Performance Study and Analysis towards Discrete System Introducing Jury Test Simulator

Biswarup Neogi, Susmita Das, Swati Ghosh, Nilotpal Manna & Swapan Bhattacharyya
JIS College of Engg
WB, India
biswarupneogi@gmail.com, susmitad2011@gmail.com, mistiswati@gmail.com, nmanna324@gmail.com,
Tanmoy Singha
Modern Institute of Engg. & Technology
WB, India;
tanmoy.singha@mietcollege.org

ABSTRACT

Jury stability analysis technique is one of the best suited tools in z domain and it is being applied in industrial and research fields to determine the stability of discrete system. This paper presents the proper Jury stability analysis algorithmic approach and also the advanced Jury’s test simulator effectively. Here the stability of a discrete intelligent control system stability analysis is presented through the simulation software and this presentation will be potential enough to ascertain the stability in z domain. It is an intelligent system in the domain of Computer Aided Control System Design (CACSD). For the betterment of the performance of the system the worst case time complexity and space complexity of the algorithm is given with big oh, theta and big omega notation. The Jury test software support approach and mainly engineering based advancements are focused in this study.

Keywords- Jury stability, Advance Jury Test Simulator algorithm, Time complexity, Space complexity, Simulating software.

I. INTRODUCTION

Applications of Jury stability test in the field of electronics industry have a great position. The problem of determination of the root distribution problem of a real polynomial with respect to the unit circle, in terms of the coefficients of the polynomial, was solved by Jury (1964). The calculations were presented in tabular form (Jury's table). This result is now classical and is as important in the stability analysis of digital control systems as its continuous time counterpart. The Jury’s Technique, a potential and generally accepted method can make available appropriate knowledge on the stability of a digital system In case of Biological system the output response is necessarily non deterministic. In such a system the output may be sampled and digitized accordingly [1]. Then, utilizing the application of the Jury’s technique system, the stability can be easily projected. Technologists have distinguished these special cases in which the stability of biological system can be established by the application of the Jury’s stability testing. There are several applications of Jury’s stability testing. Microcontroller-Based Temperature Monitoring and Control is an essential and practical guide for all engineers involved in the use of microcontrollers in measurement and control systems [2].

The application of Jury stability test is the research on high-order accurate finite-difference time-domain (FDTD) method for the solution of Maxwell's equations is very popular [3]. Determination of root distribution of univariate polynomials with real or complex-valued coefficients, the Bistritz tabular form offers a significant computational advantage. Stability studies of two-dimensional (2-D) discrete-time systems involve univariate polynomials possessing parameter-dependent coefficients, where the parameter takes values on the unit circle in the complex plane. The only use those coefficients of the characteristic polynomial equation for stability analysis instead of directly solving it. Such treatment can greatly simplify the procedure of stability analysis. The Jury's Stability Test can be used to analyze the stability of the system without explicitly solving for the poles of the system. Stability analysis of microcontroller-based real-time systems is the one of the important application of Jury test [4]. Therefore, it is used to determine the bounds on the parameters which result in a stable transfer function in the z-domain. Three-dimensional (3-D) signal processing offers many advantages over two-dimensional (2-D) processing, because it preserves 3-D correlations. In this paper the design and the stability of 3-D rotated filters are considered.
These filters are designed by rotating a one-dimensional (1-D) digital filter in 3-D space. The rotated filters are valuable in the design of various 3-D filters which possess prescribed spectral specifications. An efficient algorithm for the design of 3-D low pass (LP) digital filters, with approximately spherically symmetric magnitude responses, is introduced [5]. To achieve the desirable spectral characteristics, a number of 3-D rotated filters are cascaded. The stability of the spherically symmetric filters designed is considered, and stable realizations are proposed. The stability of a linear transfer function is fundamental in its real time realization. Several tests have been developed in the past to test whether a given transfer function is stable. Invariably these tests rely on the Jury-test.

The Jury test in automatic control theory is introduced, which only use those coefficients of the characteristic polynomial equation for stability analysis instead of directly solving it. Such treatment can greatly simplify the procedure of stability analysis. This type of treatment can greatly simplify the procedure of stability analysis. Numerous applications such as air-traffic handling, missile interception, and anti-submarine warfare require the use of discrete-time data to predict the kinematics of a dynamic object are based on Jury’s stability test. The uses of passive sonobuoys which have limited power capacity constrain us to implement target-trackers which are computationally inexpensive. Based on Jury stability test, an Advanced Jury Test Simulator (AJTS) is developed. Also the emphasis is given on performance analysis of the AJTS software.

