (44) and $\mu_{(xi)}$ is membership function.
error base fuzzy controller can be defined as

$$\tau_{fuzzy} = \psi(e, \dot{e}) \tag{50}$$

According to the formulation sliding function

$$if\ S = 0\ then\ -\dot{e} = \lambda e \tag{51}$$

the fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min \left[Sup_{x \in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \right| \right] \tag{52}$$

Where $\theta^{*}$is the minimum error, $sup_{x\in U}|\sum_{l=1}^{M}\theta^{T}\zeta(x)-\tau_{equ}|$ is the minimum approximation error. The adaptive controller is used to find the minimum errors of $\theta-\theta^{*}$.

suppose K_j is defined as follows

$$K_j=\frac{\sum_{l=1}^{M}\theta_j^l[\mu_A(S_j)]}{\sum_{l=1}^{M}[\mu_A(S_j)]}=\theta_j^T\zeta_j(S_j) \tag{53}$$

Where $\zeta_j(S_j)=[\zeta_j^1(S_j),\zeta_j^2(S_j),\zeta_j^3(S_j),\ldots\ldots,\zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j)=\frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i\mu_{(A)}{}_j^l(S_j)} \tag{54}$$

the adaption low is defined as

$$\dot{\theta}_j=\gamma_{sj}S_j\zeta_j(S_j) \tag{55}$$

where the $\gamma_{sj}$is the positive constant.
According to the formulation (53) and (54) in addition from (50) and (48)

$$M(q)\ddot{q}+V(q,\dot{q})\dot{q}+G(q)=\sum_{l=1}^{M}\theta^{T}\zeta(x)-\lambda S-K \tag{56}$$

The dynamic equation of robot manipulator can be written based on the sliding surface as;

$$M\dot{S}=-VS+M\dot{S}+VS+G-\tau \tag{57}$$

It is supposed that

$$S^T(\dot{M}-2V)S=0 \tag{58}$$

it can be shown that

$$M\dot{S}+(V+\lambda)S=\Delta f-K \tag{59}$$

where $\Delta f=[M(q)\ddot{q}+V(q,\dot{q})\dot{q}+G(q)]-\sum_{l=1}^{M}\theta^{T}\zeta(x)$
as a result $\dot{V}$ is became

$$\begin{aligned}\dot{V}&=\frac{1}{2}S^T\dot{M}S-S^TVS+\sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j\\&=S^T(-\lambda S+\Delta f-K)+\sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j\\&=\sum_{j=1}^{m}[S_j(\Delta f_j-K_j)]-S^T\lambda S+\sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j\\&=\sum_{j=1}^{m}[S_j(\Delta f_j-\theta_j^T\zeta_j(S_j))]-S^T\lambda S+\sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j\\&=\sum_{j=1}^{m}[S_j(\Delta f_j-(\theta_j^*)^T\zeta_j(S_j)+\phi_j^T\zeta_j(S_j))]-S^T\lambda S+\sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j\\&=\sum_{j=1}^{m}[S_j(\Delta f_j-\left((\theta_j^*)^T\zeta_j(S_j)\right)]-S^T\lambda S)]+\sum(\frac{1}{\gamma_{sj}}\phi_j^T[\gamma_{sj}\zeta_j(S_j)S_j+\dot{\phi}_j])\end{aligned}$$

where $\dot{\theta}_j=\gamma_{sj}S_j\zeta_j(S_j)$ is adaption law, $\dot{\phi}_j=-\dot{\theta}_j=-\gamma_{sj}S_j\zeta_j(S_j)$,
consequently $\dot{V}$ can be considered by

$$\dot{V}=\sum_{j=1}^{m}[S_j\Delta f_j-\left((\theta_j^*)^T\zeta_j(S_j)\right)]-S^T\lambda S \tag{60}$$

the minimum error can be defined by

$$e_{mj} = \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j) \right) \tag{61}$$

$\dot{V}$ is intended as follows

$$\begin{aligned} \dot{V} &= \sum_{j=1}^{m} [S_j e_{mj}] - S^T \lambda S \\ &\le \textstyle\sum_{j=1}^{m} |S_j||e_{mj}| - S^T \lambda S \\ &= \sum_{j=1}^{m} |S_j||e_{mj}| - \lambda_j {S_j}^2 \\ &= \sum_{j=1}^{m} |S_j| \left(|e_{mj}| - \lambda_j S_j \right) \end{aligned} \tag{62}$$

For continuous function $g(x)$, and suppose $\varepsilon > 0$ it is defined the fuzzy logic system in form of (46) such that

$$Sup_{x \in U} |f(x) - g(x)| < \epsilon \tag{63}$$

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

$$if \ \lambda_j = \alpha \quad that \quad \alpha |S_j| > e_{mj} \ (S_j \ne 0) \qquad then \quad \dot{V} < 0 \ for \ (S_j \ne 0) \tag{64}$$

Figure 5 is shown the proposed method which it has an acceptable performance.

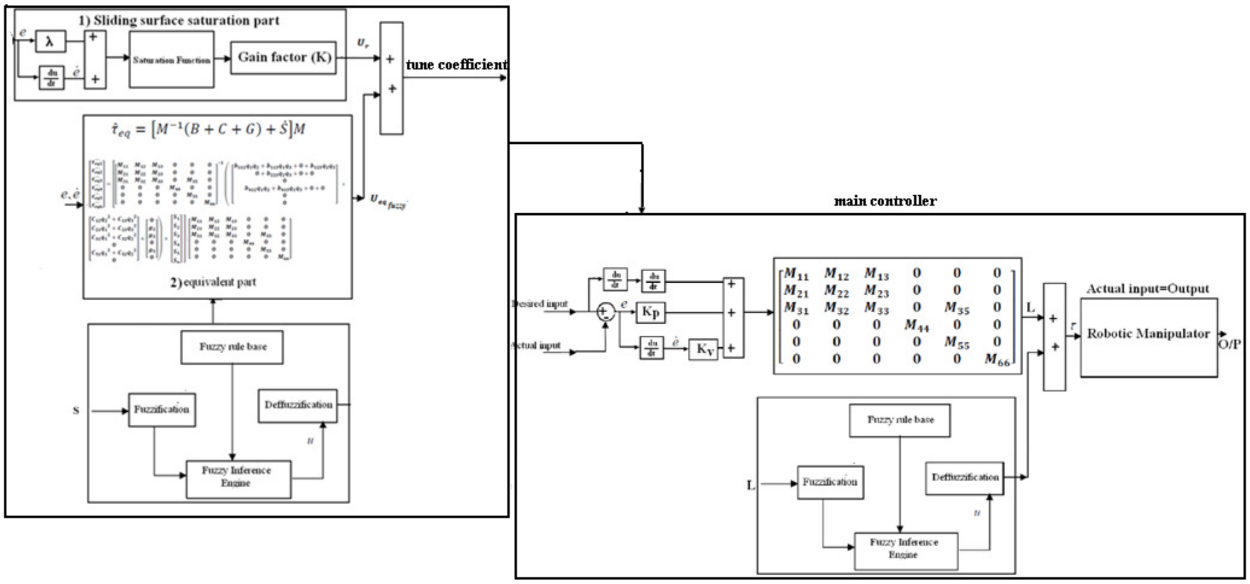


Figure 5: Proposed adaptive fuzzy sliding mode algorithm applied to inverse dynamic like controller: applied to robot manipulator

4. SIMULATION RESULTS

Inverse dynamic controller and SISO proposed adaptive inverse dynamic like controller were tested to ramp response trajectory. This simulation applied to three degrees of freedom robot arm therefore 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. It is noted that, these systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude. This type of noise is used to external disturbance in continuous and hybrid systems.

Tracking Performances

Figure 6 is shown tracking performance for first, second and third link in inverse dynamic control and adaptive inverse dynamic like control without disturbance for ramp trajectories. By comparing ramp response trajectory without disturbance in inverse dynamic controller and adaptive inverse dynamic like controller it is found that the inverse dynamic controller’s overshoot (1%) is higher than adaptive inverse dynamic like controller (0%), although almost both of them have about the same rise time.

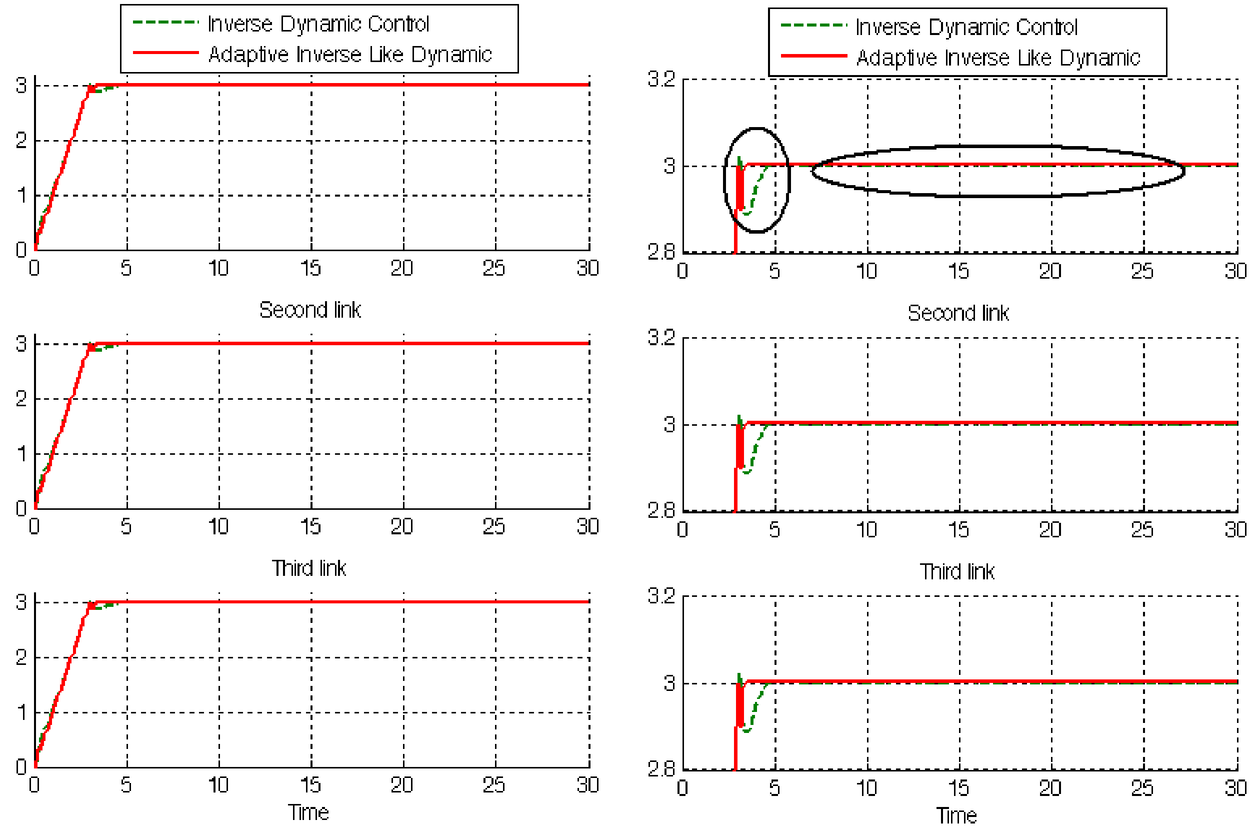


FIGURE 6: Inverse dynamic control Vs. adaptive inverse dynamic like controller trajectory: applied to robot manipulator.

Disturbance Rejection

Figure 7 has shown the power disturbance elimination in inverse dynamic control and adaptive inverse dynamic like control. The main target in these controllers is disturbance rejection as well as reduces the oscillation. A band limited white noise with predefined of 40% the power of input signal is applied to above controllers. It found fairly fluctuations in inverse dynamic control trajectory responses.

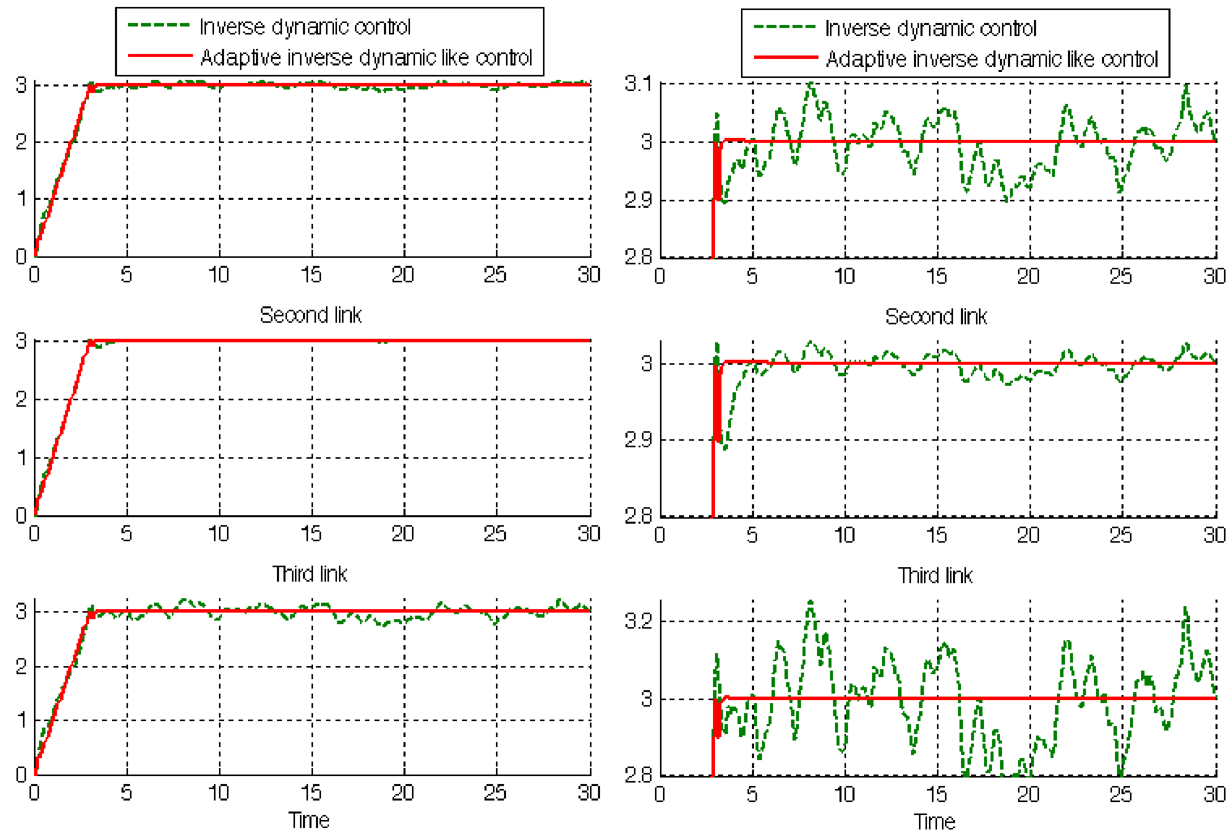


FIGURE 7: Inverse dynamic controller Vs inverse dynamic like controller trajectory with external disturbance: applied to robot manipulator

Among above graph relating to trajectory following with external disturbance, inverse dynamic controller has fairly fluctuations. By comparing some control parameters such as overshoot and rise time it found that the inverse dynamic control's overshoot (10%) is higher than adaptive inverse dynamic like controller (0%).

Error Calculation

Figure 8 and Table 1 are shown error performance in inverse dynamic controller and adaptive inverse dynamic like controller in presence of external disturbance. Inverse dynamic controller has oscillation in tracking which causes instability. As it is obvious in Table 2 the integral of absolute error is calculated to compare between classical method and proposed adaptive classic combined by artificial intelligence method. Figure 8 is shown steady state and RMS error in inverse dynamic control and adaptive inverse dynamic like control in presence of external disturbance.

TABLE 1: RMS Error Rate of Presented controllers

RMS Error Rate	Inverse dynamic controller	Adaptive inverse dynamic like controller
Without Noise	**1.8e-3**	**1e-4**
With Noise	**0.012**	**1.3e-4**

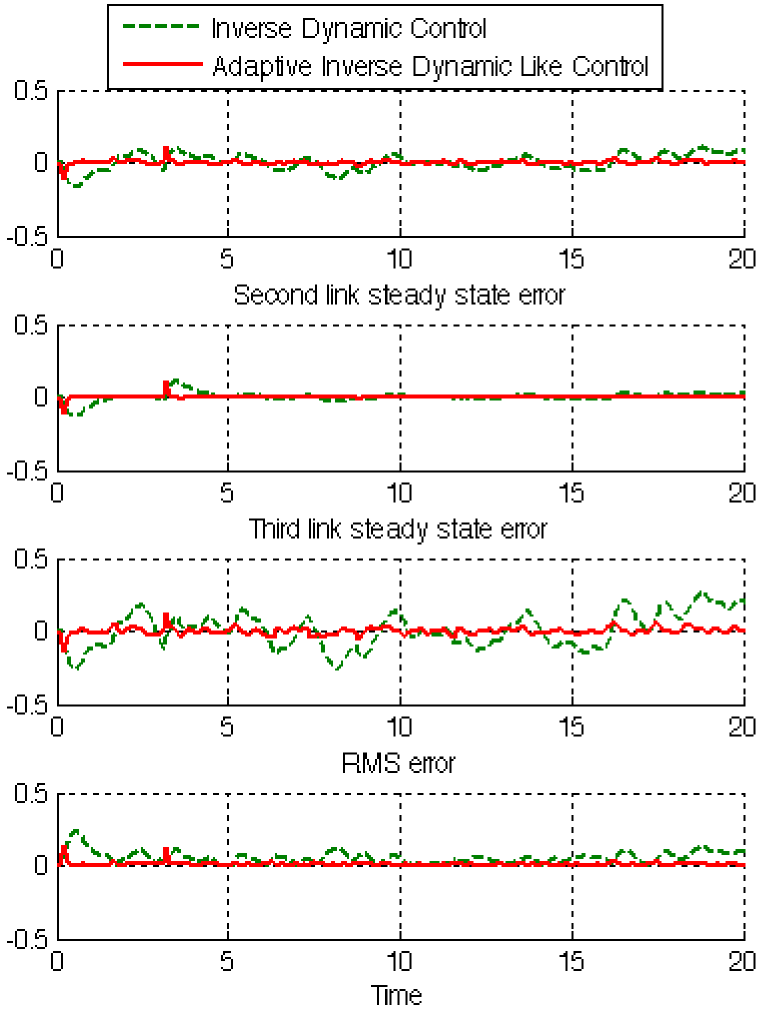


FIGURE 8: Adaptive inverse dynamic like controller Vs. inverse dynamic controller error performance with external disturbance: applied to robot manipulator

In these methods if integration absolute error (IAE) is defined by (75), table 2 is shown comparison between these two methods.

$$IAE = \int_0^\infty |e(t)|\, dt \qquad \textbf{(65)}$$

TABLE 2: Calculate IAE

Method	**Traditional IDC**	**Fuzzy Estimator IDC**	**AIDLC**
IAE	**490.1**	**411**	**202**

5. CONCLUSIONS

In this research, a novel SISO adaptive fuzzy sliding algorithm inverse dynamic like method design and application to robotic manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Each method by adding to the previous controller has covered negative points. The main target in this research is analyses and design of adaptive inverse dynamic like controller for robot manipulator to reach an acceptable performance. Robot manipulators are nonlinear and a number of parameters are uncertain, this research focuses on implement these controllers as accurate as possible using both analytical and empirical paradigms and the advantages and disadvantages of each one is presented through a comparative study, inverse dynamic controller and adaptive inverse dynamic like controller is used to selected the best controller for the industrial manipulator. In the first part studies about inverse dynamic controller show that: although this controller has acceptable performance with known dynamic parameters such as stability and robustness but there are two important disadvantages as below: oscillation and mathematical nonlinear dynamic in controller part. Second step focuses on applied fuzzy inference method as estimate in inverse dynamic controller to solve the dynamic nonlinear part problems in classical inverse dynamic controller. This controller works very well in certain and sometimes in uncertain environment but it has high computation in uncertain area. The system performance in inverse dynamic control and inverse dynamic like controller are sensitive to the controller gain, area and external disturbance. Therefore, compute the optimum value of controller gain for a system is the third important challenge work. This problem has solved by adjusting controller gain of the adaptive method continuously in real-time. In this way, the overall system performance has improved with respect to the classical inverse dynamic controller. This controller solved oscillation as well as mathematical nonlinear dynamic part by applied fuzzy supervisory estimated method in fuzzy inverse dynamic like controller and tuning the controller gain. By comparing between adaptive inverse dynamic like controller and inverse dynamic like controller, found that adaptive fuzzy inverse dynamic like controller has steadily stabilised in output response (e.g., disturbance rejection) but inverse dynamic controller has slight oscillation in the presence of 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] Piltan, F., et al., “Design sliding mode controller for robot manipulator with artificial tunable gain,” Canaidian Journal of pure and applied science, 5 (2): 1573-1579, 2011.

