



Accurate correction of Motion Artifacts in Medical Image

M.V.Sruthi *
JNTU Anantapur,
Anantapur
India.

Dr.K.Soundararajan,
JNTU Anantapur,
Anantapur,
India.

Dr.V.Usha Shree
JBIET ,
Hyderabad,
India.

Abstract- Respiratory Motion during the scan causes motion artifacts, in order to reduce these artifacts here is an algorithm called Pixel-specific back projection. PSBP is an approximate filtered back-projection algorithm that corrects for in-plane motion by performing the reconstruction using a coordinate system that is specific to each pixel. To reduce artifacts caused by respiratory motion in chest CT scans, an algorithm called pixel specific back-projection (PSBP) has been developed. PSBP is an approximate filtered back-projection algorithm that corrects for in-plane motion by performing the reconstruction using a coordinate system that is specific to each pixel. The coordinate systems move according to the in-plane motion in the slice at the time the projection was acquired. PSBP reduced artifacts caused by motion in both simulated and patient scan data. The coordinate systems move according to the in-plane motion in the slice at the time the projection was acquired. Depending on the intensity of the pixels this algorithm works.

Keywords: Artifacts, back projection, PSBP, Warping

1. INTRODUCTION

Cardiac and respiratory motion can cause artifacts in computed tomography scans of the chest. These artifacts are clinically significant because they may obscure pathology or mimic disease. These noise, or artifact, sources include: line noise from the power grid, eye blinks, eye movements, heart beat, breathing, and other muscle activity. Some artifacts, such as eye blinks, produce voltage changes of much higher amplitude than the endogenous brain activity. In this situation the data must be discarded unless the artifact can be removed from the data. Crawford introduces an alternative algorithm called filtered back-projection. In this algorithm time-varying magnification motion model (TVMBP) was used. But this algorithm was not effective because the time-varying magnification model did not accurately describe in-plane respiratory motion nor did it account for cross-plane motion.

In this research further improvement of the algorithm is done. We are using Pixel-specific back-projection algorithm which implements time-varying magnification motion model on a pixel-by-pixel basis. Reconstruction is performed in a coordinate system specific to each system. By using this model we can minimize the motion artifacts in an image.

i) Eye Blink artifact: It is very common in EEG data, produces a high amplitude signal that can be many times greater than EEG signals of interest. Because of its high amplitude an eye blink can corrupt data on all electrodes, even those at the back of the head. Eye artifacts are often measured more directly in the electrooculogram (EOG), pairs of electrodes placed above and around the eyes. Unfortunately, these measurements are contaminated with EEG signals of interest and so simple subtraction is not a removal option even if an exact model of EOG diffusion across the scalp is available

ii) Eye Movement: These artifacts are caused by the re orientation of the retino corneal dipole [3]. The effect of this artifact is stronger than

that of the eye blink artifact. Eye blinks and movements often occur at close intervals.

ii) Line Noise: Strong signals from A/C power supplies can corrupt EEG data during transfer from the scalp electrodes to the recording device. Notch filters are often used to filter this artifact containing lower frequency line noise and harmonics. Notch filtering at these frequencies can remove useful information. Line noise can corrupt the data from some or all of the electrodes depending on the source of the problem.

iv) Muscle Activity: These artifacts are caused by activity in different muscle groups including neck and facial muscles. These signals have a wide frequency range and can be distributed across different sets of electrodes depending on the location of the source muscles.

v) Pulse. When an electrode is placed on or near a blood vessel, it causes pulse, or heart beat, artifact. The expansion and contraction of the vessel introduce voltage changes into the recordings

2. DESCRIPTION OF PSBP

In TVMBP, the two-dimensional object function from which parallel projections are being measured is $f(x,y)$. Crawford assumed that during projection acquisition, respiratory motion transformed the object function into $f(x',y')$ where $x'=\alpha_x+\beta_x x$ and $y'=\alpha_y+\beta_y y$. In these expressions, the α 's represent translations and the β 's represent magnification factors. Crawford showed that $f(x,y)$ could be reconstructed by

$$f(x, y) = \int_0^\pi q_\theta \left(\left[\frac{(x-\alpha_x)}{\beta_x} \right] \cos \theta + \left[\frac{(y-\alpha_y)}{\beta_y} \right] \sin \theta \right) d\theta, \quad (1)$$

, where θ is the projection angle, and $q_\theta()$ is a projection convolved with a ramp filter. Note that in (1) the terms in square brackets are the

inverses of the mapping of (x',y') into (x,y) . Therefore, (1) demonstrates that $f(x,y)$ can be recovered by back-projecting into the coordinates where the pixel was located at the time the projection was acquired. To derive (1), the time-varying motion model was assumed to be applied to the entire image plane. In PSBP we applied the time-varying motion model to individual pixels in the image plane. We assumed that the correction of one pixel would not generate artifacts in neighboring pixels that would affect the pixel being corrected. Instead of defining a specific motion model, we defined a general, spatially- and temporally-varying model of the form $x'=G(x,y,\theta)$ and $y'=H(x,y,\theta)$, where G and H were called warping functions and θ was a function of time.