2. CLASSICAL OVERVIEW OF JURY’S STABILITY

The root detection of a characteristic polynomial \( F(z) = 1 + G(z) H(z) \) inside the unit circle for resolve of system stability of a sampled-data control system is performed by Jury’s stability criterion[4]. Considering the following nth order polynomial in

\[
F(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n} > 0
\]  

(1)

Necessary conditions for the roots of the above polynomial \( F(z) \) to lie within a unit circle centered at the origin of z-plane are: \( F(1) > 0 \) and \((-1)^n F(-1) > 0\), \( a_n < a_0 \) Schwarz matrix for the discrete case was developed by Jury. In the above mentioned array, the elements of 2nd, 4th, 6th etc. up to \( (2^n-4) \) rows are respectively the elements of 1st, 3rd, 5th…. etc. up to \( (2^n-5) \) rows in reverse order. The elements of 3rd, 5th…. etc. up to \( (2^n-3) \) rows are determined by following second order determinants.

\[
\begin{align*}
    b_k &= \text{det} [a_0, a_{n-1}; a_n, a_k] \quad \text{for 3rd row, } k=0,1\ldots(n-1) \\
    c_k &= \text{det} [b_0, b_{n-1-k}; b_n, b_k] \quad \text{for 5th row, } k=0,1\ldots(n-2) \\
    d_k &= \text{det} [c_0, c_{n-2-k}; c_n, c_k] \quad \text{for 7th row, } k=0,1\ldots(n-3) \\
    q_1 &= \text{det} [p_3, p_0; p_0, p_3] \quad \text{and} \\
    q_0 &= \text{det} [p_3, p_2; p_0, p_3] \quad \text{for } (2^n-3) \text{ row}
\end{align*}
\]

Sufficient conditions for the roots of the characteristic polynomial \( F(z) \) to lie within a unit circle in z-plane are specified by:

\[
\begin{align*}
    |a_0| &< |a_n| \\
    |b_0| &>|b_{n-1}| \\
    |c_0| &>|c_{n-2}| \\
    |p_3| &>|p_0| \\
    |p_2| &>|q_0| \quad \text{with } (n-1) \text{ constraints.}
\end{align*}
\]
3. PRIMARY ALGORITHM & FLOWCHART
GENERATION OF THE ADVANCE JURY
TEST SIMULATOR (AJTS)

ALGORITHM:

STEP 1: START THE PROGRAM

STEP 2: READ THE VALUE OF ‘n’ (ORDER OF THE POLYNOMIAL)

STEP 3: REPEAT STEP 4 FOR n (1) 0

STEP 4: READ THE CONSTANT COEFFICIENT \(a_i\), i = n(1) 0

STEP 5: BUILD THE FUNCTION \(F(Z) = \sum a_iz_i\), i = n (1) 0

STEP 6: COMPUTE \(F(1), (-1)^nF(-1), |a_0|\)

STEP 7: CHECK \(F(1) > 0, (-1)^nF(-1) > 0\)

STEP 8: IF STEP 7: SATISFIED THEN GOTO STEP 9 ELSE GOTO STEP 14

STEP 9: REPEAT STEP 10: UPTO ‘2n-3’ ROW

STEP 10: COMPUTE

\((a_0 * a_k) - (a_n * a_{n-k})\) As \(b_k\) FOR \(k = 0 \ (1) (n-1)\)

\((b_0 * b_k) - (b_{n-1} * b_{n-1-k})\) As \(c_k\) FOR \(k = 0 \ (1) (n-2)\)

\((c_0 * c_k) - (c_{n-2} * c_{n-2-k})\) As \(d_k\) FOR \(k = 0 \ (1) (n-3)\)

\vdots

\((p_{2n-4} * p_{2n-4-k})\) As \(q_k\) FOR \(k = 0 \ (1) (2n-3)\)

STEP 11: CHECK

\(|a_0| < |a_n|,\)

\(|b_0| > |b_{n-1}|,\)

\(|c_0| > |c_{n-2}|,\)

\(|q_0| > |q_{2n-4}|\)

STEP 12: IF STEP 11: SATISFIED THEN GO STEP 13: ELSE GO TO STEP 14


STEP 14: PRINT: THE REQUIRED NECESSARY AND SUFFICIENT CONDITIONS ARE NOT SATISFIED, HENCE THE SYSTEM IS NOT STABLE STOP THE PROGRAM.