[4] L. Cheng*, et al.*, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo*, et al.*, *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*, et al.*, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* vol. 52, pp. 2085-2102, 2007.

[8] J. Wang*, et al.*, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* vol. 122, pp. 21-30, 2001.

[9] Farzin Piltan, A. R. Salehi and Nasri B Sulaiman.,” Design artificial robust control of second order system based on adaptive fuzzy gain scheduling,” International Journal of Robotics and Automation (IJRA), 2 (4), 2011

[10] F. Piltan*, et al.*, "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," *Australian Journal of Basic and Applied Sciences, 5(6)*, pp. 509-522, 2011.

[11] Piltan, F., et al., “Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain,” International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[12] Piltan, F., et al., “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.

[13] Piltan, F., et al., “Design of FPGA based sliding mode controller for robot manipulator,” International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[14] Piltan, F., et al., “A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator,” World Applied Science Journal, 12 (12): 2330-2336, 2011.

[15] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

[16] Piltan, F., et al., “Design Adaptive Fuzzy Robust Controllers for Robot Manipulator,” World Applied Science Journal, 12 (12): 2317-2329, 2011.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo*, et al.*, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* vol. 76, pp. 212-232, 2002.

[19] K. D. Young*, et al.*, "A control engineer's guide to sliding mode control," 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* 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,* vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics,* 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,* 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,* vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," 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,* vol. 18, pp. 798-810, 2010.

[30] B. Wu*, et al.*, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," 2006, pp. 5-945.

[31] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems,* vol. 90, pp. 111-127, 1997.

[32] L. Reznik, *Fuzzy controllers*: Butterworth-Heinemann, 1997.

[33] J. Zhou and P. Coiffet, "Fuzzy control of robots," 2002, pp. 1357-1364.

[34] S. Banerjee and P. Y. Woo, "Fuzzy logic control of robot manipulator," 2002, pp. 87-88.

[35] K. Kumbla*, et al.*, "Soft computing for autonomous robotic systems," *Computers and Electrical Engineering,* vol. 26, pp. 5-32, 2000.

[36] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Transactions on systems, man and cybernetics,* vol. 20, pp. 404-418, 1990.

[37] R. J. Wai*, et al.*, "Implementation of artificial intelligent control in single-link flexible robot arm," 2003, pp. 1270-1275.

[38] R. J. Wai and M. C. Lee, "Intelligent optimal control of single-link flexible robot arm," *Industrial Electronics, IEEE Transactions on,* vol. 51, pp. 201-220, 2004.

[39] M. B. Menhaj and M. Rouhani, "A novel neuro-based model reference adaptive control for a two link robot arm," 2002, pp. 47-52.

[40] S. Mohan and S. Bhanot, "Comparative study of some adaptive fuzzy algorithms for manipulator control," *International Journal of Computational Intelligence,* vol. 3, pp. 303–311, 2006.

[41] F. Barrero*, et al.*, "Speed control of induction motors using a novel fuzzy sliding-mode structure," *Fuzzy Systems, IEEE Transactions on,* vol. 10, pp. 375-383, 2002.

[42] Y. C. Hsu and H. A. Malki, "Fuzzy variable structure control for MIMO systems," 2002, pp. 280-285.

[43] Y. C. Hsueh*, et al.*, "Self-tuning sliding mode controller design for a class of nonlinear control systems," 2009, pp. 2337-2342.

[44] R. Shahnazi*, et al.*, "Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control," *Energy Conversion, IEEE Transactions on,* vol. 23, pp. 138-147, 2008.

[45] C. C. Chiang and C. H. Wu, "Observer-Based Adaptive Fuzzy Sliding Mode Control of Uncertain Multiple-Input Multiple-Output Nonlinear Systems," 2007, pp. 1-6.

[46] H. Temeltas, "A fuzzy adaptation technique for sliding mode controllers," 2002, pp. 110-115.

[47] C. L. Hwang and S. F. Chao, "A fuzzy-model-based variable structure control for robot arms: theory and experiments," 2005, pp. 5252-5258.

[48] C. G. Lhee*, et al.*, "Sliding mode-like fuzzy logic control with self-tuning the dead zone parameters," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 343-348, 2002.

[49] Lhee. C. G., J. S. Park, H. S. Ahn, and D. H. Kim, "Sliding-Like Fuzzy Logic Control with Self-tuning the Dead Zone Parameters," *IEEE International fuzzy systems conference proceeding*, 1999,pp.544-549.

[50] X. Zhang*, et al.*, "Adaptive sliding mode-like fuzzy logic control for high order nonlinear systems," pp. 788-792.

[51] M. R. Emami*, et al.*, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems,* vol. 6, 1998.

[52] H.K.Lee, K.Fms, "A Study on the Design of Self-Tuning Sliding Mode Fuzzy Controller. Domestic conference," *IEEE Conference*, *1994,* vol. 4, pp. 212-218.

[53] Z. Kovacic and S. Bogdan, *Fuzzy controller design: theory and applications*: CRC/Taylor & Francis, 2006.

[54] F. Y. Hsu and L. C. Fu, "Nonlinear control of robot manipulators using adaptive fuzzy sliding mode control," 2002, pp. 156-161.

[55] R. G. Berstecher*, et al.*, "An adaptive fuzzy sliding-mode controller," *Industrial Electronics, IEEE Transactions on,* vol. 48, pp. 18-31, 2002.

[56] V. Kim, "Independent joint adaptive fuzzy control of robot manipulator," 2002, pp. 645-652.

[57] Y. Wang and T. Chai, "Robust adaptive fuzzy observer design in robot arms," 2005, pp. 857-862.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar*, et al.*, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* 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,* 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,* vol. 19, pp. 362-368, 2004.

[62] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

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

[64] D. Nguyen-Tuong*, et al.*, "Computed torque control with nonparametric regression models," 2008, pp. 212-217.

[65] Farzin Piltan*, et al.*, "Design of model free adaptive fuzzy computed torque controller for a nonlinear second order system," International Journal of Robotics and Automation (IJRA), 2(4), 2011.

Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator

Farzin Piltan SSP.ROBOTIC@yahoo.com
Department of Electrical and Electronic
Engineering, Faculty of Engineering,Universiti
Putra Malaysia 43400 Serdang, Selangor, Malaysia

N. Sulaiman nasri@eng.upm.edu.my
Department of Electrical and Electronic
Engineering, Faculty of Engineering,Universiti
Putra Malaysia 43400 Serdang, Selangor, Malaysia

Samaneh Roosta 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

Atefeh Gavahian 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

Samira Soltani 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

This paper expands a fuzzy sliding mode based position controller whose sliding function is on-line tuned by backstepping methodology. The main goal is to guarantee acceptable position trajectories tracking between the robot manipulator end-effector and the input desired position. The fuzzy controller in proposed fuzzy sliding mode controller is based on Mamdani's fuzzy inference system (FIS) and it has one input and one output. The input represents the function between sliding function, error and the rate of error. The second input is the angle formed by the straight line defined with the orientation of the robot, and the straight line that connects the robot with the reference cart. The outputs represent angular position, velocity and acceleration commands, respectively. The backstepping methodology is on-line tune the sliding function based on self tuning methodology. The performance of the backstepping on-line tune fuzzy sliding mode controller (TBsFSMC) is validated through comparison with previously developed robot manipulator position controller based on adaptive fuzzy sliding mode control theory (AFSMC). Simulation results signify good performance of position tracking in presence of uncertainty and external disturbance.

Keywords: Fuzzy Sliding Mode Controller, Backstepping Controller, Robot Manipulator, Backstepping on-Line Tune Fuzzy Sliding Mode Controller

1. INTRODUCTION

In the recent years robot manipulators not only have been used in manufacturing but also used in vast area such as medical area and working in International Space Station. Control methodologies and the mechanical design of robot manipulators have started in the last two decades and the most of researchers work in these methodologies [1]. PUMA 560 robot manipulator is an articulated 6 DOF serial robot manipulator. This robot is widely used in industrial and academic area and also dynamic parameters have been identified and documented in the literature [2-3].There are several methods for controlling a robot manipulator (e.g., PUMA robot manipulator), which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes[1]. Both of these controllers are closed loop which they have been used to provide robustness and rejection of disturbance effect. One of the simplest ways to analysis control of multiple DOF's robot manipulators are analyzed each joint separately such as SISO systems and design an independent joint controller for each joint. In this controller, inputs only depends on the velocity and displacement of the corresponding joint and the other parameters between joints such as coupling presented by disturbance input. Joint space controller has many advantages such as one type controllers design for all joints with the same formulation, low cost hardware, and simple structure [1, 4].

Sliding mode controller (SMC) is one of the influential nonlinear controllers in certain and uncertain systems which are used to present a methodical solution for two main important controllers' challenges, which named: stability and robustness. Conversely, this controller is used in different applications; sliding mode controller has subsequent drawbacks , the first one is chattering phenomenon, which it is caused to some problems such as saturation and heat for mechanical parts of robot manipulators or drivers and the second one is nonlinear equivalent dynamic formulation in uncertain systems[1, 5-12]. In order to solve the chattering in the systems output, boundary layer method should be applied so beginning able to recommended model in the main motivation which in this method the basic idea is replace the discontinuous method by saturation (linear) method with small neighborhood of the switching surface. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[13]. Slotine has presented sliding mode with boundary layer to improve the industry application [14]. R. 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[15]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[16]. As mentioned [16]sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most exceptional stability and robustness. Conversely above method has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. To eliminate the above problems control researchers have applied artificial intelligence method (e.g., fuzzy logic) in nonlinear robust controller (e.g., sliding mode controller) besides this technique is very useful in order to implement easily. Estimated uncertainty method is used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [17-20]. Wu et al. [22] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Elmali et al. [19]and Li and Xu [21]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering. This method was tested for the tracking control of the first two links of a SCARA type HITACHI robot. In this technique, digital controller is used to increase the system's response quality. Conversely this method has the following advantages; increasing the controller's response speed and reducing dependence on dynamic system model by on-line control, the main disadvantage are chattering phenomenon and need to improve the performance.

In recent years, artificial intelligence theory has been used in sliding mode control systems. Neural network, fuzzy logic and neuro-fuzzy are synergically combined with nonlinear classical controller and used in nonlinear, time variant and uncertainty plant (e.g., robot manipulator). Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control nonlinear, uncertain and noisy systems. This method is free of some model-based techniques as in classical controllers. As mentioned that fuzzy logic application is not only limited to the modelling of nonlinear systems [23-28]but also this method can help engineers to design easier controller. The main reasons to use fuzzy logic technology are able to give approximate recommended solution for unclear and complicated systems to easy understanding and flexible. Fuzzy logic provides a method which is able to model a controller for nonlinear plant with a set of IF-THEN rules, or it can identify the control actions and describe them by using fuzzy rules. The applications of artificial intelligence such as neural networks and fuzzy logic in modelling and control are significantly growing especially in recent years. For instance, the applications of artificial intelligence, neural networks and fuzzy logic, on robot arm control have reported in [29-31]. Wai et al. [29-30]have proposed a fuzzy neural network (FNN) optimal control system to learn a nonlinear function in the optimal control law. This controller is divided into three main groups: arterial intelligence controller (fuzzy neural network) which it is used to compensate the system's nonlinearity and improves by adaptive method, robust controller to reduce the error and optimal controller which is the main part of this controller. Mohan and Bhanot [32] have addressed comparative study between some adaptive fuzzy, and a new hybrid fuzzy control algorithm for manipulator control. They found that self-organizing fuzzy logic controller and proposed hybrid integrator fuzzy give the best performance as well as simple structure. Research on combinations of fuzzy logic systems with sliding mode method is significantly growing as nonlinear control applications. For instance, the applications of fuzzy logic on sliding mode controller have reported in [11, 33-37].

Research on applied fuzzy logic methodology in sliding mode controller (FSMC) to reduce or eliminate the high frequency oscillation (chattering), to compensate the unknown system dynamics and also to adjust the linear sliding surface slope in pure sliding mode controller considerably improves the robot manipulator control process [34-35]. H.Temeltas [38] has proposed fuzzy adaption techniques for SMC 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. C. L. Hwang *et al.* [39]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode control based on N fuzzy based linear state-space to estimate the uncertainties. A multi-input multi-output FSMC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a robot manipulator [34]. Investigation on applied sliding mode methodology in fuzzy logic controller (SMFC) to reduce the fuzzy rules and refine the stability of close loop system in fuzzy logic controller has grown specially in recent years as the robot manipulator control [10]; [40-42]. Lhee et al. [40]have presented a fuzzy logic controller based on sliding mode controller to more formalize and boundary layer thickness. Emami *et al.* [43]have proposed a fuzzy logic approximate inside the boundary layer. H.K.Lee *et al.* [44] have presented self tuning SMFC to reduce the fuzzy rules, increase the stability and to adjust control parameters control automatically. However the application of FSMC and SMFC are growing but the main SMFC drawback compared to FSMC is calculation the value of sliding surface λ pri-defined very carefully. Moreover, the advantages of SMFC compared to FLC reduce the number of fuzzy rule base and increase the robustness and stability. At last FSMC compare to the SMFC is more suitable for implementation action.

In various dynamic parameters systems that need to be training on-line tuneable gain control methodology is used. On-line tuneable control methodology can be classified into two main groups, namely, traditional adaptive method and fuzzy adaptive method. Fuzzy adaptive method is used in systems which want to training parameters by expert knowledge. Traditional adaptive method is used in systems which some dynamic parameters are known. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, on-line tuneable method is applied to artificial sliding mode controller. F Y Hsu et al. [45]have presented adaptive fuzzy sliding mode control which can update fuzzy rules to compensate nonlinear parameters and guarantee the stability robot manipulator controller. Y.C. Hsueh et al. [35] have presented self tuning sliding mode controller which can resolve the chattering problem without to using saturation function. For nonlinear dynamic systems (e.g., robot manipulators) with various parameters, on-line control technique can train the dynamic parameter to have satisfactory performance. Calculate sliding surface slope is common challenge in classical sliding mode controller and fuzzy sliding mode controller. Research on adaptive (on-line tuneable) fuzzy control is significantly growing, for instance, different adaptive fuzzy controllers have been reported in [32, 46-48]. The adaptive sliding mode controller is used to estimate the unknown dynamic parameters and external disturbances. For instance, the applications of adaptive fuzzy sliding mode controller to control the robot manipulators have been reported in [11, 16, 37]. Yoo and Ham [49]have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. In $n-DOF$ robot manipulator with k membership function for each input variable, the number of fuzzy rules for each joint is equal to $3k^{2n}$ that causes to high computation load and also this controller has chattering. This method can only tune the consequence part of the fuzzy rules. Medhafer et al. [50] have proposed an indirect adaptive fuzzy sliding mode controller to control robot manipulator. This MIMO algorithm, applies to estimate the nonlinear dynamic parameters. If each input variable have K_2 membership functions, the number of fuzzy rules for each joint is $(m+1)K_2^m+K_2$. Compared with the previous algorithm the number of fuzzy rules have reduced by introducing the sliding surface as inputs of fuzzy systems. Y. Guo and P. Y. Woo [51]have proposed a SISO fuzzy system compensate and reduce the chattering. First suppose each input variable with K_2 membership function the number of fuzzy rules for each joint is K_2 which decreases the fuzzy rules and the chattering is also removed. C. M. Lin and C. F. Hsu [52] can tune both systems by fuzzy rules. In this method the number of fuzzy rules equal to K_2 with low computational load but it has chattering. Shahnazi et al., have proposed a SISO PI direct adaptive fuzzy sliding mode controller based on Lin and Hsu algorithm to reduce or eliminate chattering with K_2 fuzzy rules numbers. The bounds of PI controller and the parameters are online adjusted by low adaption computation [36]. Table 1 is illustrated a comparison between sliding mode controller [1, 5-11, 13], fuzzy logic controller (FLC)[23-32], applied sliding mode in fuzzy logic controller (SMFC)[10, 40-42], applied fuzzy logic method in sliding mode controller (FSMC)[45-46, 51] and adaptive fuzzy sliding mode controller [5-11].

This paper is organized as follows:
In section 2, design proposed backstepping on-line tunable gain in fuzzy sliding mode controller is presented. Detail of dynamic equation of robot arm is presented in section 3. In section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. DESIGN PROPOSED BACKSTEPPING ON-LINE TUNE FUZZY SLIDING MODE CONTROLLER

Sliding mode controller (SMC) is a influential nonlinear, stable and robust controller which it was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices[1, 5-11]. A time-varying sliding surface $s(x,t)$ is given by the following equation:

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

where λ is the constant and it is positive. The derivation of S, namely, $\dot{S}$ can be calculated as the following formulation [5-11]:

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \tag{2}$$

The control law for a multi degrees of freedom robot manipulator is written as:

$$U = U_{eq} + U_r \tag{3}$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and it can be calculate as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{4}$$

Where $M(q)$ is an inertia matrix which it is symmetric and positive, $V(q,\dot{q}) = B + C$ is the vector of nonlinearity term and $G(q)$ is the vector of gravity force and U_r with minimum chattering based on [5-11] is computed as;

$$U_r = K \cdot (\mathrm{mu} + \mathrm{b}) \left({S}/{\emptyset}\right) \tag{5}$$

Where $\emptyset_u = mu + b = saturation_{fuction}$ is a dead zone (saturation) function and, u and b are unlimited coefficient, by replace the formulation (5) in (3) the control output can be written as;

$$U = U_{eq} + K.(\mathrm{mu} + \mathrm{b})\left({S}/{\emptyset}\right) = \begin{cases} U_{eq} + K.\mathrm{sgn}(S) & ,|S| \geq \emptyset \\ U_{eq} + K.{S}/{\emptyset} & ,|S| < \emptyset \end{cases} \tag{6}$$

Where the function of $sgn(S)$ defined as;

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{7}$$

The fuzzy system can be defined as below

$$f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(S) \tag{8}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \ldots\ldots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \ldots\ldots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{9}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \ldots\ldots, \theta^M)$ is adjustable parameter in (8) and $\mu_{(xi)}$ is membership function.

error base fuzzy controller can be defined as

$$U_{fuzzy} = \psi(S) \tag{10}$$

The fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min \left[Sup_{x \in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - U_{equ} \right|\right] \tag{11}$$

Where $\theta^*$is the minimum error, $sup_{x \in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \right|$ is the minimum approximation error.