This block diagram is a further extension of the TVMBP algorithm of Crawford. The reading and writing images are located at the beginning and end of the data processing pipeline. These classes are known as data sources (readers) and data sinks (writers). Generally speaking they are referred to as filters, although readers have no pipeline input and writers have no pipeline output.

CTX Algorithm reduces motion artifacts by performing the back-projection in a frame of reference that moves with the object. The motion during scanning is modeled as a shift and as a magnification about some origin point $(X0, Y0)$. CTX uses projection data during back-projection that corresponds to the location at which each point resided at the time each projection was measured.

In CTX, let $f(x, y)$ be the cross section that is to be reconstructed. A magnified and shifted version of $f(x, y)$, $f'(x, y)$ is given by

$$f'(x, y) = f(\alpha_x + \beta_x x, \alpha_y + \beta_y y)$$

Where α_x and α_y , are shift factors, and β_x and β_y are magnification factors. Both the α 's and β 's are assumed to be functions of projection angle θ , which is in turn a function of time. The direction of the z axis is assumed to be the superior-inferior direction, and the z axis is perpendicular to the xy -image plane.

A parallel projection $P'(\theta, t)$ of $f'(x, y)$ is given by the radon transform

$$P'(\theta, t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f'(x, y) \delta(t - x \cos \theta - y \sin \theta)$$

By substituting the inverses of G and H into (1),

$$f(x, y) = \int_0^\pi q_\theta [G^{-1}(x, y, \theta)] \cos \theta + [H^{-1}(x, y, \theta)] \sin \theta d\theta, \quad (2)$$

$f(x,y)$ can be recovered from projections measured from $f(x',y')$.

3. METHODS AND MATERIALS

Here is a small block diagram which will explain about the motion artifacts correction.

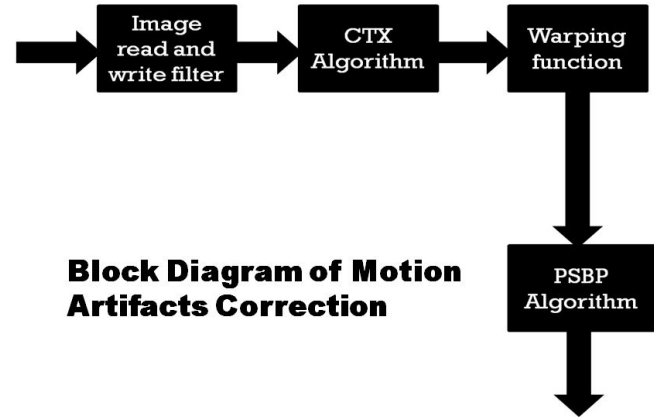


Image warping is the process of digitally manipulate image such that any shapes portrayed in the image have been significantly distorted. Warping may be used for correcting image distortion as well as for creative purposes. While an image can be transformed in various ways, pure warping means that points are mapped to points without changing the colors. This can be based mathematically on any function from (part of) the plane to the plane. If the function is injective the original can be reconstructed. If the function is a bijection any image can be inversely transformed. **PSBP Algorithm**, each pixel is reconstructed in a frame of reference local to that pixel. To develop such an algorithm, we made the assumption that local Correction was valid in CT reconstruction. Although the CTX algorithm is mathematically correct, it is based on a model that does not describe motion in the chest. Furthermore, an exact back-projection algorithm could not be derived that accounted for this motion. Therefore, we made the assumption that the CTX model was valid only in a small region around each pixel, and that α and β parameters of the CTX model for each of these regions need not be identical. We then developed an algorithm in which each pixel was reconstructed in a frame of reference local to that pixel. To develop such an algorithm, we made the assumption that local Correction was valid in CT reconstruction.

To perform local correction, the motion in a body is first described by a temporally and spatially varying function (called a warping function). These warping functions are, in General, a function of space and projection angle.

$$x' = G(x, y, \theta)$$

$$y' = H(x, y, \theta)$$

where (x', y') are the warped Cartesian coordinates, (x, y) are the nonwarped coordinates, θ is the projection angle, and G and H are the warping functions.