STEP 15: STOP
The Jury's Technique would be more advantageous than the existing Davis' Method in many ways. In the first place, the stability test in z-domain can be established with an algorithmic approach, following Jury's Technique something that cannot be achieved in the Davis' System. Due to introduction of too many variables, make the Davis' System complicated and also risk of errors would be moved up. Jury's Technique, on the contrary, provides a simpler solution by an array representation. The Jury's Technique is far more exact with stability preserving method being its priority. Comparing with the existing recent developments for discrete system's [7] stability test, the new method is more straight-forward than Harn and Chen’s work. Also, this method is much easier than the continued fraction method.

3. FLOWCHART OF THE JURY TEST SIMULATING SOFTWARE
4. FRONTEND REPRESENTATION OF ADVANCE JURY TEST SIMULATOR (AJTS) WITH SUCCESSFUL TESTING

In order to test the efficacy of the technique a ninth order transfer function has been considered, which is the standard transfer function of a biological prosthetic arm in z domain [6].

\[
T(z) = \frac{C(z)}{R(z)} = \frac{0.0000z^2 + 0.0222z + 0.0143z^9 + (-1.8441)z^8 + 0.6365z^7 + 0.5541z^6 + 0.3747z^5 + 0.0054z^4 + 0.5070z^3 + (-0.6453)z^2 + 0.0841z + 0.1138}{z^9 + (-1.8441)z^8 + 0.6365z^7 + 0.5541z^6 + (-0.3747)z^5 + 0.0054z^4 + 0.5070z^3 + (-0.6453)z^2 + 0.0841z + 0.1138}
\]

\[
\text{Characteristic polynomial: } F(z) = z^9 + (-1.8441)z^8 + 0.6365z^7 + 0.5541z^6 + (-0.3747)z^5 + 0.0054z^4 + 0.5070z^3 + (-0.6453)z^2 + 0.0841z + 0.1138
\]

Figure 1. Design view for entering the order of polynomial

Figure 2. Design view for entering the co-efficient of polynomial
Algorithm Efficiency is measured through complexity analysis. Algorithm has both time and space requirements called complexity. The main issues related to the efficiency of algorithms are Speed of algorithm (determined by the number of elementary operations: addition, subtraction, multiplication, division, comparison and the number of elementary operations are dependent on problem size & nature of input data) and Efficient memory allocation. Time complexity (Time required for executing the algorithm) and Space complexity (How much memory space is required to execute the algorithm?) are the two types of complexity. The measuring of either time/space complexity of an algorithm. Measuring of time complexity is more important. Actual time can’t be computed for the algorithm. The function of problem size that is directly proportional to time requirement called growth-rate function. How the time requirement grows as the problem size grows, measured by this function. Usually worst-case time complexity is considered. Worst case efficiency is the maximum number of steps that an algorithm can perform for any collection of data values. Best case efficiency is the minimum number of steps that an algorithm can perform for any collection of data values. Average case efficiency is the efficiency averaged on all possible inputs and a distribution of the input must be assumed. It is also a uniform distribution supposed to be assumed (all keys are equally probable) and if the input has size $n$, efficiency will be a function of $n$.

5.1 "Big Oh" - Upper Bounding Running Time:
For the Asymptotic Big O notation,

Let $f$ and $g$ be two functions such that

$$f(n): N \rightarrow R^+ \text{ and } g(n): N \rightarrow R^+$$

if there exists positive constants $c$ and $n_0$ such that

$$f(n) \leq cg(n), \text{ for all } n > n_0$$

or if $0 \leq \lim f(n)/g(n) = c < \infty$ where $n \to \infty$

Then we write $f(n) = O(g(n))$.