TABLE 1: Comparison of six important algorithms

Type of method	Advantages	Disadvantages	What to do?
1.SMC	• Good control performance for nonlinear systems • In MIMO systems • In discrete time circuit	• Equivalent dynamic formulation • Chattering • It has limitation under condition of : uncertain system and external disturbance	Applied artificial intelligent method in SMC (e.g., FSMC or SMFC)
2.FLC	• Used in unclear and uncertain systems • Flexible • Easy to understand • Shortened in design	• Quality of design • Should be to defined fuzzy coefficient very carefully • Cannot guarantee the stability • reliability	Applied adaptive method in FLC, tuning parameters and applied to classical linear or nonlinear controller
3.SMFC	• Reduce the rule base • Reduce the chattering • Increase the stability and robustness	• Equivalent part • Defined sliding surface slope coefficient very carefully • Difficult to implement • Limitation in noisy and uncertain system	Applied adaptive method, self learning and self organizing method in SMFC
4.FSMC	• More robust • Reduce the chattering • Estimate the equivalent • Easy to implement	• Model base estimate the equivalent part • Limitation in noisy and uncertain system	Design fuzzy error base like equivalent controller and applied adaptive method
5.Adaptive FSMC	• More robust • eliminate the chattering • Estimate the equivalent	• Model base estimate the equivalent part	

suppose K_j is defined as follows

$$K_j = \frac{\sum_{l=1}^{M} \theta_j^l [\mu_A(S_j)]}{\sum_{l=1}^{M} [\mu_A(S_j)]} = \theta_j^T \zeta_j(S_j) \quad (12)$$

Where $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad \textbf{(13)}$$

where the γ_{sj} is the positive constant.

According to the nonlinear dynamic equivalent formulation of robot manipulator the nonlinear equivalent part is estimated by (8)

$$\left[M^{-1}(B + C + G) + \dot{S}\right]M = \sum_{l=1}^{M} \theta^T \zeta(x) - \lambda S - K \quad \textbf{(14)}$$

Based on (3) the formulation of proposed fuzzy sliding mode controller can be written as;

$$U = U_{eq_{fuzzy}} + U_r \quad \textbf{(15)}$$

Where $U_{eq_{fuzzy}} = \left[M^{-1}(B + C + G) + \dot{S}\right]M + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

Figure 1 is shown the proposed fuzzy sliding mode controller.

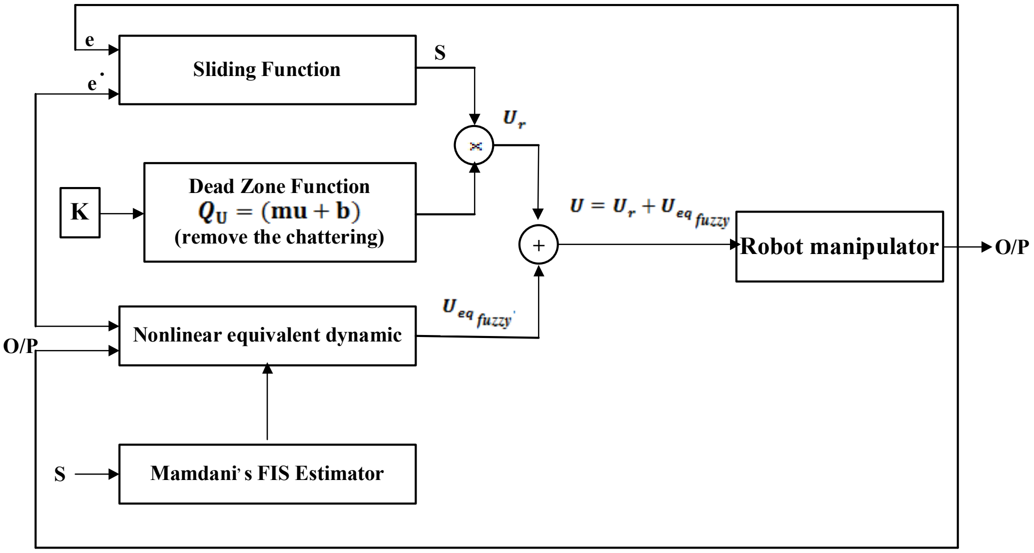


FIGURE 1: Proposed fuzzy sliding mode algorithm: applied to robot manipulator

As mentioned above pure sliding mode controller has nonlinear dynamic equivalent limitations in presence of uncertainty and external disturbances in order to solve these challenges this work applied Mamdani's fuzzy inference engine estimator in sliding mode controller. However proposed FSMC has satisfactory performance but calculate the sliding surface slope by try and error or experience knowledge is very difficult, particularly when system has structure or unstructured uncertainties; backstepping self tuning sliding function fuzzy sliding mode controller is recommended. The backstepping method is based on mathematical formulation which this method is introduced new variables into it in form depending on the dynamic equation of robot manipulator. This method is used as feedback linearization in order to solve nonlinearities in the system. To use of nonlinear fuzzy filter this method in this research makes it possible to create dynamic nonlinear equivalent backstepping estimator into the online tunable fuzzy sliding control process to eliminate or reduce the challenge of uncertainty in this part. The backstepping controller is calculated by;

$$U_{B.S} = U_{eq_{B.S}} + M.I \tag{15}$$

Where $U_{B.S}$ is backstepping output function, $U_{eq_{B.S}}$ is backstepping nonlinear equivalent function which can be written as (16) and I is backstepping control law which calculated by (17)

$$U_{eq_{B.S}} = [(B + C + G)] \tag{16}$$

$$I = \left[\ddot{\theta} + K_1(K_1 - 1) \cdot e + (K_1 + K_2) \cdot \dot{e}\right] \tag{17}$$

Based on (10) and (16) the fuzzy backstepping filter is considered as

$$(B + C + G) = \sum_{l=1}^{M} \theta^T \zeta(x) - \lambda S - K \tag{18}$$

Based on (15) the formulation of fuzzy backstepping filter can be written as;

$$U = U_{eqB.S_{fuzzy}} + MI \tag{19}$$

Where $U_{eq_{B.Sfuzzy}} = [(B + C + G)] + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

The adaption low is defined as

$$\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j) \tag{20}$$

where the γ_{sj} is the positive constant and $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \tag{21}$$

The dynamic equation of robot manipulator can be written based on the sliding surface as;

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \tag{22}$$

It is supposed that

$$S^T(\dot{M} - 2V)S = 0 \tag{23}$$

The derivation of Lyapunov function ($\dot{V}$) is written as

$$\begin{aligned}
\dot{V} &= \frac{1}{2} S^T \dot{M} S - S^T V S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j \\
&= S^T(-\lambda S + \Delta f - K) + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - K_j)] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - \theta_j^T \zeta_j(S_j))] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - (\theta_j^*)^T \zeta_j(S_j) + \phi_j^T \zeta_j(S_j))] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j \\
&= \sum_{j=1}^{m} [S_j(\Delta f_j - ((\theta_j^*)^T \zeta_j(S_j))] - S^T \lambda S)] + \sum (\frac{1}{\gamma_{sj}} \phi_j^T [\gamma_{sj} \zeta_j(S_j) S_j + \dot{\phi}_j])
\end{aligned}$$

Where $\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j)$ is adaption law and $\dot{\phi}_j = -\dot{\theta}_j = -\gamma_{sj} S_j \zeta_j(S_j)$, consequently $\dot{V}$ can be considered by

$$\dot{V} = \sum_{j=1}^{m} [S_j \Delta f_j - ((\theta_j^*)^T \zeta_j(S_j))] - S^T \lambda S \tag{24}$$

The minimum error can be defined by

$$e_{mj} = \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right) \tag{25}$$

$\dot{V}$ is intended as follows

$$\begin{aligned}\dot{V} &= \sum_{j=1}^{m} [S_j\, e_{mj}] - S^T \lambda S \\ &\le \textstyle\sum_{j=1}^{m} |S_j\,||e_{mj}| - S^T \lambda S \\ &= \sum_{j=1}^{m} |S_j\,||e_{mj}| - \lambda_j {S_j}^2 \\ &= \sum_{j=1}^{m} |S_j\,|\left(|e_{mj}| - \lambda_j S_j\right)\end{aligned} \tag{26}$$

For continuous function $U_{eqB.S_{fuzzy}}$ and suppose $\varepsilon > 0$ it is defined the fuzzy backstepping controller in form of (19) such that

$$Sup_{x\in U}\left|U_{eqB\ S_{fuzzy}} + MI\right| < \epsilon \tag{27}$$

As a result TBsFSMC is very stable which it is one of the most important challenges to design a controller with suitable response. Figure 2 is shown the block diagram of proposed TBsFSMC.

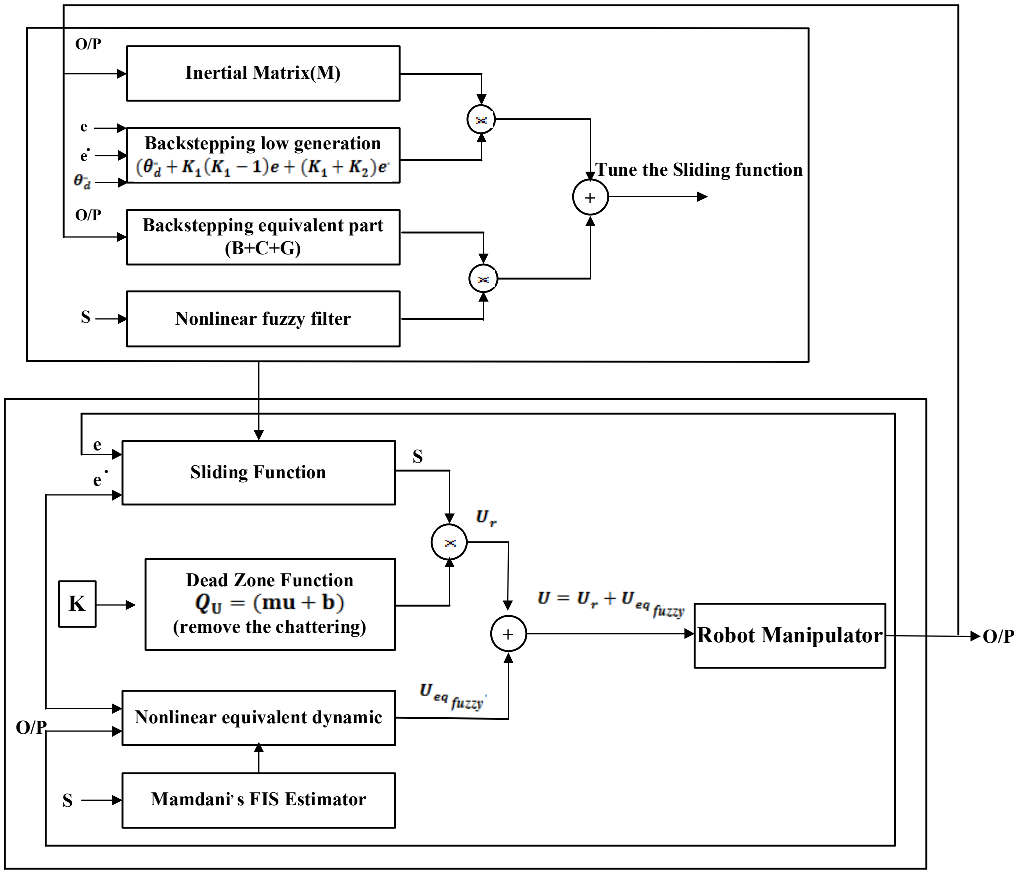


FIGURE 2: Proposed backstepping fuzzy like on line tuning FSMC algorithm: applied to robot manipulator

3. APPLICATION: DYNAMIC OF ROBOT MANIPULATOR

It is well known that the equation of an *n-DOF* robot manipulator governed by the following equation [1-3]:

$$M(q)\ddot{q} + N(q,\dot{q}) = \tau \tag{28}$$

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:

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\ \dot{q}] + C(q)[\dot{q}]^2 + G(q) \tag{29}$$

Where the matrix of coriolios torque is $B(q)$, $C(q)$ 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 [2-3, 5-11]:

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \tag{30}$$

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

Position control of PUMA-560 robot manipulator is analyzed in this paper; as a result the last three joints are blocked. The dynamic equation of PUMA-560 robot manipulator is given as

$$M(\theta)\begin{bmatrix}\ddot{\theta}1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix} + \mathrm{B}(\theta)\begin{bmatrix}\dot{\theta}_1\dot{\theta}_2\\ \dot{\theta}_1\dot{\theta}_3\\ \dot{\theta}_2\dot{\theta}_3\end{bmatrix} + C(\theta)\begin{bmatrix}\dot{\theta}_1^2\\ \dot{\theta}_2^2\\ \dot{\theta}_3^2\end{bmatrix} + G(\theta) = \begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} \tag{31}$$

Where

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0\\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0\\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0\\ 0 & 0 & 0 & M_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & M_{55} & 0\\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix} \tag{32}$$

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 & 0\\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{33}$$

$$C(q) = \begin{bmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0\\ C_{21} & 0 & C_{23} & 0 & 0 & 0\\ C_{31} & C_{32} & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ C_{51} & C_{52} & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{34}$$

$$G(q) = \begin{bmatrix} 0\\ g_2\\ g_3\\ 0\\ g_5\\ 0 \end{bmatrix} \tag{35}$$

Suppose $\ddot{q}$ is written as follows

$$\ddot{q} = M^{-1}(q).\{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{36}$$

and K is introduced as

$$K = \{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{37}$$

$\ddot{q}$ can be written as

$$\ddot{q} = M^{-1}(q).K \tag{38}$$

4. RESULT: VALIDITY CHECKING BETWEEN TBSFSMC, SMC AND FSMC

To validation of this work it is used 6-DOF's PUMA robot manipulator and implements proposed TBsFSMC, SMC and FSMC in this robot manipulator.

Tracking performances Figure 3 is shown tracking performance in TBsFSMC, SMC and FSMC without disturbance for proposed trajectory.

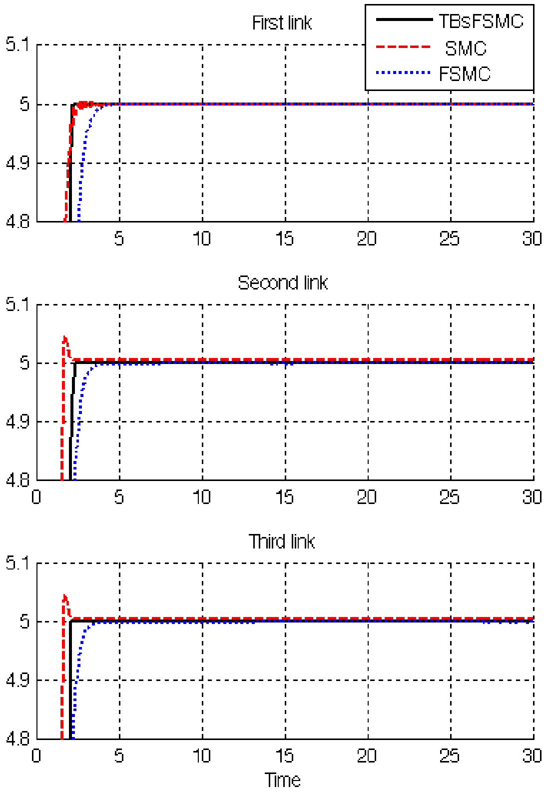


FIGURE 3: TBsFSMC, SMC and FSMC: without disturbance

By comparing this response, Figure 3, conversely the TBsFSMS and FSMC's overshoot are lower than SMC's, SMC's response is faster than TBsFSMC. The Settling time in TBsFSMC is fairly lower than SMC and FSMC.

Disturbance rejection: Figure 4 is indicated the power disturbance removal in TBsFSMC, SMC and FSMC. Besides a band limited white noise with predefined of 40% the power of input signal is applied to above controllers; it found slight oscillations in SMC and FSMC trajectory responses.

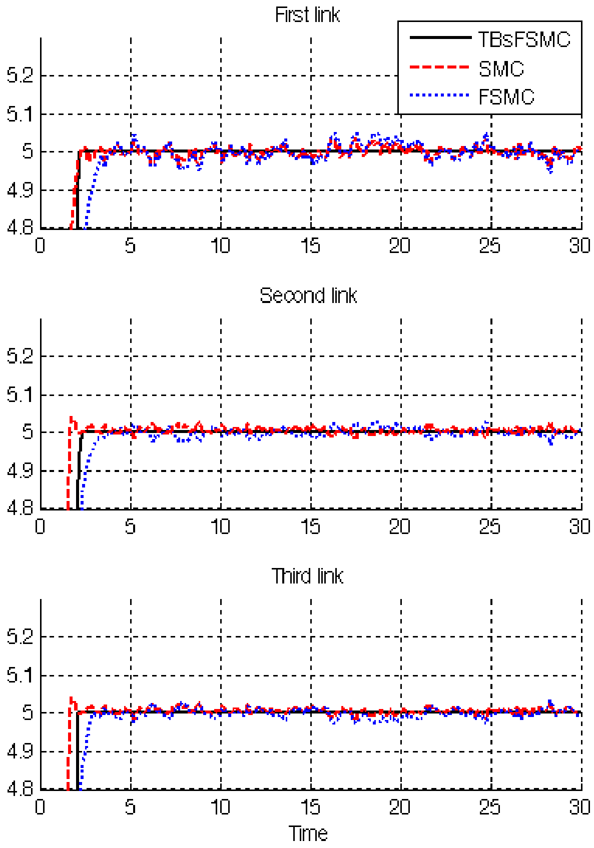


FIGURE 4: TBsFSMC, SMC and FSMC: with disturbance.

Among above graph, relating to step trajectory following with external disturbance, SMC and FSMC have slightly fluctuations. By comparing overshoot and rise time; SMC's overshoot **(4.4%)** is higher than FSMC and TBsFSMC, SMC’s rise time **(0.6 sec)** is considerably lower than FSMC and TBsFSMC. As mentioned in previous section, chattering is one of the most important challenges in sliding mode controller which one of the major objectives in this research is reduce or remove the chattering in system’s output. Figure 4 also has shown the power of boundary layer (saturation) method to reduce the chattering in above controllers. Overall in this research with regard to the step response, TBsFSMC has the steady chattering compared to the SMC and FSMC.