4. RESULTS AND DISCUSSION

Experiments for motion artifact reduction, a major problem of wearable images, was accomplished with real-time algorithm. The paper summarizes the overview of Artifacts and their removal in medical image. Various techniques has been discussed for artifact removal. Artifacts are removed by using the Back-projection method of each pixels of one slice of image from a list of frame, and depending upon the intensity of each pixel in an image. Reconstruction of an image is done. Rescale Intensity Image filter is also used for pixel shifting.

Fig-1 shows a CT image of the chest with motion artifacts and Fig-2 shows an image after reconstruction using PSBP algorithm.

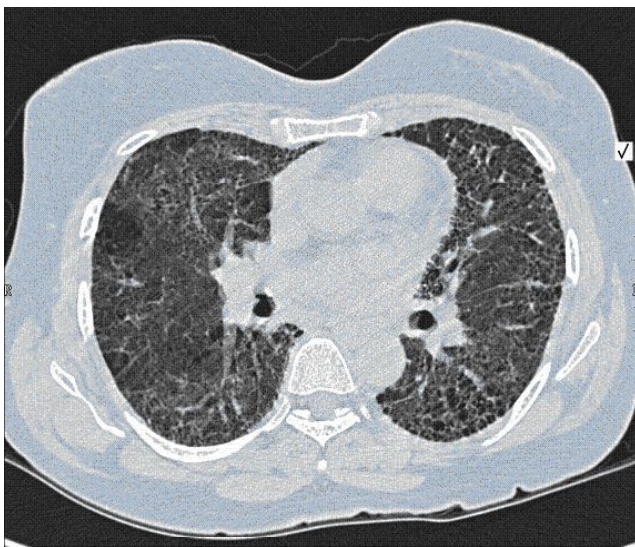


Fig 1- Image before reconstruction with artifacts

5. ACKNOWLEDGMENT

I thank Dr K Soundararajan, Dr V Ushasree and S Ismail saheb for their comments and suggestions on this manuscripts.

6. REFERENCES

[1] Rohtash Dhiman, J.S. Saini, Priyanka, A.P Mittal*
Deenbandhu Chhoturam University of Science & Technology,
Murthal ARTIFACT REMOVAL FROM EEG RECORDINGS – AN
OVERVIEW ,NCCI 2010 Computational Instrumentation CSIO
Chandigarh, INDIA, 19-20 March 2010

[2] L.R. Kuhns and G. Borlaza. "The twinkling star sign: An aid in
Differentiating pulmonary vessels from pulmonary nodules on
computed Tomograms," *Radiol.*, vol. 135, pp. 763-764, 1980.

[3] R. J. Alfidi, W. J. MacIntyre, and J. R. Haaga, "The effects of
biological motion on CT resolution," *Amer. J. Radiol.*, vol. 127, pp.
11-15, 1976.



Fig 2- Image after reconstruction using PSBP algorithm

[4] J. D. Godwin, R. S. Breiman, and J. M. Speckman, "Problems and
pitfalls in the evaluation of thoracic dissection by computed
tomography," *JCAT*, vol. 6. pp. 750-756, 1982.

[5] C. R. Crawford, K. F. King, J. D. Godwin, C. J. Ritchie, and Y.
Kim, "Respiratory compensation in projection imaging,"
Medical Physics, in press,

[6] R. L. Ehman, and J. P. Felmlee, "Adaptive technique for
highdefinition MR imaging of moving strictures," *Ra&ology*, qol.
173, pp. 255-263, 1989.

[7] N. J. Pelc and G. H. Glover, "Method for reducing image artifacts
due to projection measurement inconsistencies," U.S. Patent
4,580,219, 1986.

[8] D. L. Parker, V. Smith, and J. H. Stanley, "Dose minimization in
computed tomography overscanning," *Med. Phys.*, vol. 8, pp. 706-71
1, 1981.

[9] Digital Image Processing – Gonzalez and Woods- 3d edition.

[10] ITK.org [10] E. M. Haacke and J. L. Patrick, "Reducing motion
artifacts in twodimensional Fourier transform imaging," *Magnet.
Reson. Imag.*, vol. 4, pp. 359-376. 1986

[11] T.P. Jung, S. Makeig, M. Westerfield, J. Townsend, E.
Courchesne, and T.J. Sejnowski. Removal of eye activity artifacts
from visual eventrelated Potentials innormal and clinical subjects.
clinicalNeurophysiology, 111(10):1745-58, 2000.



Implementation of Pipelined Data Encryption Standard for Security Enhancement through Verilog

U.Ratna Kumari
Department of ECE,
GIT, GITAM University,
Vishakhapatnam, Andhra Pradesh, India

T.K.Rasagna
Department of ECE,
GIT, GITAM University,
Vishakhapatnam, Andhra Pradesh, India

Abstract: This paper specifies a cryptographic algorithm in order to protect sensitive data. To maintain the data confidentially and to protect it, we need to convert the data into different form that differs completely from input and then transmit it. That data has to be again decrypted at the receiver. The algorithm defines the steps needed to encrypt the data and also to decrypt it. The pipelined DES has three modules: DES, pipeline, Control module. This design is programmed in Verilog. By pipelining we can achieve high throughput and by implementing triple DES, the security can be increased.