Figure 3. Design view after entering the co-efficient
So \( g(n) \) is an asymptotic upper-bound for \( f(n) \) as \( n \) increases (\( g(n) \) bounds \( f(n) \) from above)[8]. \( cg(n) \) is an approximation to \( f(n) \), bounding from above "Big Oh" - Upper Bounding Running Time.

\[
\text{Figure 4: Asymptotic upper-bound for } f(n)[11]
\]

5.2 "Theta" - Tightly Bounding Running Time:
Let \( f \) and \( g \) be two functions such that \( f(n) : N \rightarrow R^+ \text{ and } g(n) : N \rightarrow R^+ \)

if there exists positive constants \( c_1, c_2, \) and \( n_0 \) such that 
\[
c_1 g(n) \leq f(n) \leq c_2 g(n), \text{ for all } n > n_0
\]
or if \( 0 \leq \lim_{n \rightarrow \infty} f(n)/g(n) = c < \infty \) where \( n \rightarrow \infty \) then we write \( f(n) = \Theta(g(n)) \)

So \( g(n) \) is an asymptotic tight-bound for \( f(n) \) as \( n \) increases[9] 
\( f(n) = \Theta(g(n)) \)

\[
\text{Figure 5: Asymptotic tight-bound for } f(n)[12]
\]

5.3 Big-Omega Notation:
Let \( f \) and \( g \) be two functions such that \( f(n) : N \rightarrow R^+ \text{ and } g(n) : N \rightarrow R^+ \)

if there exists positive constants \( c \) and \( n_0 \) such that 
\[
f(n) \geq cg(n), \text{ for all } n > n_0
\]
or if, $\lim_{n \to \infty} \frac{f(n)}{g(n)} = c < \infty$ where $n \to \infty$ [10] then we write $f(n) = \Omega(g(n))$

So $g(n)$ is an asymptotic lower-bound for $f(n)$ as $n$ increases
($g(n)$ bounds $f(n)$ from below)

$f(n) = \Omega(g(n))$

Figure 6: Asymptotic lower-bound for $f(n)$[12]

6. MEASUREMENT OF THE WORST CASE TIME COMPLEXITY OF JURY’S STABILITY TEST ALGORITHM

6.1 Computation of worst case Time complexity of Algorithm:

Constant time is considered to read variable $n$. So, $O(1)$ time is required

Constant time $O(1)$ is required for the statement

Set $I = n$ ....... (1)

The statement $i \geq 0$ is executed $(n+1)$ times.

= $n$. $O(1)$ + $O(1)$

= $O(n)$ .....

The statement Read $a_i$ is executed $(n+1)$. $p1 + (t_n + t_{n-1} + t_{n-2} + \ldots + t_1 + t_0)$ times.

I.e. $(n+1).p1 + \sum t_i$

= $(n+1).p1 + O(1) + O(1) + \ldots + O(1)$ (n+1) times.

= $(n+1).p1 + O(1)$

= $n.(p1+O(1)) +1.(p1+O(1))$

= $O(n)$ .....

[Since, $p1$ time (constant time) is required for each Read statement. Read statement repeatedly executed $(n+1)$ times. So, total $(n+1).p1$ times are required. Secondly, $t_i$ constant time is required for each $a_i$. For $i=n, n-1, \ldots, 1, 0$. [ $t_i$ time is required for $a_n$. $t_{n-1}$ time is required for $a_{n-1}$. . . $t_0$ time is required for $a_0$. All $t_i$ $(i=n,n-1,\ldots,1,0)$ are constant.]

The statement $i = i - 1$ is computed in following ways.

Let $T(n)$ denotes the time for decreasing from $n$ to $-1$ and $T(n-1)$ be the time for decreasing from $(n-1)$ to $-1$. So, a recurrence relation is formed as follows:

$T(n-1) = T(n) - 1$

$T(n) = T(n-1) + 1$

= $(T(n-2) + 1) + 1$

= $T(n-3) + 1 + 1 + 1$

= $T(n-n) + 1 + 1 + \ldots + n$ times.

= $T(0) + n$ since, $T(0)=1$

= $1 + n$

= $O(n)$ .....