Errors in The Model

Although SMC and FSMC have the same error rate (refer to Table.2), they have high oscillation tracking which causes instability and chattering phenomenon at the presence of disturbances. As it is obvious in Table.2 proposed TBsFSMC has error reduction in noisy environment compared to the other controllers and displays smoother trend in above profiles.

TABLE2: RMS Error Rate of Presented controllers

RMS Error Rate	**SMC**	**FSMC**	**TBsFSMC**
Without Noise	**1e-3**	**1.2e-3**	**1e-5**
With Noise	**0.012**	**0.013**	**1e-5**

5. CONCLUSION

Refer to the research, a position backstepping on-line tuning fuzzy sliding mode control (TBsFSMC) design and application to 6 DOF's robot manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Regarding to the positive points in backstepping algorithm, sliding mode methodology, estimate the equivalent nonlinear part by applied fuzzy logic methodology and on-line tunable method, the output has improved. Each method by adding to the previous algorithms has covered negative points. In this work in order to solve uncertainty challenge in pure SMC, fuzzy logic estimator method is applied to sliding mode controller. In this paper Mamdani's fuzzy inference system has considered with one input (sliding function) fuzzy logic controller instead of mathematical nonlinear dynamic equivalent part. The system performance in fuzzy sliding mode controller is sensitive to the sliding function especially in presence of external disturbance. This problem is solved by adjusting sliding function of the fuzzy sliding mode controller continuously in real-time by on-line fuzzy like backstepping algorithm. In this way, the overall system performance has improved with respect to the fuzzy sliding mode controller and sliding mode controller. As mentioned in result, this controller solved chattering phenomenon as well as mathematical nonlinear equivalent part in presence of uncertainty and external disturbance by applied backstepping like fuzzy supervisory method in fuzzy sliding mode controller and on-line tuning the sliding function.

REFERENCES

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

[2] B. Armstrong*, et al.*, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," 2002, pp. 510-518.

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

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

[5] F. Piltan*, et al.*, "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," *Australian Journal of Basic and Applied Sciences, 5(6)*, pp. 509-522, 2011.

[6] Piltan, F., et al., 2011. Design sliding mode controller for robot manipulator with artificial tunable gain. Canaidian Journal of pure and applied science, 5 (2): 1573-1579.

[7] Piltan, F., et al., 2011. Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain, International Journal of Robotic and Automation, 2 (3): 205-220.

[8] Piltan, F., et al., 2011. Design Mathematical Tunable Gain PID-Like Sliding Mode Fuzzy Controller with Minimum Rule Base, International Journal of Robotic and Automation, 2 (3): 146-156.

[9] Piltan, F., et al., 2011. Design of FPGA based sliding mode controller for robot manipulator, International Journal of Robotic and Automation, 2 (3): 183-204.

[10] Piltan, F., et al., 2011. A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator, World Applied Science Journal, 12 (12): 2330-2336.

[11] Piltan, F., et al., 2011. Design Adaptive Fuzzy Robust Controllers for Robot Manipulator, World Applied Science Journal, 12 (12): 2317-2329.

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

[13] J. J. Slotine and S. Sastry, "Tracking control of non-linear systems using sliding surfaces, with application to robot manipulators†," *International Journal of Control,* vol. 38, pp. 465-492, 1983.

[14] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* vol. 40, pp. 421-434, 1984.

[15] R. Palm, "Sliding mode fuzzy control," 2002, pp. 519-526.

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

[17] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics,* vol. 10, pp. 239-263, 2000.

[18] 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,* vol. 41, pp. 1063-1068, 2002.

[19] H. Elmali and N. Olgac, "Implementation of sliding mode control with perturbation estimation (SMCPE)," *Control Systems Technology, IEEE Transactions on,* vol. 4, pp. 79-85, 2002.

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

[21] 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,* vol. 18, pp. 798-810, 2010.

[22] B. Wu*, et al.*, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," 2006, pp. 5-945.

[23] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems,* vol. 90, pp. 111-127, 1997.

[24] L. Reznik, *Fuzzy controllers*: Butterworth-Heinemann, 1997.

[25] J. Zhou and P. Coiffet, "Fuzzy control of robots," 2002, pp. 1357-1364.

[26] S. Banerjee and P. Y. Woo, "Fuzzy logic control of robot manipulator," 2002, pp. 87-88.

[27] K. Kumbla*, et al.*, "Soft computing for autonomous robotic systems," *Computers and Electrical Engineering,* vol. 26, pp. 5-32, 2000.

[28] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Transactions on systems, man and cybernetics,* vol. 20, pp. 404-418, 1990.

[29] R. J. Wai*, et al.*, "Implementation of artificial intelligent control in single-link flexible robot arm," 2003, pp. 1270-1275.

[30] R. J. Wai and M. C. Lee, "Intelligent optimal control of single-link flexible robot arm," *Industrial Electronics, IEEE Transactions on,* vol. 51, pp. 201-220, 2004.

[31] M. B. Menhaj and M. Rouhani, "A novel neuro-based model reference adaptive control for a two link robot arm," 2002, pp. 47-52.

[32] S. Mohan and S. Bhanot, "Comparative study of some adaptive fuzzy algorithms for manipulator control," *International Journal of Computational Intelligence,* vol. 3, pp. 303–311, 2006.

[33] F. Barrero*, et al.*, "Speed control of induction motors using a novel fuzzy sliding-mode structure," *Fuzzy Systems, IEEE Transactions on,* vol. 10, pp. 375-383, 2002.

[34] Y. C. Hsu and H. A. Malki, "Fuzzy variable structure control for MIMO systems," 2002, pp. 280-285.

[35] Y. C. Hsueh*, et al.*, "Self-tuning sliding mode controller design for a class of nonlinear control systems," 2009, pp. 2337-2342.

[36] R. Shahnazi*, et al.*, "Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control," *Energy Conversion, IEEE Transactions on,* vol. 23, pp. 138-147, 2008.

[37] C. C. Chiang and C. H. Wu, "Observer-Based Adaptive Fuzzy Sliding Mode Control of Uncertain Multiple-Input Multiple-Output Nonlinear Systems," 2007, pp. 1-6.

[38] H. Temeltas, "A fuzzy adaptation technique for sliding mode controllers," 2002, pp. 110-115.

[39] C. L. Hwang and S. F. Chao, "A fuzzy-model-based variable structure control for robot arms: theory and experiments," 2005, pp. 5252-5258.

[40] C. G. Lhee*, et al.*, "Sliding mode-like fuzzy logic control with self-tuning the dead zone parameters," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 343-348, 2002.

[41] Lhee. C. G., J. S. Park, H. S. Ahn, and D. H. Kim, "Sliding-Like Fuzzy Logic Control with Self-tuning the Dead Zone Parameters," *IEEE International fuzzy systems conference proceeding*, 1999,pp.544-549.

[42] X. Zhang*, et al.*, "Adaptive sliding mode-like fuzzy logic control for high order nonlinear systems," pp. 788-792.

[43] M. R. Emami*, et al.*, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems,* vol. 6, 1998.

[44] H.K.Lee, K.Fms, "A Study on the Design of Self-Tuning Sliding Mode Fuzzy Controller. Domestic conference," *IEEE Conference*, *1994,* vol. 4, pp. 212-218.

[45] F. Y. Hsu and L. C. Fu, "Nonlinear control of robot manipulators using adaptive fuzzy sliding mode control," 2002, pp. 156-161.

[46] R. G. Berstecher*, et al.*, "An adaptive fuzzy sliding-mode controller," *Industrial Electronics, IEEE Transactions on,* vol. 48, pp. 18-31, 2002.

[47] V. Kim, "Independent joint adaptive fuzzy control of robot manipulator," 2002, pp. 645-652.

[48] Y. Wang and T. Chai, "Robust adaptive fuzzy observer design in robot arms," 2005, pp. 857-862.

[49] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* vol. 8, pp. 186-199, 2002.

[50] H. Medhaffar*, et al.*, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* vol. 1, pp. 23-29, 2006.

[51] 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,* vol. 33, pp. 149-159, 2003.

[52] C. M. Lin and C. F. Hsu, "Adaptive fuzzy sliding-mode control for induction servomotor systems," *Energy Conversion, IEEE Transactions on,* vol. 19, pp. 362-368, 2004.

An Expert System Algorithm for Computer System Diagnostics

Aaron Don M. Africa *aaronafrica@yahoo.com*
Faculty of Engineering
Electronics and Communications Engineering Department
De La Salle University Manila
2401 Taft Avenue Manila Philippines

Abstract

In troubleshooting Computer Systems the two most common causes of delay are Trial and Error and having Incomplete Information. The problems in Computer Systems will be fixed faster if the Possible Cause of the Problem is already known. A solution to this is to use an Expert System. This system can reproduce the ability of an expert to diagnose by giving an accurate recommendation on the possible cause of the problem for effective troubleshooting.

To know the Possible Cause of a problem there must be a complete set of information. These data will be the one to be inputted in the Expert System to give an accurate recommendation. A problem is that in reality a complete set of data will not always be obtained. There will be instances when the information gathered will be incomplete.

This research solved the two most causes of delay which are Trial and Error and having Incomplete Information. This is done by developing an Expert System Algorithm that creates the rules of an Expert System. The rules created from the algorithm are nominal in terms that only the necessary information needs to be inputted. In instances that the data gathered are incomplete the correct Possible Cause can still be suggested. A theorem is also presented in this research about and the Information Dependency of Data which can be used with Incomplete Information Systems and unknown data. Formal Proof of the theorem is provided and its correctness was verified with actual data.

Keywords: Computer Systems, Expert Systems, Real time systems, Database Engineering, Information Management.

---

1. INTRODUCTION

An Expert System is an Artificial Intelligence Based System that performs task that otherwise is performed by a human expert [1]. This type of system usually has a knowledge base containing accumulated experience and a set of rules for applying the knowledge base to each particular solution.

The most common cause of delay in solving a problem is trial and error [2]. The problem can be solved earlier if the person diagnosing it already knows the cause of the problem rather than resorting to trial and error. There are instances that because of this trial and error, the problem gets worse rather than being solved. Some problems can be solved quickly; there are situations when it only takes a few minutes to solve a problem but because the person diagnosing it does not know the cause of the problem, troubleshooting takes days or months causing much inconvenience.

An example in Computer Systems, a technician encountered an error of “MOM Alerts on Server: SVREBPPDBS01” and this is the first time he has encountered this problem. He will attempt several troubleshooting techniques in finding the Possible Cause (PC). It is often rigorous and time consuming requiring the mobilization of resources. He may guess that it is a Computer Virus Problem and reinstall new Anti Viral programs or a Hardware problem and replace the Database server causing huge amounts of money. But the real Possible Cause of that symptom is “Microsoft Office Manager (MOM) Alerts on Server” which means that the server is already full. The solution to this PC is to shrink the Database, which only takes less than 5 minutes. Knowing this problem before hand will save time and resources. This is the primary use of Expert Systems - it reduces trial and error in problems on a specific domain.

Data on Information Systems is important in any type of enterprise. The data is often used to interpret information and make decisions [3]. An example is in an Expert System enough information must be inputted in order to give the correct conclusion. In reality, you will not be able to obtain all the data that you need. Data will be vague and incomplete, thus, it will be difficult to produce any conclusion [4]. Knowing the right and necessary attributes to obtain is important especially if you have limited time and resources [5]. Coming up with the correct conclusion even with minimal information is a great advantage [6].

2. OVERVIEW

2.1. Example Symptoms and Possible Causes

Consider this Example Information System:

Case	Possible Cause	Symptoms
1	**PC1:** FTP Software Trouble	**S1:** Error Connection Appears, **S2:** Cannot Access Network Drives, **S3:** Destination unreachable error appears, **S4:** Page Cannot be accessed Error Appears
2	**PC2:** Server connection failure	**S2:** Cannot Access Network Drives, **S3:** Destination unreachable error appears, **S4:** Page Cannot be accessed Error Appears
3	**PC2:** Server connection failure	**S2:** Cannot Access Network Drives, **S4:** Page Cannot be accessed Error Appears
4	**PC2:** Server connection failure	**S2:** Cannot Access Network Drives, **S4:** Page Cannot be accessed Error Appears
5	**PC3:** Email Queues Increasing	**S2:** Cannot Access Network Drives, **S3:** Destination unreachable error appears, **S4:** Page Cannot be accessed Error Appears

TABLE 1: Symptoms and Possible Cause (PC)

Table 1 list some network and internetwork problems or trouble which may be encountered by Computer Systems. It presents us some possible causes, symptoms and solutions which we could undertake so to resolve particular errors.

ID	Possible Cause
PC1	FTP Software Trouble
PC2	Server connection failure
PC3	Email Queues Increasing

TABLE 2: List of Possible causes

The Table 2 presents list of possible causes of network failure. It states that FTP Software Trouble may arise if there’s a conflict on the software that we are using. FTP Software Trouble might hinder the user from transferring information or data from one computer to the other. Another possible causes is the Server Connection Failure, this may arise if there’s a problem on the physical connection of the server. Accessing the server from the client workstation may be unreachable. Lastly, the Email Queues Increasing may arise if there’s a problem on the Internet or intranet connection which leads to the increase on the amount of email messages on the queue.

ID	Symptom
S1	Error Connection Appears
S2	Cannot Access Network Drives
S3	Destination unreachable error appears
S4	Page Cannot be accessed Error Appears

TABLE 3: List Symptoms

Table 3 presents the List of Symptoms of network connection failure presented on the other table of Possible Causes. This table summarizes the symptoms that we should know so that we could be able to anticipate network errors. Symptom S1 tells about the Error Connection Appears, this might prompt us on some error messages on our screen. Symptom S2 states that the network drives cannot be access. Symptom S3 tells about Destination unreachable error appears on the screen. This symptom simply states that the particular workstation cannot be reached by a particular connecting workstation. The last one which is symptom S4 presents about page cannot be accessed error appears. This error pertains to the Internet or intranet Connection Error wherein it has no capability to access the particular page due to no connection.

E	D \ Q	S1	S2	S3	S4
1	**PC1**	1	1	1	1
2	**PC2**	0	1	1	1
3	**PC2**	0	1	0	1
4	**PC2**	0	1	0	1
5	**PC3**	0	1	1	1

TABLE 4: Information System of Table 1

Table 4 shows the Data in Table 1 converted to an Information System.

2.2. List of Mathematical Symbols

The following are the list of Mathematical Symbols used in this research and their explanations:

Symbol	Name	Explanation
S	Information System	A 4–tuple $S = \langle D, Q, V, \rho \rangle$
D	Set of Possible Causes	It is a set of Possible Causes. For example PC1 – FTP Software Trouble, PC2 – Server connection failure and PC3 – Email Queues Increasing as shown in Table 2. *D* = {PC1, PC2, PC3}.
Q	Set of Symptoms	It is a set of Symptoms. For example S1 – Error Connection Appears, S2 – Cannot Access Network Drives, S3 – Destination unreachable error appears and S4 – Page Cannot be accessed Error Appears as shown in Table 3. *Q* = {S1, S2, S3, S4}.
E	Set of Cases	*E* = {1,2,3,….a} for some natural number *a*. For example in Table 4 *E* = {1,2,3,4,5}.
ρ	Relation from $D \times Q$ to V	Let ρ be the relation from $D \times Q$ to V which assigns at least one value for $(i, j) \in (E \times Q)$. For example in Table

		4: $\rho(PC2, S3)$ = 0 or 1 $\rho(PC1, S1)$ = 1
V	Codomain of ρ	For example in Table 4: $V = \{1,0\}$.
$(p)(q) = f$	Notation	Value or values associated with Selected Possible cause p and Selected Symptom q. Let f be called the value of a Symptom $f \subseteq V$. For example in Table 4: $(PC1)(S1)=\{1\}$ $(PC2)(S3)=\{1,0\}$ $(PC3)(S1)=\{0\}$
ab	Index	Indicates the location of a variable in a Mathematical object. For example M_{ab}, ab is its index. [7]
p	Selected Possible cause	$p \in D$ and the Possible cause arbitrarily Selected. For example D = {PC1, PC2, PC3}. If PC1 is selected $p = PC1$.
p'	Other Possible causes	Possible causes other than the selected Possible cause p, that is p' is an element of D such that $p' \neq p$.
Q	Selected Symptom	$q \in Q$ and the symptom arbitrarily selected. For example Q = {S1, S2, S3, S4}. If S1 is selected it will be the q.
q^f	Equality in associated format.	This is another way to write equality. q^f means q has a value of f. For example $q = 1$. It can be written as q^1 [8].
=>	Dependence Notation	$(q = f) => p$ means that $(q=f)$ is a sufficient condition for p. For example $(Q_1 = 1) => x$. If the value of the Selected Symptom Q_1 is 1 then it can be concluded that x is satisfied. .

TABLE 5: List of Mathematical Symbols

2.3. Incomplete Information System and Information Dependency of Data.

In Computer Systems, Data is important. Data is often used to interpret and make decisions [9]. In Expert Systems for example, Data gathered is used as a Knowledge Base. The rules of Expert Systems are from the Knowledge Base Data. The more Data in the system, the better it can interpret information [10]. However, in reality you will be able to gather the Data that you need. There will be situations that due to limited time and resources, you will have to prioritize your Information Gathering [11].

An Incomplete Information System (IIS) is a 4–tuple $S = \langle D, Q, V, \rho \rangle$ (1), In this tuple D is a set of Possible causes, Q is a set of Symptoms and ρ is the relation from $D \times Q$ to V (2) which assigns at least one value for $(i, j) \in (E \times Q)$. F is the value of a symptom which may contain an unknown value represented by the symbol "*" [12].

To further explain the concepts of Incomplete Information System consider the following example in System Network Performance:

E	**D \ Q**	**S1**	**S2**	**S3**	**S4**
1	**PC1**	1	*	1	1
2	**PC1**	1	0	*	*
3	**PC2**	0	1	*	1
4	**PC2**	0	1	0	1
5	**PC3**	0	*	1	*

TABLE 6: An Incomplete Information System

In Table 6:

S1: Error Connection Appears
S2: Cannot Access Network Drives
S3: Destination unreachable error appears
S4: Page cannot be accessed Error Appears

PC1: FTP Software Trouble
PC2: Server connection failure
PC3: Email Queues Increasing
1: Symptom exist
0: Symptom does not exist
* : Cannot obtain the data

S1, S2, S3 and *S4* are the Symptoms and *D* is the Possible cause. This is for a total of 6 cases.

$$Q = \{S1, S2, S3, S4\} \quad (3)$$
$$D = \{PC1, PC2, PC3\} (4)$$
$$E = \{1,2,3,4,5,6\} \quad (5)$$
$$V = \{1,0,*\} \quad (6)$$

Table 6 gives an example of an Incomplete Information System. Equation 3 shows the Symptoms used which are S1, S2, S3 and S4. Equation 4 shows the Possible causes which can either be PC1, PC2 or PC3.
Equation 5 shows the cases which are from 1 to 6. Equation 2 showed that the relation ρ is the product set of *D* and *Q* mapped into *V* which assigns at least one value for $(i, j) \in (C \times Q)$ and can have a value of either 1,0 or * as shown in equation 6.