$T(n)$ time is required to compute $F(1) = a_n + a_{n-1} + a_{n-2} + \ldots a_0$

Therefore ,

$T(n) = T(n-1) + t_n$
\[ T(n) = T(n-1) + C_n \]
\[ = T(n-1) + C_{n-1} + C_n \]
\[ = \ldots + C_1 + C_0 + T(0) \]
\[ = O(n) \] \hspace{1cm} \text{(6)}

To compute \((-1)^n F(-1) = a_n - a_{n-1} + \ldots + (-1)^n a_0\)

Firstly, it is needed to compute \((-1)^{n-i}\) for \(i = n \) to 0.

So, \((n+1) \cdot p\) times are required.

Therefore, time is required to compute \((-1)^{n-i}\) for \(i = n\) to 0.

\[ = O(n) \] \hspace{1cm} \text{(7)}

where \(p\) is constant time taken to compute \((-1)\).

Then to compute \(\sum (-1)^{n-i} a_i\) for \(i = n\) to 0, where \(T(n)\) time required to add \((n+1)\) elements \([\text{starting from } a_n \text{ to } a_0]\)

\(C_n\) time is required to add \((-1)^n a_n\), \(C_{n-1}\) time is required to add \((-1)^{n-1} a_{n-1}\), \(C_1\) time is required to add \((-1)^0 a_1\) and \(C_0\) constant time is required to add \((-1)^{0} a_0\)

Therefore, \(T(n) = T(n-1) + C_n\)
\[ = T(n-1) + C_{n-1} + C_n \]
\[ = \ldots + C_1 + C_0 + T(0) \]
\[ = O(1) + O(1) + \ldots + O(n) \]
\[ = O(n) \] \hspace{1cm} \text{(8)}

To compute \(|a_0|\) and \(|a_n|\) \(O(1)\) time can be required. \hspace{1cm} \text{(9)}

Check if

\(F(1) > 0\) and \((-1)^n F(-1) > 0\) \& \(a_0 < |a_n|\), \(O(1) + O(1) + O(1) = 3\). \(O(1)\) time is considered \hspace{1cm} \text{(10)}

constant time \(O(1)\) is required to set \(I = n\) and to test \(I < 2n - 3\) respectively \hspace{1cm} \text{(11)}

for each operation \(b_k = (a_0 * a_k) - (a_n * a_{n-k})\), \(P_k\) time is required for \(k = 0, 1, 2, \ldots, n-1\). [each \(P_k\) is constant time.]

So, the time is mandatory to compute \(n\) operations = \(P_0 + P_1 + \ldots + P_{n-1} = \text{O}(n)\). \(O(1)\)

Similarly, for \(c_k = (b_0 * b_k) - (b_{n-1} * b_{n-1-k})\) operations,

\((n-1)\).O(1) time is required for \(d_k = (c_0 * c_k) - (c_{n-2} * c_{n-2-k})\) operations, \((n-2)\).O(1) time is required.

\hspace{1cm} \text{for each operation } b_k = (a_0 * a_k) - (a_n * a_{n-k}), \text{ each } P_k \text{ is constant time.}

\(O(1)\) time is required to test \(I < 2n - 3\) respectively for each operation \(b_k = (a_0 * a_k) - (a_n * a_{n-k})\).

\(O(1)\) time is required to set \(I = n\) and to test \(I < 2n - 3\) respectively for each operation \(b_k = (a_0 * a_k) - (a_n * a_{n-k})\).

\(O(1)\) time is required for \(d_k = (c_0 * c_k) - (c_{n-2} * c_{n-2-k})\) operations, \((n-2)\).O(1) time is required.

\hspace{1cm} \text{for each operation } b_k = (a_0 * a_k) - (a_n * a_{n-k}), \text{ each } P_k \text{ is constant time.}

\(O(1)\) time is required to test \(I < 2n - 3\) respectively for each operation \(b_k = (a_0 * a_k) - (a_n * a_{n-k})\).
To compute following statements
\[ |a_0| < |a_n|, \]
\[ |b_0| > |b_{n-1}|, \]
\[ |c_0| > |c_{n-2}|, \]
\[ \ldots \ldots \ldots \ldots \ldots \]
\[ |q_0| > |q_2|, \]

Required time = \( O(1) + O(1) + O(1) + \ldots \) up to \((n-1)\) times.
= \((n-1).O(1)\)
=\(O(n)\)  \hspace{1cm} (13)

To get total time in worst case it is need to aggregate \((1), (2), (3), \ldots \ldots (13)\).
\[
\text{Time}_{\text{worst}} = O(1) + O(n) + O(n) + O(n) + O(n) + O(n) + O(n) + O(n) + O(n) + O(n^2) + O(n^2) + O(n^2) + O(n)
\]
= \( O(n^2) \)

7. MEASUREMENT OF SPACE COMPLEXITY OF THIS JURY’S STABILITY TEST ALGORITHM

The space complexity of an algorithm \(X\) can be defined as \(S(X) = C(X) + I(X)\) where \(C(X)\) is required constant space whereas, \(I(X)\) is the instantaneous space requirement for algorithm \(X\).