In Case 1 and 2 of Table 6 for example S1 = PC1 is needed for D to be PC1. Let S1 = PC1 be defined as essential information needed to satisfy the D to be PC1. It can be said that value of D being PC1 is dependent on S1 = PC1. The Possible cause “PC1” has many data conditions and some of them are unknown. For example in Case 2 where S3 and S4 are unknown and S1 = 1, the other data is unimportant as long as the value of S1 = 1 it can be said that D = PC1. The concept of dependent is important in Incomplete Information Systems. For Example in Table 6 where D = PC1 is dependent on S1 = 1, the only information needed to be obtain is if S1 = 1 and not the other information in S3 and S4 which are incomplete.

2.4. Nominality of a Rule

Initially to make the rules each case will be checked. One rule is for one case. For example in Table 4 Case 1 will produce the following Rule:

Rule 1: (S1 = 1) & (S2 = 1) & (S3 = 1) & (S4 = 1) => (D = PC1)

The Symptoms will have a value of 1 if it exists in the case and a value of 0 if it does not. For Rule 1 S1, S2, S3 and S4 must exist for D to be PC1. All 5 cases will have the following Rules:

Rule 1: (S1 = 1) & (S2 = 1) & (S3 = 1) & (S4 = 1) => (D = PC1)
Rule 2: (S1 = 0) & (S2 = 1) & (S3 = 1) & (S4 = 1) => (D = PC2)
Rule 3: (S1 = 0) & (S2 = 1) & (S3 = 0) & (S4 = 1) => (D = PC2)
Rule 4: (S1 = 0) & (S2 = 1) & (S3 = 0) & (S4 = 1) => (D = PC2)
Rule 5: (S1 = 0) & (S2 = 1) & (S3 = 1) & (S4 = 1) => (D = PC3)

In a typical process of troubleshooting, the technician will check all the symptoms needed to satisfy the possible cause in order to conclude that it is the actual Cause. Verifying the existence of the symptom takes time and resources. For example in Rule 1 the technician must verify if Error Connection Appears, Network Drives cannot be accessed, Destination unreachable error appears and Page Cannot be accessed Error Appears. Verifying just one of the symptoms takes time like Destination Unreachable Error Appears. To verify this symptom the technician will have to ping the computers in the network. If there are many computers in the network doing this verification takes time.

The rules of the Information System can still be reduced. For example in Table 4 D = PC1 is dependent on the value of S1 being 1. Therefore to satisfy D = PC1 verification needs to be done only in S1, not needing S2, S3 and S4. So even if S2, S3 or S4 are incomplete it can still be concluded as D = PC1. The rules that are reduced are called in nominal form.

2.5. Theorem

Theorem 1: Consider an Information System $S = \langle D, Q, V, \rho \rangle$. Let *p* be a selected Possible Cause and let *q* be a selected Symptom. Assume $(y)(q) \neq *$ for all $y \in D$. If $(p)(q)$ is a singleton and is not a subset or equal to the value of $(p')(q)$ then the selected Possible Cause is dependent on the value of the selected Symptom *f.*

Observe that in the above theorem an Information System maybe incomplete. However the condition $(y)(q) \neq *$ for all $y \in D$ requires that column *q* of the Information System be complete.

Proof:

Consider the sample Information System:

C	**D \ Q**	Q_1	Q_2	Q_3	Q_4	$\ldots Q_b$
1	D_1	C_1	C_2	C_3	C_4	$\ldots C_{ab}$
2	D_1	C_2	C_3	C_4	C_4	$\ldots C_{ab}$
3	D_2	C_2	C_2	C_2	C_1	$\ldots C_{ab}$
4	D_3	C_4	C_3	C_2	C_1	$\ldots C_{ab}$
$\vdots$ a	$\vdots$ D_{ab}	$\vdots$ C_{ab}	$\vdots$ C_{ab}	$\vdots$ C_{ab}	$\vdots$ C_{ab}	$\vdots$ C_{ab}

TABLE 7: Information System of Data

In this example Information System

$$Q = \{Q_1, Q_2, Q_3, Q_4, \ldots Q_b\}$$
$$C = \{1, 2, 3, 4, \ldots a\}$$
$$V = \{C_1, C_2, C_3, C_4, \ldots C_{ab}\}$$

Attributes Q_1 to Q_b are Symptoms D is the Possible cause.

$Q = Q_4$
$p = D_1$
$p' =: D_2, D_{3,} .. D_{ab}$
$f = \{C_4\}$
$q^f = Q_4^{C_4}$

In the Information System $(p)(q)$ is a singleton and is not a subset or equal to the value of $(p')(q)$.

The Information System will then be translated from tabular form to logical form.

$$\begin{aligned}
&[(Q_1 = C_1) \wedge (Q_2 = C_2) \wedge (Q_3 = C_3) \wedge (Q_4 = C_4) \ldots \wedge (Q_b = C_{ab}) \wedge (D = D_1)] \vee \\
&[(Q_1 = C_2) \wedge (Q_2 = C_3) \wedge (Q_3 = C_4) \wedge (Q_4 = C_4) \ldots \wedge (Q_b = C_{ab}) \wedge (D = D_1)] \vee \\
&[(Q_1 = C_2) \wedge (Q_2 = C_2) \wedge (Q_3 = C_2) \wedge (Q_4 = C_1) \ldots \wedge (Q_b = C_{ab}) \wedge (D = D_2)] \vee \\
&[(Q_1 = C_4) \wedge (Q_2 = C_3) \wedge (Q_3 = C_2) \wedge (Q_4 = C_1) \ldots \wedge (Q_b = C_{ab}) \wedge (D = D_3)] \vee \ldots \\
&[(Q_1 = C_{ab}) \wedge (Q_2 = C_{ab}) \wedge (Q_3 = C_{ab}) \wedge (Q_4 = C_{ab}) \ldots \wedge (Q_b = C_{ab}) \wedge (D = D_{ab})]
\end{aligned}$$

Rewriting the equation in a simplified format:

$$\begin{aligned}
&\left(Q_1^{C_1} Q_2^{C_2} Q_3^{C_3} Q_4^{C_4} \ldots Q_b^{C_{ab}} D^{D_1}\right) \vee \left(Q_1^{C_2} Q_2^{C_3} Q_3^{C_4} Q_4^{C_4} \ldots Q_b^{C_{ab}} D^{D_1}\right) \vee \left(Q_1^{C_2} Q_2^{C_2} Q_3^{C_2} Q_4^{C_1} \ldots Q_b^{C_{ab}} D^{D_2}\right) \vee \\
&\left(Q_1^{C_4} Q_2^{C_3} Q_3^{C_2} Q_4^{C_1} \ldots Q_b^{C_{ab}} D^{D_3}\right) \vee \left(Q_1^{C_{ab}} Q_2^{C_{ab}} Q_3^{C_{ab}} Q_4^{C_{ab}} \ldots Q_b^{C_{ab}} D^{D_{ab}}\right)
\end{aligned}$$

Writing the Decision Matrix for the Selected Possible Cause *p* which is D_1

E	3	4	…a
1	$Q_1^{C_1} Q_3^{C_3} Q_4^{C_4} \ldots Q_b^{C_{ab}}$	$Q_1^{C_1} Q_2^{C_2} Q_3^{C_3} Q_4^{C_4} \ldots Q_b^{C_{ab}}$	$Q_4^{C_4} \ldots Q_b^{C_{ab}}$
2	$Q_2^{C_3} Q_3^{C_4} Q_4^{C_4} \ldots Q_b^{C_{ab}}$	$Q_1^{C_2} Q_3^{C_4} Q_4^{C_4} \ldots Q_b^{C_{ab}}$	$Q_4^{C_4} \ldots Q_b^{C_{ab}}$

TABLE 8: Decision Matrix

Since the q^f will always be present in all the intersections of the decision matrix in *p* then we can conclude that $(q = f) \Rightarrow p$.

3. DATA TAGGING ALGORITHM

3.1. Flow Chart of the Algorithm

The information can be organized in a Problem Symptom relationship pattern where different Problems can be associated with different Symptoms. Also the same type of symptoms can be present in different problems. The same Possible Cause (PC) can also have a different set of symptoms. These data relationships can be organized in an Information System. Given a dataset the attributes can be dicretize and find a subset from the original value therefore simplifying it. The resulting information will be used as the rules of the Expert System. The rules created in the algorithm are nominal in where only the minimal information is needed. It is very useful in actual applications where it will not be possible to obtain all the information that you need. Knowing the right information to obtain and confirm is helpful especially with limited time and resources. The Data Tagging algorithm for Expert System rule creation is presented in Figure 1.

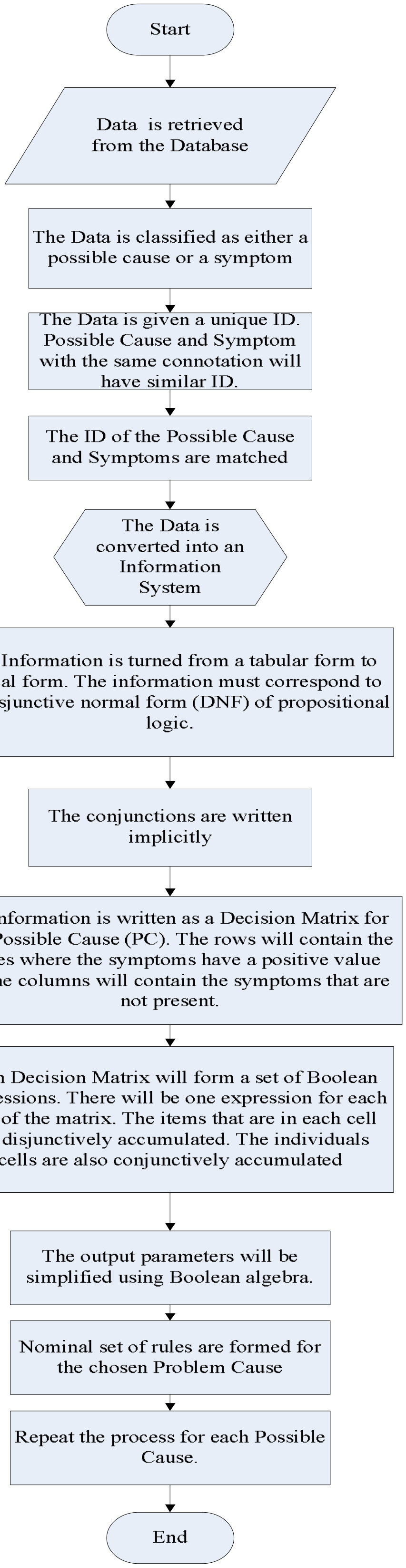


FIGURE 1: Data Tagging algorithm

3.2. Illustrative Example of the Algorithm

The following shows an illustrative example showing all the steps necessary to implement the algorithm:

1. Data is retrieved from the Database

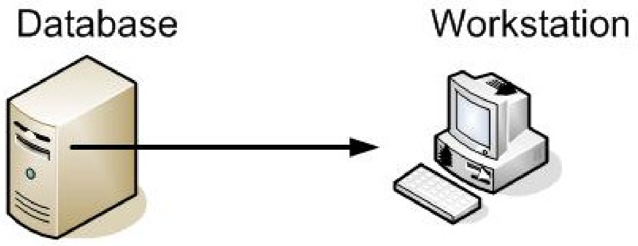


FIGURE 2: Retrieval of Data

2. Data is classified as either a Possible Cause or Symptom

   Possible Causes: FTP Software Trouble, Server connection failure, Email Queues Increasing, FTP Program Problem and Server cannot connect.
   Symptoms: Error Connection Appears, Cannot Access Network Drives, Destination unreachable error appears Page cannot be accessed Error Appears, Network Drive Error and Destination Cannot be reached.

3. Data is given a unique ID. Possible Cause and Symptoms with the same connotation will have the same ID.

   There are Possible Cause and Symptoms with the same connotation meaning they have the same meaning. For example in the Symptom: Error Connection Appears is the same as Network Drive Error. They will have the same ID.

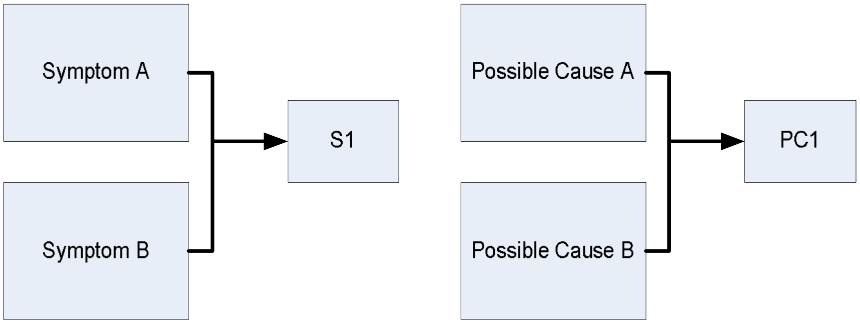


FIGURE 3: Assigning of unique ID

4. The ID of the Possible Cause and Symptoms are matched

   The Problems and Symptoms are matched with their corresponding ID. For example S1 will be the ID for the Symptom “Error Connection Appears”. The structure of the technical data will be in a Possible Cause, Symptom and solution relationship.

   In Table 1 a new technique to input the technical data if an ICT organization is presented. The information that will be inputted are for the cases that have already been resolved.

5. The Data is converted into an Information System. The technical data can then be converted into an Information System as shown in Table 4.
6. The Information System is turned from a tabular form to logical form. The Information must correspond to the Disjunctive Normal Form (DNF) of propositional logic.

   The next step is to turn the Information System from tabular form to logical form by expressing the set of objects as the following disjunction, which corresponds to the disjunctive normal form (DNF) of propositional logic.

$$\begin{aligned}
&[(S1=1)\wedge(S2=1)\wedge(S3=1)\wedge(S4=1)\wedge(D=PC1)]\vee\\
&[(S1=0)\wedge(S2=1)\wedge(S3=1)\wedge(S4=1)\wedge(D=PC2)]\vee\\
&[(S1=0)\wedge(S2=1)\wedge(S3=0)\wedge(S4=1)\wedge(D=PC2)]\vee\\
&[(S1=0)\wedge(S2=1)\wedge(S3=0)\wedge(S4=1)\wedge(D=PC2)]\vee\\
&[(S1=0)\wedge(S2=1)\wedge(S3=1)\wedge(S4=1)\wedge(D=PC3)]
\end{aligned}$$

7. The Conjuctions are simplified.

$$\begin{aligned}
&\left(S_1{}^1S_2{}^1S_3{}^1S_4{}^1D^{PC1}\right)\vee\left(S_1{}^0S_2{}^1S_3{}^1S_4{}^1D^{PC2}\right)\vee\left(S_1{}^0S_2{}^1S_3{}^0S_4{}^1D^{PC2}\right)\vee\\
&\left(S_1{}^0S_2{}^1S_3{}^0S_4{}^1D^{PC2}\right)\vee\left(S_1{}^0S_2{}^1S_3{}^1S_4{}^1D^{PC3}\right)
\end{aligned}$$

8. The Information is written as a Decision Matrix for each Possible Cause (PC). The rows will contain the values where the symptoms have a positive value and the columns will contain the symptoms that are not present.

   The Target Possible Cause is chosen. For this example the Possible Cause PC1 is chosen. The upper and lower approximation of the System Attribute is now chosen.

E	**2**	**3**	**4**	**5**
1	$S_1{}^1$	$S_1{}^1, S_3{}^1$	$S_1{}^1, S_3{}^1$	$S_1{}^1$

TABLE 9: Decision Matrix for D = PC1

9. Each Decision Matrix will form a set of Boolean Expressions. There will be one expression for each row of the matrix. The items that are in each cell are disjunctively accumulated. The individual cells are also conjunctively accumulated.

   Boolean Expressions from the boundaries: $\left(S_1{}^1\right)\wedge\left(S_1{}^1\vee S_3{}^1\right)\wedge\left(S_1{}^1\vee S_3{}^1\right)\wedge\left(S_1{}^1\right)$

10. The output parameters will be simplified using Boolean algebra.
    Using Boolean algebra the expression is simplified to: $S_1{}^1$
11. Nominal Set of Rules is formed for the chosen Possible Cause.
    Rule 1. (S1 = 1) => (PC = 1)
12. Repeat the process for each Possible Cause.

The Algorithm produced a nominal set of rules. It is capable of handling Different Possible causes with unique set of symptoms.