Here, \(n, I, F(1), z, k\) \([\text{where } z \text{ denotes } (-1)^n \text{ or } F(-1)\] \) are constant variable.
If the space required for a single constant variable is \(k\) bytes then the total space required for all the constant variables used in this algorithm is \(C(X) = \text{space requirement of } (n, I, F(1), z, k) = (k + k + k + k + k)\) bytes = \(5k\) bytes. The space required for array \(a[\] \) depends on \(n\) and \(n\) is either equal to or less than array size. So, space required for the array is at least \((n \cdot k)\) bytes, if \(k\) bytes of space are required for each array content. Similarly, the space required for array \(b[\], c[\], \ldots q[\] \) depends on \((n-1), (n-2), \ldots 2\) respectively. So, the space required for array \(b[\], c[\], \ldots q[\] \) is at least \((n-1) \cdot k, (n-2) \cdot k, \ldots 2k\) respectively.

So, the space complexity of this algorithm is \(S(X) = 5k + n \cdot k + (n-1) \cdot k + (n-2) \cdot K + \ldots + 2k\)
= \(5k + (n + n-1 + n-2 + \ldots + 2) \cdot k\)
= \(5k + [n \cdot (n+1)/2 - 1] \cdot k\)
= \((n^2 + 8) \cdot k\) bytes.

8. CONCLUSION

The programming for the system is done with Visual Basic technique which could be universally applied for any digitized system transfer function. However, it should be mentioned here that with the increment of degrees of the characteristic equation, the time-space complexity enhances. For optimization of the application for larger degrees of characteristic equation the authors are keen to conduct further researches on the problem so that it becomes more simplified than now. In this manner, this application software working on visual basics platform, of Computer Aided Control System Design (CACSD) serves as a bridge between Control System and Computer based Algorithm. The speed of this algorithm is computed in terms of time and space complexity.

ACKNOWLEDGEMENT

The authors would like to express their gratitude to the TEQIP phase-II, Sub-component 1.2, JIS College of Engg., Kalyani for supporting the research work financially. Additionally authors are very much thankful to ECE & EIE Dept. for their cooperation.
REFERENCES


[12] INTRODUCTION TO ALGORITHMS (English) 3rd Edition by Al. Cormen, Publisher: WILEY INDIA,2009.

AUTHORS’ BRIEF

Dr. Biswarup Neogi is awarded PhD (Engineering) from Jadavpur University, India. He received M.Tech degree in ECE from Kalyani Govt. Engg. College in 2007. Before that He obtained B.E in ECE fromUIT, The University of Burdwan in 2005. He has a experience on various project of All India Radio attach with the Webel Mediatronics Ltd, Kolkata. He was a lecturer in ECE Dept, Haldia Institute of Technology, WB, India. He was working as a faculty in ECE Dept, Durgapur Institute of Adv. Tech. & Mgmt. Currently he is engaged with JIS College of Engineering, Kalyani as a Faculty member and flourished R&D related activity. He is also engaged as a consultant executive engineer of YECOES Ltd, Hooghly, and attached as an advising body of different Engineering College under WBUT. Several Government Funded project have been coordinated by him as a Personal Investigator. He is successfully registered one Indian Patent about Odour Effect Creation. His research interest includes Prosthetic Control, Biomedical Engineering, Digital Simulation, Microcontroller based Embedded System. He is guiding five Ph.D theses in this area. He has published about fifty several papers in International and National Journal and Conference conducted both in India and abroad. Additionally, he attached as a reviewer of several journals and yearly conference.