Rule 1. (S1 = 1) => (PC = 1)
Rule 2. (S3 = 0) => (PC = 2)
Rule 3. (S1 = 0) & (S3 = 1) => (PC = 2) OR (PC = 3)

4. DATA AND RESULTS

4.1. Presentation of Actual Data

The Theorem and the algorithm will be tested and validated using actual Data. They are the problems encountered by a Computer System division of a telecommunication company. The following are the Data with the Possible Cause and its Symptoms:

Case	Possible Cause	Symptoms
1	**PC1:** Runtime Errors	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message
2	**PC2:** Divide Errors	**S3:** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S5:** Error message regarding autoexec.bat or config.sys
3	**PC3:** msgsrv32 Error	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S4:** Memory Overflow message appears
4	**PC4:** Not valid Win32 Application	**S4:** Memory Overflow message appears, **S6:** USB Virus message, **S7:** To many programs running on startup
5	**PC5:** Network Connection Failure	**S8:** The URL Cannot be accessed through the MDB Portal, **S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed
6	**PC6:** Network Dataport Problem	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S15:** CPU hangs
7	**PC7:** LAN Card malfunction	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S15:** CPU hangs
8	**PC7:** LAN Card malfunction	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
9	**PC8**: Server Alerts are Encountered in Office Manager	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S16:** Clicking anything can take minutes before computer response, **S29:** Registry error message keeps on appearing
10	**PC8**: Server Alerts are Encountered in Office Manager	**S10:** Mapped Drive Cannot be accessed, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S14:** SVRMDBADDC12 Cannot be accessed, **S29:** Registry error message keeps on appearing
11	**PC9:** Blue Alerts (Software) in Office Manager	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S17:** Computer cannot recognize Mc Afee Installed
12	**PC10:** Yellow Alerts (Hardware) in Office Manager	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs, **S29:** Registry error message keeps on appearing
13	**PC11:** Network not properly Mapped	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S17:** Computer cannot recognize Mc Afee Installed
14	**PC12:** Multiple Antivirus Programs are active	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S17:** Computer cannot recognize Mc Afee Installed
15	**PC13:** Memory Overflow Problem	**S3.** Computer Motherboard beeps, **S18:** Video Card Slot is loose, **S19:** DVI Slot is shorted
16	**PC14:** Video card Problem	**S18:** Video Card Slot is loose, **S20:** Distorted Screen, **S21:** Windows monitor driver error appears
17	**PC14:** Videocard Problem	**S18:** Video Card Slot is loose, **S21:** Windows monitor driver error appears
18	**PC14:** Videocard Problem	**S19:** DVI Slot is shorted, **S21:** Windows monitor driver error appears
19	**PC15:** DVI cable Defect	**S22:** Scraped marks on the DVI Cable, **S29:** Registry error message keeps on appearing
20	**PC16:** Monitor Component Defect	**S18:** Video Card Slot is loose, **S23:** Monitor will not power on

21	**PC16:** Monitor Component Defect	**S19:** DVI Slot is shorted, **S20:** Distorted Screen, **S23:** Monitor will not power on
22	**PC17:** MOM Alerts Critical Error	**S1:** Motherboard BIOS beeps, **S4:** Memory Overflow message appears, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
23	**PC17:** MOM Alerts Critical Error	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs
24	**PC17:** MOM Alerts Critical Error	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
25	**PC18:** MOM Alerts on Application	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S29:** Registry error message keeps on appearing
26	**PC18:** MOM Alerts on Application	**S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S29:** Registry error message keeps on appearing
27	**PC19:** MOM Alerts on Database	**S4:** Memory Overflow message appears, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs
28	**PC19:** MOM Alerts on Database	**S4:** Memory Overflow message appears, **S8:** The URL Cannot be accessed through the MDB Portal
29	**PC19:** MOM Alerts on Database	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S11:** MOM Alerts on Server: SVREBPPDBS01
30	**PC20:** MOM Alerts on Services and Performance	**S3.** Computer Motherboard beeps, **S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears
31	**PC20:** MOM Alerts on Services and Performance	**S4:** Memory Overflow message appears, **S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears
32	**PC21:** MOM Critical Alerts - Services Unavailable	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
33	**PC21:** MOM Critical Alerts - Services Unavailable	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs
34	**PC22:** Server Harddisk Full	**S4:** Memory Overflow message appears, **S15:** CPU hangs, **S21:** Windows monitor driver error appears
35	**PC23:** Cannot Log-On to Network	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S22:** Scraped marks on the DVI Cable
36	**PC23:** Cannot Log-On to Network	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S22:** Scraped marks on the DVI Cable
37	**PC24:** Domain Server Unavailable	**S1:** Motherboard BIOS beeps, **S4:** Memory Overflow message appears, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
38	**PC24:** Domain Server Unavailable	**S8:** The URL Cannot be accessed through the MDB Portal, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32
39	**PC24:** Domain Server Unavailable	**S1:** Motherboard BIOS beeps, **S5:** Error message regarding autoexec.bat or config.sys, **S8:** The URL Cannot be accessed through the MDB Portal, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S22:** Scraped marks on the DVI Cable
40	**PC25:** Program Application :"Low Virtual Memory" Alert Encountered	**S4:** Memory Overflow message appears, **S15:** CPU hangs, **S21:** Windows monitor driver error appears
41	**PC26:** Network connection Failure	**S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S22:** Scraped marks on the DVI Cable
42	**PC26:** Network connection Failure	**S1:** Motherboard BIOS beeps, **S10:** Mapped Drive Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
43	**PC27:** Network connection Intermittent	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs, **S22:** Scraped marks on the DVI Cable
44	**PC28:** MS Office Cannot Be Accessed	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S24:** MS Office Program error in running
45	**PC29:** MS Office Communicator Cannot Be Accessed	**S9:** Network Connection Error Appears, **S21:** Windows monitor driver error appears, **S24:** MS Office Program error in running

46	**PC29:** MS Office Communicator Cannot Be Accessed	**S1:** Motherboard BIOS beeps, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S24:** MS Office Program error in running
47	**PC30:** MS Excel Error Encountered	**S4:** Memory Overflow message appears, **S5:** Error message regarding autoexec.bat or config.sys, **S24:** MS Office Program error in running
48	**PC31:** MS Office Clipart Gallery Does not Work	**S1.** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S7:** To many programs running on startup, **S24:** MS Office Program error in running
49	**PC31:** MS Office Clipart Gallery Does not Work	**S1.** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S5:** Error message regarding autoexec.bat or config.sys, **S16:** Clicking anything can take minutes before computer response, **S24:** MS Office Program error in running
50	**PC32:** MS Office Shortcuts not working properly	**S1.** Motherboard BIOS beeps, **S4:** Memory Overflow message appears, **S10:** Mapped Drive Cannot be accessed, **S20:** Distorted Screen, **S24:** MS Office Program error in running
51	**PC33:** Print Half Page Only	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S17:** Computer cannot recognize Mc Afee Installed, **S25:** Printer Error Light Blinks
52	**PC33:** Print Half Page Only	**S3.** Computer Motherboard beeps, **S5:** Error message regarding autoexec.bat or config.sys, **S16:** Clicking anything can take minutes before computer response, **S25:** Printer Error Light Blinks
53	**PC34:** Error Code 28	**S21:** Windows monitor driver error appears, **S22:** Scraped marks on the DVI Cable, **S23:** Monitor will not power on
54	**PC34:** Error Code 28	**S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S21:** Windows monitor driver error appears, **S22:** Scraped marks on the DVI Cable, **S23:** Monitor will not power on
55	**PC35:** Monitor Blackout	**S19:** DVI Slot is shorted, **S22:** Scraped marks on the DVI Cable, **S23:** Monitor will not power on
56	**PC35:** Monitor Blackout	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S18:** Video Card Slot is loose, **S23:** Monitor will not power on
57	**PC35:** Monitor Blackout	**S3.** Computer Motherboard beeps, **S19:** DVI Slot is shorted, **S20:** Distorted Screen
58	**PC36:** Monitor Blurred / Flickers	**S20:** Distorted Screen, **S21:** Windows monitor driver error appears, **S22:** Scraped marks on the DVI Cable
59	**PC36:** Monitor Blurred / Flickers	**S18:** Video Card Slot is loose, **S20:** Distorted Screen, **S21:** Windows monitor driver error appears
60	**PC37:** Printer Head Problem	**S4:** Memory Overflow message appears, **S25:** Printer Error Light Blinks
61	**PC38:** CPU Power Supply Problem	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S23:** Monitor will not power on, **S26:** CPU Turns off few minutes after opening
62	**PC38:** CPU Power Supply Problem	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S26:** CPU Turns off few minutes after opening
63	**PC39:** CPU Slowdown Encountered	**S1:** Motherboard BIOS beeps, **S3:** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S7:** To many programs running on startup, **S16:** Clicking anything can take minutes before computer response
64	**PC39:** CPU Slowdown Encountered	**S1:** Motherboard BIOS beeps, **S4:** Memory Overflow message appears, **S15:** CPU hangs
65	**PC39:** CPU Slowdown Encountered	**S3:** Computer Motherboard beeps, **S7:** To many programs running on startup, **S24:** MS Office Program error in running
66	**PC40:** Email Service Slowdown	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S15:** CPU hangs
67	**PC40:** Email Service Slowdown	**S2:** Computer Virus Message, **S3:** Computer Motherboard beeps, **S17:** Computer cannot recognize Mc Afee Installed
68	**PC40:** Email Service Slowdown	**S1:** Motherboard BIOS beeps, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S14:** MS SVRMDBADDC12 Cannot be accessed
69	**PC41:** Program Application Infected with Virus	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S17:** Computer cannot recognize Mc Afee Installed
70	**PC41:** Program Application Infected with Virus	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears **S15:** CPU hangs, **S17:** Computer cannot recognize Mc Afee Installed

71	**PC41:** Program Application Infected with Virus	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed
72	**PC42:** OS Performs Illegal Operations	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S6:** USB Virus message
73	**PC42:** OS Performs Illegal Operations	**S5:** Error message regarding autoexec.bat or config.sys, **S7:** To many programs running on startup, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32
74	**PC42:** OS Performs Illegal Operations	**S1:** Motherboard BIOS beeps, **S4:** Memory Overflow message appears, **S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears
75	**PC42:** OS Performs Illegal Operations	**S1:** Motherboard BIOS beeps, **S8:** The URL Cannot be accessed through the MDB Portal, **S10:** Mapped Drive Cannot be accessed, **S15:** CPU hangs
76	**PC43:** OS Performs Illegal Operations	**S2:** Computer Virus Message, **S5:** Error message regarding autoexec.bat or config.sys, **S6:** USB Virus message
77	**PC43:** OS Performs Illegal Operations	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed
78	**PC43:** OS Performs Illegal Operations	**S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs
79	**PC44:** LCA Cannot Be Accessed	**S1:** Motherboard BIOS beeps, **S14:** SVRMDBADDC12 Cannot be accessed
80	**PC44:** LCA Cannot Be Accessed	**S2:** Computer Virus Message, **S6:** USB Virus message, **S10:** Mapped Drive Cannot be accessed, **S15:** CPU hangs
81	**PC44:** LCA Cannot Be Accessed	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S6:** USB Virus message, **S8:** The URL Cannot be accessed through the MDB Portal, **S13:** SVR-MDBSPPS-01 Cannot be accessed
82	**PC44:** LCA Cannot Be Accessed	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
83	**PC45:** Kronos problem	**S2:** Computer Virus Message, **S6:** USB Virus message, **S7:** To many programs running on startup, **S27:** CPU Clock keeps on Changing
84	**PC45:** Kronos problem	**S1:** Motherboard BIOS beeps, **S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S27:** CPU Clock keeps on Changing
85	**PC45:** Kronos problem	**S1:** Motherboard BIOS beeps, **S15:** CPU hangs, **S27:** CPU Clock keeps on Changing
86	**PC45:** Kronos problem	**S5:** Error message regarding autoexec.bat or config.sys, **S7:** To many programs running on startup, **S16:** Clicking anything can take minutes before computer response, **S27:** CPU Clock keeps on Changing
87	**PC46:** Network IP Address Conflict	**S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed
88	**PC46:** Network IP Address Conflict	**S8:** The URL Cannot be accessed through the MDB Portal, **S10:** Mapped Drive Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
89	**PC46:** Network IP Address Conflict	**S7:** To many programs running on startup, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32
90	**PC46:** Network IP Address Conflict	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S14:** SVRMDBADDC12 Cannot be accessed
91	**PC47:** CPU COM/Serial Port Problem	**S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S15:** CPU hangs
92	**PC47:** CPU COM/Serial Port Problem	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S15:** CPU hangs, **S18:** Video Card Slot is loose
93	**PC47:** CPU COM/Serial Port Problem	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S15:** CPU hangs, **S26:** CPU Turns off few minutes after opening
94	**PC48:** OS Disk Error	**S3.** Computer Motherboard beeps, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32
95	**PC48:** OS Disk Error	**S4:** Memory Overflow message appears, **S5:** Error message regarding

		autoexec.bat or config.sys, **S26:** CPU Turns off few minutes after opening
96	**PC48:** OS Disk Error	**S2:** Computer Virus Message, **S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S9:** Network Connection Error Appears
97	**PC49:** Printer Fuser Assembly error	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S25:** Printer Error Light Blinks
98	**PC49:** Printer Fuser Assembly error	**S1:** Motherboard BIOS beeps, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S17:** Computer cannot recognize Mc Afee Installed, **S25:** Printer Error Light Blinks
99	**PC50:** Internet Email cannot received/sent	**S1:** Motherboard BIOS beeps, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S12:** MOM Alerts on Server: SVREBPPEBS32
100	**PC50:** Internet Email cannot received/sent	**S1:** Motherboard BIOS beeps, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed
101	**PC50:** Internet Email cannot received/sent	**S1:** Motherboard BIOS beeps, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
102	**PC50:** Internet Email cannot received/sent	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S7:** To many programs running on startup, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed
103	**PC51:** Defective USB Port	**S2:** Computer Virus Message, **S6:** USB Virus message, **S16:** Clicking anything can take minutes before computer response
104	**PC51:** Defective USB Port	**S10:** Mapped Drive Cannot be accessed, **S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response
105	**PC52:** File Cannot Be Copied	**S2:** Computer Virus Message, **S6:** USB Virus message, **S7:** To many programs running on startup
106	**PC52:** File Cannot Be Copied	**S1:** Motherboard BIOS beeps, **S3.** Computer Motherboard beeps, **S15:** CPU hangs
107	**PC52:** File Cannot Be Copied	**S1:** Motherboard BIOS beeps, **S4:** Memory Overflow message appears **S5:** Error message regarding autoexec.bat or config.sys, **S15:** CPU hangs
108	**PC53:** Files Cannot be Download	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S6:** USB Virus message, **S29:** Registry error message keeps on appearing
109	**PC53:** Files Cannot be Download	**S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S26:** CPU Turns off few minutes after opening, **S29:** Registry error message keeps on appearing
110	**PC53:** Files Cannot be Download	**S9:** Network Connection Error Appears, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S29:** Registry error message keeps on appearing
111	**PC54:** Public Folder Cannot Be Accessed	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S28:** Network Sharing Error
112	**PC54:** Public Folder Cannot Be Accessed	**S3.** Computer Motherboard beeps, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S28:** Network Sharing Error
113	**PC54:** Public Folder Cannot Be Accessed	**S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S28:** Network Sharing Error
114	**PC55:** Cannot Log-in to Domain	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears
115	**PC55:** Cannot Log-in to Domain	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed
116	**PC55:** Cannot Log-in to Domain	**S7:** To many programs running on startup, **S9:** Network Connection Error Appears, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
117	**PC56:** Garbled Images in the monitor	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S6:** USB Virus message
118	**PC56:** Garbled Images in the monitor	**S18:** Video Card Slot is loose, **S19:** DVI Slot is shorted, **S21:** Windows monitor driver error appears, **S22:** Scraped marks on the DVI Cable
119	**PC56:** Garbled Images in the monitor	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears,

		S18: Video Card Slot is loose, **S19:** DVI Slot is shorted
120	**PC56:** Garbled Images in the monitor	**S18:** Video Card Slot is loose, **S19:** DVI Slot is shorted, **S20:** Distorted Screen
121	**PC57:** Cannot Access Application Error	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S6:** USB Virus message, **S7:** To many programs running on startup
122	**PC57:** Cannot Access Application Error	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S15:** CPU hangs, **S16:** Clicking anything can take minutes before computer response, **S18:** Video Card Slot is loose
123	**PC57:** Cannot Access Application Error	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S5:** Error message regarding autoexec.bat or config.sys, **S16:** Clicking anything can take minutes before computer response, **S17:** Computer cannot recognize Mc Afee Installed, **S26:** CPU Turns off few minutes after opening
124	**PC58:** File Folder Cannot be Established	**S2:** Computer Virus Message, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S29:** Registry error message keeps on appearing
125	**PC58:** File Folder Cannot be Established	**S6:** USB Virus message, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32, **S29:** Registry error message keeps on appearing
126	**PC58:** File Folder Cannot be Established	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs, **S24:** MS Office Program error in running, **S29:** Registry error message keeps on appearing
127	**PC58:** File Folder Cannot be Established	**S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S5:** Error message regarding autoexec.bat or config.sys, **S29:** Registry error message keeps on appearing
128	**PC59:** (ISNet) Defective	**S8:** The URL Cannot be accessed through the MDB Portal, **S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed
129	**PC59:** (ISNet) Defective	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S11:** MOM Alerts on Server: SVREBPPDBS01, **S12:** MOM Alerts on Server: SVREBPPEBS32
130	**PC59:** (ISNet) Defective	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed
131	**PC60:** Harddisk Bad Sector found	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S6:** USB Virus message, **S26:** CPU Turns off few minutes after opening
132	**PC60:** Harddisk Bad Sector found	**S2:** Computer Virus Message, **S3.** Computer Motherboard beeps, **S6:** USB Virus message
133	**PC61:** File Print Problem	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S15:** CPU hangs, **S25:** Printer Error Light Blinks
134	**PC61:** File Print Problem	**S1:** Motherboard BIOS beeps, **S15:** CPU hangs, **S25:** Printer Error Light Blinks
135	**PC61:** File Print Problem	**S2:** Computer Virus Message, **S3.** Computer Motherboard beeps, **S6:** USB Virus message, **S25:** Printer Error Light Blinks
136	**PC62:** OS Registry Corrupted	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S6:** USB Virus message, **S29:** Registry error message keeps on appearing
137	**PC62:** OS Registry Corrupted	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S14:** SVRMDBADDC12 Cannot be accessed, **S15:** CPU hangs, **S29:** Registry error message keeps on appearing
138	**PC62:** OS Registry Corrupted	**S5:** Error message regarding autoexec.bat or config.sys, **S26:** CPU Turns off few minutes after opening, **S29:** Registry error message keeps on appearing
139	**PC62:** OS Registry Corrupted	**S2:** Computer Virus Message, **S3.** Computer Motherboard beeps, **S4:** Memory Overflow message appears, **S6:** USB Virus message, **S29:** Registry error message keeps on appearing
140	**PC63:** Printer Sensor Problem	**S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S25:** Printer Error Light Blinks
141	**PC63:** Printer Sensor Problem	**S4:** Memory Overflow message appears, **S6:** USB Virus message, **S25:** Printer Error Light Blinks
142	**PC63:** Printer Sensor Problem	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S4:** Memory Overflow message appears, **S25:** Printer Error Light Blinks
143	**PC64:** CPU Fan Not Functioning	**S15:** CPU hangs, **S23:** Monitor will not power on, **S26:** CPU Turns off few minutes after opening
144	**PC64:** CPU Fan Not Functioning	**S4:** Memory Overflow message appears, **S15:** CPU hangs,

		S26: CPU Turns off few minutes after opening
145	**PC65:** MOBO Driver Installed but not working	**S15:** CPU hangs, **S21:** Windows monitor driver error appears, **S29:** Registry error message keeps on appearing, **S30:** CPU has no sound
146	**PC65:** MOBO Driver Installed but not working	**S13:** SVR-MDBSPPS-01 Cannot be accessed, **S17:** Computer cannot recognize Mc Afee Installed, **S24:** MS Office Program error in running, **S29:** Registry error message keeps on appearing, **S30:** CPU has no sound
147	**PC65:** MOBO Driver Installed but not working	**S2:** Computer Virus Message, **S21:** Windows monitor driver error appears, **S26:** CPU Turns off few minutes after opening, **S30:** CPU has no sound
148	**PC65:** MOBO Driver Installed but not working	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S6:** USB Virus message, **S15:** CPU hangs, **S26:** CPU Turns off few minutes after opening, **S30:** CPU has no sound
149	**PC66:** Black and White Output	**S19:** DVI Slot is shorted, **S20:** Distorted Screen, **S22:** Scraped marks on the DVI Cable
150	**PC66:** Black and White Output	**S18:** Video Card Slot is loose, **S19:** DVI Slot is shorted, **S20:** Distorted Screen, **S22:** Scraped marks on the DVI Cable
151	**PC66:** Black and White Output	**S19:** DVI Slot is shorted, **S20:** Distorted Screen, **S21:** Windows monitor driver error appears, **S22:** Scraped marks on the DVI Cable
152	**PC67:** Sound Card Problem	**S15:** CPU hangs, **S26:** CPU Turns off few minutes after opening, **S30:** CPU has no sound
153	**PC67:** Sound Card Problem	**S15:** CPU hangs, **S30:** CPU has no sound
154	**PC68:** Code 10 Error	**S2:** Computer Virus Message, **S6:** USB Virus message, **S30:** CPU has no sound
155	**PC68:** Code 10 Error	**S7:** To many programs running on startup, **S9:** Network Connection Error Appears, **S30:** CPU has no sound
156	**PC68:** Code 10 Error	**S1:** Motherboard BIOS beeps, **S2:** Computer Virus Message, **S9:** Network Connection Error Appears, **S30:** CPU has no sound
157	**PC68:** Code 10 Error	**S9:** Network Connection Error Appears, **S10:** Mapped Drive Cannot be accessed, **S30:** CPU has no sound
158	**PC69:** Error 0xc0000142	**S2:** Computer Virus Message, **S29:** Registry error message keeps on appearing, **S30:** CPU has no sound
159	**PC69:** Error 0xc0000142	**S6:** USB Virus message, **S29:** Registry error message keeps on appearing, **S30:** CPU has no sound
160	**PC69:** Error 0xc0000142	**S4:** Memory Overflow message appears, **S26:** CPU Turns off few minutes after opening, **S29:** Registry error message keeps on appearing, **S30:** CPU has no sound
161	**PC70:** CPU speaker is not functioning	**S4:** Memory Overflow message appears, **S15:** CPU hangs, **S30:** CPU has no sound
162	**PC70:** CPU speaker is not functioning	**S16:** Clicking anything can take minutes before computer response **S30:** CPU has no sound
163	**PC71:** Device Manager Error Code 19	**S1:** Motherboard BIOS beeps, **S5:** Error message regarding autoexec.bat or config.sys, **S29:** Registry error message keeps on appearing
164	**PC71:** Device Manager Error Code 19	**S1:** Motherboard BIOS beeps, **S5:** Error message regarding autoexec.bat or config.sys, **S6:** USB Virus message, **S29:** Registry error message keeps on appearing
165	**PC71:** Device Manager Error Code 19	**S2:** Computer Virus Message, **S3:** Computer Motherboard beeps, **S6:** USB Virus message, **S29:** Registry error message keeps on appearing

TABLE 10: Symptoms in Computer System with their Possible Cause (PC)

E	D\Q	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24	S25	S26	S27	S28	S29	S30
1	PC1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	PC2	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	PC3	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	PC4	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	PC5	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	PC6	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

7	PC7	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	PC7	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	PC8	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
10	PC8	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
11	PC9	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	PC10	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
13	PC11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	PC12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
15	PC13	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
16	PC14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0
17	PC14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
18	PC14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
19	PC15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
20	PC16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
21	PC16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0
22	PC17	1	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	PC17	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	PC17	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	PC18	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
26	PC18	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
27	PC19	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	PC19	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	PC19	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	PC20	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	PC20	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	PC21	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	PC21	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	PC22	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
35	PC23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
36	PC2	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

	3																														
37	PC24	1	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	PC24	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
39	PC24	1	0	0	0	1	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
40	PC25	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
41	PC26	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
42	PC26	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	PC27	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
44	PC28	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
45	PC29	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
46	PC29	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
47	PC30	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
48	PC31	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
49	PC31	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
50	PC32	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
51	PC33	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
52	PC33	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
53	PC34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
54	PC34	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
55	PC35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0
56	PC35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0
57	PC35	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
58	PC36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
59	PC36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0
60	PC37	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
61	PC38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0
62	PC38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
63	PC3	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	9																														
64	PC39	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
65	PC39	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
66	PC40	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67	PC40	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
68	PC40	1	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
69	PC41	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
70	PC41	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
71	PC41	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
72	PC42	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
73	PC42	0	0	0	0	1	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
74	PC42	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
75	PC42	1	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
76	PC43	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
77	PC43	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
78	PC43	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
79	PC44	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
80	PC44	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
81	PC44	1	1	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
82	PC44	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
83	PC45	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
84	PC45	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
85	PC45	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
86	PC45	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
87	PC46	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
88	PC46	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
89	PC46	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	PC4	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	6																														
91	PC47	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
92	PC47	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
93	PC47	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
94	PC48	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
95	PC48	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
96	PC48	0	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
97	PC49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
98	PC49	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
99	PC50	1	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	PC50	1	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101	PC50	1	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
102	PC50	1	1	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
103	PC51	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
104	PC51	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
105	PC52	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
106	PC52	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
107	PC52	1	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
108	PC53	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
109	PC53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0
110	PC53	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
111	PC54	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
112	PC54	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
113	PC54	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
114	PC55	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
115	PC55	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
116	PC55	0	0	0	0	0	0	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	PC5	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

7	6																														
118	PC56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0
119	PC56	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
120	PC56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
121	PC57	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
122	PC57	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
123	PC57	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0
124	PC58	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
125	PC58	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
126	PC58	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
127	PC58	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
128	PC59	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
129	PC59	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
130	PC59	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
131	PC60	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
132	PC60	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
133	PC61	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
134	PC61	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
135	PC61	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
136	PC62	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
137	PC62	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
138	PC62	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
139	PC62	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
140	PC63	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
141	PC63	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
142	PC63	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
143	PC64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
14	PC6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

4	4																														
145	PC65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1
146	PC65	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	1
147	PC65	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1
148	PC65	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
149	PC66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0
150	PC66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0
151	PC66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
152	PC67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
153	PC67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
154	PC68	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
155	PC68	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
156	PC68	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
157	PC68	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
158	PC69	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
159	PC69	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
160	PC69	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
161	PC70	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
162	PC70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
163	PC71	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
164	PC71	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
165	PC71	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

TABLE 11: Information System of the Data

ID	Symptom
S1	Conflict with TSR Running Program
S2	Computer Virus Message
S3	Computer Motherboard beeps
S4	Memory Overflow message appears
S5	Error message regarding autoexec.bat or config.sys

S6	USB Virus message
S7	To many programs running on startup
S8	The URL Cannot be accessed through the MDB
S9	Network Connection Error Appears
S10	Mapped Drive Cannot be accessed
S11	MOM Alerts on Server: SVREBPPDBS01
S12	MOM Alerts on Server: SVREBPPEBS32
S13	SVR-MDBSPPS-01 Cannot be accessed
S14	SVRMDBADDC12 Cannot be accessed
S15	CPU hangs
S16	Clicking anything can take minutes before computer
S17	Computer cannot recognize Mc Afee
S18	Video Card Slot is loose
S19	DVI Slot is shorted
S20	Distorted Screen
S21	Windows monitor driver error appears
S22	Scraped marks on the DVI Cable
S23	Monitor will not power on
S24	MS Office Program error in running
S25	Printer Error Light Blinks
S26	CPU Turns off few minutes after opening
S27	CPU Clock keeps on Changing
S28	Network Sharing Error
S29	Registry error message keeps on appearing
S30	CPU has no sound

TABLE 12: Table of Symptoms

Table 10, 11 and 12 showed the Symptoms in Computer System Diagnostics with their Possible Cause (PC), Information System of the Data and a Table of symptoms respectively.

4.2. Decision Rules by Applying the Algorithm

The Information system is inputted into the test platform Program. Hypertext Preprocessor (PHP), integrated with Rough Sets Data Explorer was used as a test platform [13]. This PHP Test Platform applies the Data Tagging Algorithm.

Applying the complete Algorithm described in Section 3, a nominal set of rules are produced these are:

Rule #	Rule
Rule 1	(S2 = 1) & (S3 = 0) & (S4 = 0) & (S5 = 0) & (S7 = 0) & (S8 = 0) & (S12 = 0) & (S15 = 0) & (S16 = 0) & (S29 = 0) & (S30 = 0) => (D = PC1)
Rule 2	(S3 = 1) & (S4 = 1) & (S5 = 1) & (S29 = 0) => (D = PC2)
Rule 3	(S1 = 1) & (S3 = 0) & (S4 = 1) & (S6 = 0) & (S8 = 0) & (S10 = 0) & (S14 = 0) & (S15 = 0) & (S25 = 0) & (S26 = 0) => (D = PC3)
Rule 4	(S4 = 1) & (S6 = 1) & (S7 = 1) => (D = PC4)
Rule 5	(S8 = 1) & (S9 = 0) & (S13 = 1) => (D = PC5)
Rule 6	(S6 = 0) & (S9 = 0) & (S11 = 1) & (S29 = 1) => (D = PC8)
Rule 7	(S11 = 1) & (S17 = 1) => (D = PC9)
Rule 8	(S3 = 1) & (S18 = 1) => (D = PC13)

Rule 9	(S18 = 0) & (S19 = 0) & (S20 = 1) & (S22 = 0) & (S24 = 0) => (D = PC14)
Rule 10	(S9 = 0) & (S15 = 0) & (S20 = 0) & (S21 = 1) & (S22 = 0) & (S26 = 0) => (D = PC14)
Rule 11	(S22 = 1) & (S29 = 1) => (D = PC15)
Rule 12	(S15 = 0) & (S22 = 0) & (S23 = 1) => (D = PC16)
Rule 13	(S1 = 0) & (S4 = 0) & (S10 = 0) & (S12 = 0) & (S14 = 1) & (S22 = 0) & (S29 = 0) => (D = PC17)
Rule 14	(S2 = 0) & (S10 = 1) & (S11 = 0) & (S29 = 1) => (D = PC18)
Rule 15	(S1 = 0) & (S2 = 0) & (S3 = 0) & (S4 = 1) & (S7 = 0) & (S9 = 0) & (S21 = 0) & (S24 = 0) & (S25 = 0) & (S26 = 0) & (S30 = 0) => (D = PC19)
Rule 16	(S11 = 1) & (S12 = 0) & (S13 = 0) & (S14 = 0) => (D = PC19)
Rule 17	(S1 = 0) & (S8 = 1) & (S9 = 1) & (S10 = 0) & (S14 = 0) => (D = PC20)
Rule 18	(S11 = 1) & (S12 = 1) & (S14 = 1) => (D = PC21)
Rule 19	(S5 = 0) & (S14 = 0) & (S19 = 0) & (S20 = 0) & (S22 = 1) & (S23 = 0) & (S29 = 0) => (D = PC23)
Rule 20	(S8 = 1) & (S11 = 1) => (D = PC24)
Rule 21	(S6 = 0) & (S8 = 0) & (S9 = 0) & (S10 = 1) & (S16 = 0) & (S24 = 0) & (S28 = 0) & (S29 = 0) => (D = PC26)
Rule 22	(S13 = 1) & (S15 = 1) & (S22 = 1) => (D = PC27)
Rule 23	(S1 = 0) & (S12 = 1) & (S24 = 1) => (D = PC28)
Rule 24	(S15 = 0) & (S18 = 0) & (S20 = 0) & (S21 = 1) & (S23 = 0) & (S26 = 0) => (D = PC29)
Rule 25	(S1 = 1) & (S12 = 1) & (S24 = 1) => (D = PC29)
Rule 26	(S4 = 1) & (S5 = 1) & (S24 = 1) => (D = PC30)
Rule 27	(S2 = 1) & (S4 = 1) => (D = PC31)
Rule 28	(S20 = 1) & (S24 = 1) => (D = PC32)
Rule 29	(S12 = 0) & (S18 = 1) & (S25 = 1) => (D = PC33)
Rule 30	(S5 = 1) & (S16 = 1) & (S25 = 1) => (D = PC33)
Rule 31	(S21 = 1) & (S23 = 1) => (D = PC34)
Rule 32	(S16 = 1) & (S18 = 1) & (S23 = 1) => (D = PC35)
Rule 33	(S21 = 0) & (S22 = 1) & (S23 = 1) => (D = PC35)
Rule 34	(S3 = 1) & (S20 = 1) => (D = PC35)
Rule 35	(S19 = 0) & (S20 = 1) & (S22 = 1) => (D = PC36)
Rule 36	(S2 = 0) & (S3 = 0) & (S4 = 1) & (S5 = 0) & (S6 = 0) & (S8 = 0) & (S10 = 0) & (S14 = 0) & (S15 = 0) & (S29 = 0) => (D = PC37)
Rule 37	(S4 = 0) & (S16 = 1) & (S26 = 1) & (S29 = 0) => (D = PC38)
Rule 38	(S2 = 0) & (S6 = 0) & (S7 = 1) & (S12 = 0) & (S27 = 0) & (S30 = 0) => (D = PC39)
Rule 39	(S1 = 1) & (S2 = 0) & (S3 = 0) & (S4 = 1) & (S5 = 0) & (S8 = 0) & (S10 = 0) & (S14 = 0) => (D = PC39)
Rule 40	(S3 = 1) & (S17 = 1) => (D = PC40)
Rule 41	(S11 = 1) & (S13 = 0) & (S14 = 1) & (S29 = 0) => (D = PC40)
Rule 42	(S1 = 0) & (S2 = 1) & (S3 = 0) & (S6 = 0) & (S17 = 0) & (S18 = 0) & (S25= 0) & (S26 = 0) & (S29 = 0) => (D = PC40)
Rule 43	(S3 = 0) & (S11 = 0) & (S16 = 0) & (S17 = 1) & (S24 = 0) & (S25 = 0) => (D = PC41)
Rule 44	(S1 = 1) & (S12 = 1) & (S13 = 1) => (D = PC41)
Rule 45	(S1 = 1) & (S2 = 0) & (S5 = 0) & (S8 = 1) => (D = PC42)
Rule 46	(S5 = 1) & (S7 = 1) & (S27 = 0) => (D = PC42)
Rule 47	(S1 = 1) & (S4 = 1) & (S6 = 1) => (D = PC42)
Rule 48	(S4 = 0) & (S5 = 1) & (S7 = 0) & (S22 = 0) & (S24 = 0) & (S25 = 0) & (S29 = 0) => (D = PC43)
Rule 49	(S12 = 1) & (S15 = 1) => (D = PC43)
Rule 50	(S8 = 1) & (S13 = 1) & (S14 = 1) => (D = PC44)
Rule 51	(S3 = 0) & (S4 = 0) & (S5 = 0) & (S6 = 1) & (S7 = 0) & (S16 = 0) & (S29 = 0) & (S30 = 0) => (D = PC44)
Rule 52	(S1 = 1) & (S4 = 0) & (S13 = 1) & (S14 = 1) => (D = PC44)
Rule 53	(S27 = 1) => (D = PC45)
Rule 54	(S3 = 0) & (S5 = 0) & (S6 = 0) & (S8 = 0) & (S9 = 0) & (S11 = 1) & (S12 = 1) & (S14 = 0) & (S16 = 0) & (S24 = 0) => (D = PC46)
Rule 55	(S8 = 1) & (S13 = 0) & (S14 = 1) => (D = PC46)
Rule 56	(S1 = 1) & (S2 = 1) & (S15 = 1) & (S16 = 0) & (S30 = 0) => (D = PC47)
Rule 57	(S3 = 1) & (S4 = 1) & (S15 = 1) => (D = PC47)
Rule 58	(S3 = 1) & (S5 = 0) & (S6 = 0) & (S7 = 0) & (S8 = 0) & (S15 = 0) & (S17 = 0) & (S19 = 0) & (S23 = 0) & (S27 = 0) & (S28 = 0) => (D = PC48)

Rule 59	(S4 = 1) & (S5 = 1) & (S25 = 1) => (D = PC48)
Rule 60	(S4 = 0) & (S17 = 1) & (S25 = 1) => (D = PC49)
Rule 61	(S15 = 1) & (S16 = 1) & (S25 = 1) => (D = PC49)
Rule 62	(S1 = 1) & (S9 = 1) & (S10 = 1) & (S11 = 0) => (D = PC50)
Rule 63	(S5 = 0) & (S7 = 0) & (S11 = 0) & (S16 = 1 & (S17 = 0) & (S18 = 0) & (S22 = 0) & (S25 = 0) & (S26 = 0) & (S30 = 0) => (D = PC51)
Rule 64	(S5 = 1) & (S15 = 1) => (D = PC52)
Rule 65	(S1 = 0) & (S2 = 1) & (S7 = 1) & (S27= 0) => (D = PC52)
Rule 66	(S1 = 1) & (S3 = 1) & (S15 = 1) => (D = PC52)
Rule 67	(S4 = 0) & (S5 = 0) & (S26 = 1) & (S29 = 1) => (D = PC53)
Rule 68	(S1 = 0) & (S3 = 0) & (S6 = 0) & (S10 = 0) & (S14 = 0) & (S16 = 0) & (S22 = 0) & (S26 = 0) & (S29 = 1) & (S30 = 0) => (D = PC53)
Rule 69	(S28 = 1) => (D = PC54)
Rule 70	(S8 = 1) & (S9 = 1) & (S12 = 0) & (S13 = 1) => (D = PC55)
Rule 71	(S6 = 0) & (S7 = 1) & (S10 = 0) & (S11 = 0) & (S16 = 0) & (S24 = 0) & (S30 = 0) => (D = PC55)
Rule 72	(S2 = 1) & (S8 = 1) & (S9 = 1) => (D = PC55)
Rule 73	(S18= 1) & (S20 = 1) & (S21 = 0) & (S22 = 0) => (D = PC56)
Rule 74	(S3 = 0) & (S19 = 1) & (S20 = 0) & (S23 = 0) => (D = PC56)
Rule 75	(S1 = 0) & (S2 = 1) & (S3 = 0) & (S5 = 0) & (S6 = 1) & (S7 = 0) & (S10 = 0) & (S16 = 0) & (S26 = 0) & (S30 = 0) => (D = PC56)
Rule 76	(S1 = 1) & (S3 = 0) & (S16 = 1) & (S24 = 0) => (D = PC57)
Rule 77	(S1 = 1) & (S6 = 1) & (S7 = 1) => (D = PC57)
Rule 78	(S1 = 0) & (S2 = 0) & (S9 = 0) & (S14 = 0) & (S16 = 0) & (S22 = 0) & (S26 = 0) & (S29 = 1) & (S30 = 0) => (D = PC58)
Rule 79	(S2 = 1) & (S10 = 1) & (S29 = 1) => (D = PC58)
Rule 80	(S10 = 1) & (S11 = 1) & (S12 = 1) => (D = PC59)
Rule 81	(S5 = 0) & (S15 = 0) & (S26 = 1) & (S30 = 0) => (D = PC60)
Rule 82	(S4 = 0) & (S5 = 0) & (S6 = 1) & (S7 = 0) & (S8 = 0) & (S10 = 0) & (S16 = 0) & (S25 = 0) & (S29 = 0) & (S30 = 0) => (D = PC60)
Rule 83	(S15 = 1) & (S16 = 0) & (S25 = 1) => (D = PC61)
Rule 84	(S3 = 1) & (S6 = 1) & (S25 = 1) => (D = PC61)
Rule 85	(S3 = 1) & (S4 = 1) & (S6 = 1) => (D = PC62)
Rule 86	(S4 = 0) & (S15 = 0) & (S21 = 0) & (S26 = 1) => (D = PC62)
Rule 87	(S1 = 1) & (S2 = 1) & (S29 = 1) => (D = PC62)
Rule 88	(S2 = 1) & (S3 = 0) & (S15 = 0) & (S17 = 0) & (S25 = 1) => (D = PC63)
Rule 89	(S4 = 1) & (S6 = 1) & (S25 = 1) => (D = PC63)
Rule 90	(S1 = 0) & (S5 = 0) & (S6 = 0) & (S16 = 0) & (S26 = 1) & (S30 = 0) => (D = PC64)
Rule 91	(S4 = 0) & (S9 = 0) & (S18 = 0) & (S20 = 0) & (S21 = 1) & (S23 = 0) => (D = PC65)
Rule 92	(S5 = 0) & (S7 = 0) & (S11 = 0) & (S20 = 0) & (S21 = 0) & (S24 = 1) => (D = PC65)
Rule 93	(S6 = 1) & (S26 = 1) & (S30 = 1) => (D = PC65)
Rule 94	(S19 = 1) & (S20 = 1) & (S22 = 1) => (D = PC66)
Rule 95	(S1 = 0) & (S4 = 0) & (S15 = 1) & (S29 = 0) & (S30 = 1) => (D = PC67)
Rule 96	(S15 = 0) & (S16 = 0) & (S21 = 0) & (S29 = 0) & (S30 = 1) => (D = PC68)
Rule 97	(S13 = 0) & (S15 = 0) & (S29 = 1) & (S30 = 1) => (D = PC69)
Rule 98	(S6 = 0) & (S7 = 0) & (S10 = 0) & (S11= 0) & (S16 = 1) & (S17 = 0) & (S18 = 0) & (S22 = 0) & (S24 = 0) & (S25 = 0) & (S26 = 0) => (D = PC70)
Rule 99	(S4 = 1) & (S15 = 1) & (S30 = 1) => (D = PC70)
Rule 100	(S4 = 0) & (S5 = 1) & (S26 = 0) & (S29 = 1) => (D = PC71)
Rule 101	(S1 = 0) & (S2 = 1) & (S4 = 0) & (S6 = 1) & (S29 = 1) => (D = PC71)
Approximate Rules	
Rule 102	(S9 = 1) & (S15 = 1) => (D = PC6) or (D = PC7)
Rule 103	(S1 = 0) & (S9 = 1) & (S12 = 0) & (S14 = 1) => (D = PC7) or (D = PC59)
Rule 104	(S13 = 1) & (S15 = 1) & (S29 = 1) => (D = PC10) or (D = PC58)
Rule 105	(S15 = 1) & (S16 = 1) & (S17 = 1) => (D = PC11) or (D = PC12)
Rule 106	(S18 = 1) & (S20 = 1) & (S21 = 1) => (D = PC14) or (D = PC36)
Rule 107	(S1 = 1) & (S4 = 1) & (S13 = 1) => (D = PC17) or (D = PC24)
Rule 108	(S4 = 1) & (S15 = 1) & (S21 = 1) => (D = PC22) or (D = PC25)
Rule 109	(S8 = 1) & (S10 = 1) & (S13 = 0) & (S14 = 0) & (S15 = 0) & (S29 = 0) => (D = PC43) or (D = PC59)