Ms. Susmita Das is presently engaged as an Assistant professor of Electronics & Instrumentation Engg. Dept. in JISCE, Kalyani. She is awarded the degree of M.Tech in Instrumentation & Control Engg. from Calcutta University (UCSTA) in 2011. Before that she received the B.Tech degree in Electronics and Instrumentation Engg. Dept. from WBUT in 2008. She has been engaged as a successful examiner of WBUT curriculum. Presently she initiates her Ph.D. curriculum (Engineering) under the supervision of Dr. Biswarup Neogi. Her research interest includes Control Theory, Process control, Biomedical Engineering etc.
Mr. Tanmoy Singha received the B.Tech degree from I.M.P.S C.E.T Malda in Electronics & Communication Engineering in 2008 & M.E degree from Jadavpur University in ETCE Department in Control Systems Engineering specialization in 2012.In 2008, he joined R.P.B.M.J.C.E.T, Jiaganj as a Lecturer in ETCE Department and in the year 2010, he becomes H.O.D & Teacher-In-Charge. After that he joined K.I.E.M, Mankar as an Assistant Professor (ECE) in 2012. In 2013, he joined M.I.E.T, Bandel as an Assistant Professor in ECE Department and still continuing. His research interests include evolutionary optimization, machine learning, artificial intelligence, fuzzy and control system. He has more than 6 years of Teaching Experience. He has published 01 paper in FICTA 2013 (Springer).

Ms. Swati Ghosh, M.Tech final year student in Mobile Communication & Network Technology (MCNT) under WBUT. Before that she received the B.Tech degree in Electronics and Communication Engineering from WBUT in 2006. Presently working as a Teaching Assistant at JIS College Of Engineering, Kalyani in the Department of Electronics and Communication Engineering since 2013. She has experienced in various companies like, CESC Ltd. and various IT MNCs. She has worked with various Software development and Testing platforms in IT industry and also experienced in designing and implementation of Software Systems. Her domain of work at the time of B.Tech degree was Embedded Technology and Microcontroller. Presently she initiates his M.Tech final year project under the supervision of Dr. Biswarup Neogi. Her major work is in the domain of Micro Controller based Embedded System and Tactile-Display.

Mr. Nilotpal Manna obtained B.E. degree in Electronics and Telecommunication Engineering in 1979 from Bengal Engineering College, Sibpore, Kolkata, now renamed as Bengal Engineering and Science University and received M Tech degree in 1981 from Indian Institute of Technology Madras (Chennai). He has wide industrial experience of twenty-two years from semi-government sectors like Instrumentation Ltd, Kota and several private industries like Toshniwal Instruments Manufacturing Pvt Ltd and others. He served mostly in the Research and Development wings and was associated in development of various electronic and communication instruments meant for military application as well as development of analytical instruments.

At present he is Head of the Department of Electronics and Instrumentation Engineering of JIS College of Engineering, Kalyani, West Bengal, India. He has several research publications in national and international journals and conferences, and authored four technical books.

Swapan Bhattacharyya (M’06) received his B.Sc. (Hons.) degree in Physics from the University of Calcutta in 1982, the Integrated M. Tech. (B. Tech & M. Tech) degree in Radio Physics and Electronics in 1987 and the Ph. D. degree, all from the University of Calcutta. In 1988, he joined the Tata Group of Companies (TGC) at Jamshedpur, India as a Senior Customer Support Engineer, where he became the Branch Manager (Customer Support) in 1998 and in the next year he joined the Corporate Office-Mumbai as a Corporate Manager (Customer Support). In May 2004, he joined the Asansol Engineering College (India) as a System Manager, where he is the Head of the Department of Computer Science and Engineering since December 2004. At present he is the Head of the Department of Electronics & Communication Engineering at JIS College of Engineering (An Autonomous Institute). His field of research in the area of Semiconductor nanostructures, Quantum dots for information and communication applications. He has also special interest in natural language processing, and use of IT for rural development. From January 2007 to June 2007 he worked as a visiting faculty at ICFAI Institute, India.

Dr. Bhattacharyya was a recipient of a National Scholarship. He worked in academic administration of several reputed technical institute and universities in India. He has published many technical research papers in international journals, in national and international conferences. He also served as a conveyer of technical committee for several national conferences and seminars. He is a member of IEEE, Photonsics Chapter and life member of IE (Institute of Engineers, India).