TABLE 13: Decision rules applying the algorithm

The results of Theorem applied in actual data are evident. Information Dependency is apparent for PC45 and PC54. Their Symptoms S27 and S28 respectively is the essential information needed in order to satisfy the Possible Cause.

4.3. Test With Previous Live Data

The Expert System will be inputted with previous live data. It will be used as the Validating data. These data are obtained through retrieval of the information in a live scenario and the Possible Cause is known. It will be inputted in the Expert System. For this research there is a total of 50 live cases.

a.) Enter Previous live Data

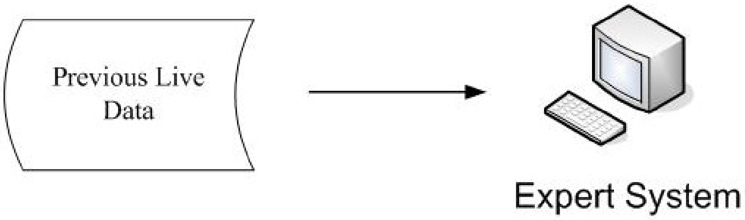


FIGURE 4: Entering of Previous Live Data

b.) Check if the Possible Cause outputted of the Expert System equals to the Possible Cause of the Validating Data

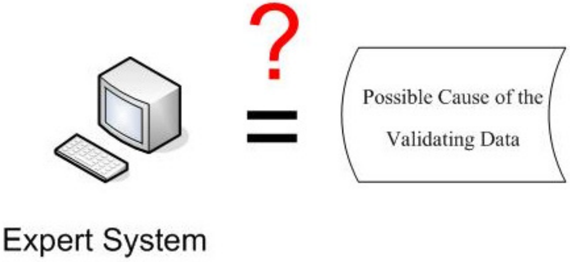


FIGURE 5: Checking of the Expert System's Output

Example in Case 6 which has S8, S9, S10 and S15 as the symptoms, the expected output is PC7. When inputted in the system it gave PC7 as the output same as the expected.

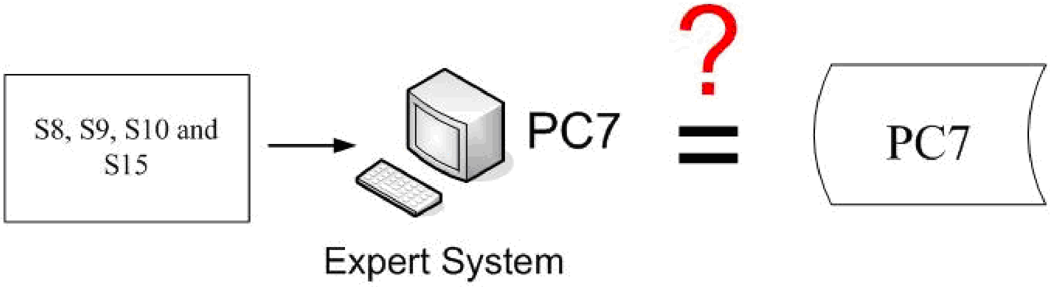


FIGURE 6: Checking of the output of the Expert System in Case 6

c.) Repeat the process for each validating Data. The number of Possible Cause that are outputted correctly out of the total previous live cases will be the score for this test.

Case	Symptoms	System Output	Expected Output
1	S1, S2, S6, S9	PC44	PC47
2	S4, S26, S29, S30	PC69	PC69
3	S4, S15, S30	PC70	PC70
4	S15, S26, S30	PC67	PC67
5	S1, S2, S6, S15, S26, S30	PC65	PC65
6	S8, S9, S10, S15	PC7	PC7
7	S8, S9, S10, S29	PC18	PC18
8	S9, S21, S24	PC29	PC29
9	S2, S7, S9, S18	PC52	PC14
10	S1, S8, S10, S15	PC42	PC42
11	S1, S3, S4, S27	PC45	PC45
12	S1, S9, S10, S12	PC50	PC50
13	S9, S10, S11	PC19	PC19
14	S6, S11, S12, S29	PC58	PC58
15	S18, S23	PC16	PC16
16	S1, S2, S4	PC3	PC3
17	S8, S9, S10	PC59	PC59
18	S1, S2, S6, S29	PC62	PC62
19	S19, S20, S22	PC66	PC66
20	S7, S9, S30	PC68	PC68
21	S16, S30	PC70	PC70
22	S2, S6, S7	PC52	PC52
23	S2, S4, S6	PC56	PC56
24	S15, S16, S17	PC12	PC12
25	S18, S21	PC14	PC14
26	S3, S8, S9	PC20	PC20
27	S4, S8, S16, S17, S26	PC55	PC11
28	S3, S4, S5	PC2	PC2
29	S8, S9, S10, S15	PC6	PC6
30	S11, S12, S16, S29	PC8	PC8
31	S7, S18, S19	PC56	PC40
32	S2, S5, S6	PC43	PC43
33	S2, S4, S15, S17	PC41	PC41
34	S2, S3, S6, S29	PC71	PC71
35	S6, S29, S30	PC69	PC69
36	S18, S19, S20, S22	PC66	PC66
37	S2, S3, S4, S6, S29	PC62	PC62
38	S1, S2, S7, S24	PC31	PC31
39	S15, S16, S18, S23	PC35	PC35
40	S15, S16, S23, S26	PC38	PC38
41	S2, S4, S15, S25	PC61	PC61
42	S2, S3, S6, S25	PC61	PC61
43	S5, S26, S29	PC62	PC62
44	S1, S2, S4, S25	PC63	PC63
45	S2, S29, S30	PC69	PC69
46	S1, S5, S6, S29	PC71	PC71
47	S1, S11, S12, S24	PC29	PC29
48	S1, S2, S5, S16, S24	PC31	PC31
49	S3, S5, S16, S25	PC33	PC33
50	S3, S4, S21, S22, S23	PC34	PC34

TABLE 14: Test with previous live data

Table 14 shows the test done when tested with previous live data. This test gave 46 / 50 or a 92% result and showed the algorithm's competence in previous live data.

4.4. Test With the Experts

The next test is the validation with the experts. Experts in the field of Computer Systems will perform their assessment on the developed Expert System. These experts will suggest and verify the validating data. These data are information on which they already know the Possible Cause from the field of Information and Communications Technology (ICT), Computers and their networking, hardware, firmware, software applications. There are 3 experts and each expert will provide 20 validating Data. In total there will be 60 validating Data. The qualifications of Experts the fields of Computer Systems are:

Expert 1: A Service Engineer from with 3 years experience in the field of Computer Systems. His expertise is Computer Assembly, Software Installations and Operating System diagnostics. His research interests are Computer Hardware and Software upgrades.

Expert 2: A Technical Support Engineer with 4 years experience in the field of Computer Systems. His expertise are Hardware troubleshooting and server farming. His research interests are software development and programming.

Expert 3: A Senior Client System Engineer from a reputable ICT organization. He has 33 years experience in the field of Computer Systems. His expertises are computer operations, facilities management and provisioning. His research interests are Facilities and Section development.

The following is an example on how this process is accomplished.

a.) Expert will enter the validating Data. These Data are cases where they already know the Possible Cause based on previous experience.

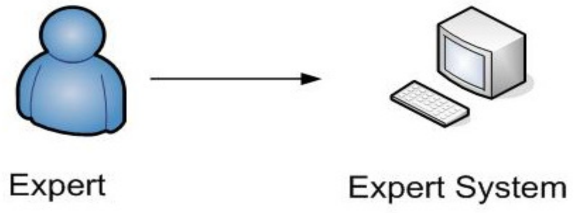


FIGURE 7: Expert entering the validating data

b.) Check if the Possible Cause outputted of the Expert System equals to the Possible Cause of the Validating Data of the Experts.

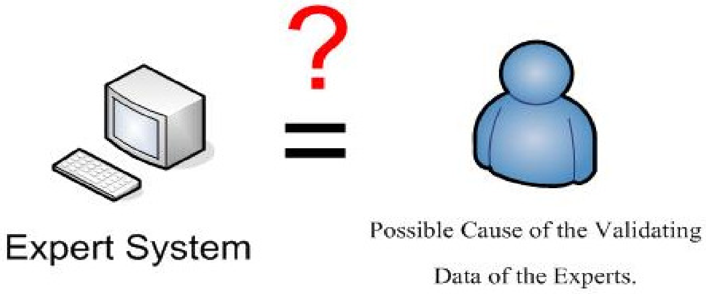


FIGURE 8: Checking of the Expert System's Output to the Possible Cause of the Expert's Validating Data

An Example is in Case 3 which has S18 and S21 as the symptoms. The expected output is PC14. When the Expert inputted those symptoms based on experience the system's output is PC14.

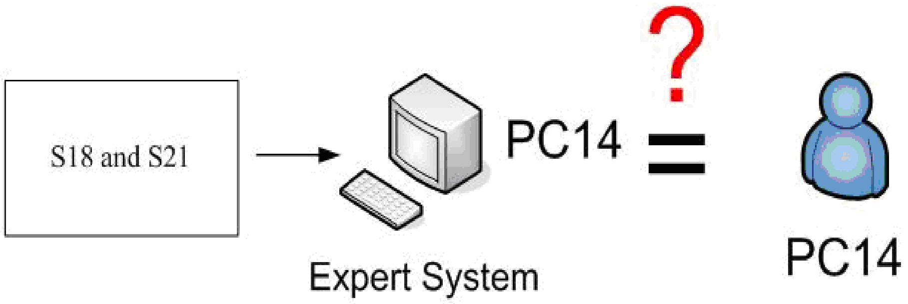


FIGURE 9: Checking of the output of the Expert System in Case 3

c.) Repeat the process for each of the expert's validating Data. The score for the test will be the number of correct answers given by the Expert System out of the total questions asked by the experts.

Case	Symptoms	System Output	Expected Output
	Expert 1		
1	S2, S29, S30	PC69	PC69
2	S11, S12, S16, S29	PC8	PC8
3	S18, S21	PC14	PC14
4	S18, S19, S20, S22	PC66	PC66
5	S8, S9, S10	PC59	PC59
6	S4, S7, S9, S10	PC39	PC66
7	S18, S23	PC16	PC16
8	S16, S30	PC70	PC70
9	S1, S2, S4, S5, S16, S17, S26	PC57	PC57
10	S2, S6, S7	PC52	PC52
11	S6, S11, S12, S29	PC58	PC58
12	S2, S3, S6, S29	PC71	PC71
13	S7, S9, S30	PC68	PC68
14	S4, S8, S9	PC20	PC20
15	S1, S9, S10, S12	PC50	PC50
16	S1, S2, S6, S29	PC62	PC62
17	S1, S3, S4, S27	PC45	PC45
18	S1, S3, S4, S7, S16	PC39	PC39
19	S3, S4, S5	PC2	PC2
20	S8, S9, S10, S15	PC6	PC6
	Expert 2		
21	S2, S4, S17, S25	PC33	PC33
22	S2, S4, S17	PC41	PC41
23	S3, S4, S21, S22, S23	PC34	PC34
24	S4, S9, S12	PC37	PC40
25	S1, S2, S7, S24	PC31	PC31
26	S15, S16, S18, S23	PC35	PC35
27	S15, S26, S30	PC67	PC67
28	S2, S4, S6	PC56	PC56
29	S15, S23, S26	PC64	PC64

30	S9, S21, S24	PC29	PC29
31	S19, S21	PC14	PC14
32	S5, S7, S16, S27	PC45	PC45
33	S2, S3, S6, S25	PC61	PC61
34	S5, S7, S11, S12	PC42	PC42
35	S1, S4, S15	PC39	PC39
36	S15, S16, S23, S26	PC38	PC38
37	S1, S12, S17, S25	PC49	PC49
38	S8, S9, S10, S15	PC7	PC7
39	S1, S2, S5, S16, S24	PC31	PC31
40	S3, S18, S19	PC13	PC13
Expert 3			
41	S1, S2, S6, S15, S26, S30	PC65	PC65
42	S1, S15, S27	PC45	PC45
43	S3, S5, S16, S25	PC33	PC33
44	S5, S26, S29	PC62	PC62
45	S1, S8, S9, S22	PC42	PC25
46	S2, S4, S18, S19	PC56	PC56
47	S15, S16, S17	PC11	PC11
48	S1, S2, S15, S26	PC47	PC47
49	S2, S3, S6	PC60	PC60
50	S20, S21, S22	PC36	PC36
51	S1, S5, S8, S11, S12, S22	PC24	PC24
52	S1, S2, S4, S25	PC63	PC63
53	S1, S5, S6, S29	PC71	PC71
54	S3, S7, S8, S15, S16	PC39	PC47
55	S4, S15, S30	PC70	PC70
56	S8, S11, S12	PC24	PC24
57	S6, S29, S30	PC69	PC69
58	S19, S22, S23	PC35	PC35
59	S22, S29	PC15	PC15
60	S4, S15, S26	PC64	PC64

TABLE 15: Test with the validating data

Table 15 shows the test with the validating data by the experts. This test gave 56 / 60 or a 93.3% result and showed the algorithm’s competence when tested with the experts.

5. ANALYSIS AND CONCLUSIONS

The research has presented, analyzed and tested a new Expert System Algorithm. The algorithm shows a novel technique to input, tag, and properly structure technical so they can be converted into the rules of an Expert System. The rules created from the algorithm are nominal in terms that only the necessary information needs to be inputted to satisfy the Possible Cause. In cases where the Data gathered is incomplete, the proper conclusion may still be suggested. A theorem is proposed on Information Dependency of data, the essential information needed in order to obtain the correct Possible Cause. A formal proof of the theorem was presented and its correctness was tested on live data. It is very vital and useful in large Information Systems. Knowing which Data is needed will not only save time in the processing of information but also conserve resources.

A future recommendation for this research is for it to be tested in other fields. This research’s scope is only for Computer Systems. In theory the theorems and algorithms can be applied in several Production Systems like in Medical diagnosis.

6. REFERENCES

[1] G. Jeon, M. Anisetti, D. Kim, V. Bellandi, E. Damiani, J. Jeong. “Fuzzy rough sets hybrid scheme for motion and scene complexity adaptive de interlacing”. Image and Vision Computing Journal, 27(4) 452-436, 2009.

[2] S. Patchararungruang, K. Halgamune, N. Shenoy. “Optimized rule-based delay proportion adjustment for proportional differentiated services”. IEEE Journal on selected areas of Communication. 23(2) 261-276, 2005.

[3] Y. Chang, C. Yang. “A complementary approach to data broadcasting in mobile Information Systems”. Data and knowledge Engineering. 40(2) 181-184, 2002.

[4] S. Wang. “Conceptual construction incomplete survey data”. Data and Knowledge Engineering. 49(3) 311-323, 2004.

[5] H. Akcan, A. Astasyn, H. Bronnimann. “Deterministic algorithms for sampling count data”. Data and knowledge Engineeriing. 64(2) 405-418, 2008.

[6] M. Winget, J. Baron, M. Spitz, D. Brenner, D. Kincaid, M. Thornquist, Z. Feng. “Development of common data elements: the experience and recommendations from the early detection research network”. International Journal of Medical Informatics. 70(1) 41-48, 2003.

[7] E. Borrowski, J. Borwein. Collins Dictionary of Mathematics. Springer Verlag. 1989.

[8] W. Ziarko, N. Wojciech. “Rough Set methodology for Data Mining”. Discovery 1: Methodology and applications. 554-576. 1998.

[9] I. Gelman. “Setting priorities for data accuracy improvements in satisfying decision making scenarios”. Decision Support Systems. 48(4) 507-520, 2010.

[10] S. Wong, S. Hamouda “The development of online knowledge-based expert system for machinability data selection”. Knowlede BasedSystems. 16(4) 215-229, 2003.

[11] J. Sheu, P. Sahoo, C. Su, W. Hu. “Efficient path planning and gathering protocols for wireless sensor network” Computer Communications. 33(3) 398-408, 2010.

[12] C. Wu, X. Wu, L. Wang, Y. Pan. “Knowledge Dependency Relationships in Incomplete Information System Based on Tolerance Relations”. IEEE International Journal on Systems and Cybernetics Conference, 2006.

[13] ROSE 2.0, http://www.idss.cs.put.poznan.pl/rose ,1999.

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 5, 2011, 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:** 2 - April 2012

i. Paper Submission: January 31, 2012 **ii. Author Notification:** March 15, 2012

iii. Issue Publication: April 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

CSC PUBLISHERS © 2011
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