Key words: DES, Key scheduling, F-function, pipelined DES, triple DES.

1. INTRODUCTION

Most of the information in today's world is in digital format. For example most of the information in the form of photos, music and private information can be transmitted through copper, optical or wireless network to a recipient anywhere in the world. One of the advantages of Internet is the open system architecture. Its flexibility makes Internet developed fast. On the other hand, the lack of privacy in the Internet becomes obstacle to growing further. However, this weakness can be eliminated with the introduction of cryptography. Cryptography is used to transform intelligible information to unintelligible data. The Data Encryption Standard (DES), known as the Data Encryption Algorithm (DEA) by ANSI and the DEA-1 by the ISO, has been a worldwide standard for over 20 years. DES is a block cipher, which takes 64-bit input and 64-bit key. A 64-bit output is produced. The effective key length is 56 bits because 8 bits are used as parity-checking bits. There are a total of 2^{56} possible keys available in 56-bit key length. DES is a symmetric algorithm. The same key is used for both encryption and decryption. DES has 16 rounds, meaning the main algorithm is repeated 16 times to produce the cipher text.

2. ALGORITHM

In each basic building block of DES, the input will be split into two, left half and right half. The right half will become left half for the next round. Meanwhile, the right half will go through the function f to produce a key-dependent output and then XOR with left half. The result will become right half for the next round. The basic building block of DES is repeated for 16 times [1]. The only difference between each round of building block is

the key used as shown in figure 1. Every eight bit of the 64-bits key is used for parity checking and otherwise ignored. After an initial permutation, the 64-bits input is split into a right and left half each 32 bits in length. DES has 16 iterations or rounds. In each round a function f is performed in which the data is combined with a 48-bits permutation of the key. After the 16th iteration, the right and left halves are concatenated and a final permutation, which is the inverse of the initial permutation, completes the algorithm [3].

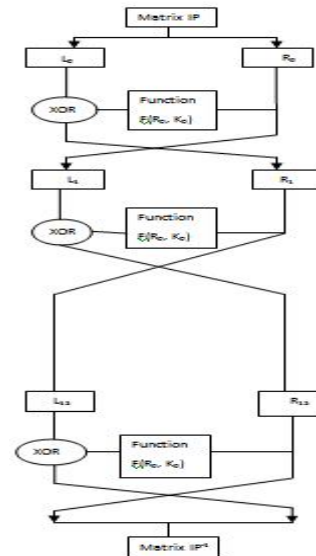


Figure 1. DES algorithm

3. F-FUNCTION

The function f of the DES algorithm is made up of four operations. Firstly, the 32-bits right half of the plaintext is expanded to 48-bits and then X-ORed with a 48-bits sub-key K_1 . The result is fed into eight substitution boxes (s-boxes), which transform the 48-bits input to a 32-bits output [4]. Finally, a straight permutation (P-permutation) is performed, the output of which is XORed with the initial left half L , to obtain the new right half R_1 . The original right half R_0 becomes the new left half L_1 as shown in Fig 3.

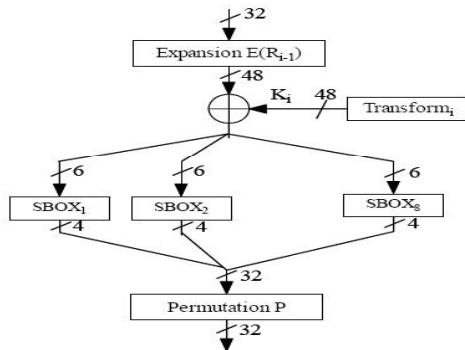


Figure 2. f-function

4. KEY SCHEDULING

Although the input key for DES is 64 bits long, the actual key used by DES is only 56 bits in length. The least significant (right-most) bit in each byte is a parity bit, and should be set so that there are always an odd number of 1s in every byte. These parity bits are ignored, so only the seven most significant bits of each byte are used, resulting in a key length of 56 bits. The initial step in the in this procedure is to remove the parity check bits in the 64-bit key. Every eighth bit is used for parity checking, leaving 56-bits. A different 48-bits sub key is now generated for each of the 16 rounds of DES [7][8]. The sub-keys are determined by first splitting the 56-bits into two 28- bits lengths of data. Then both halves are shifted left by either one or two bits depending on the round number. The procedure for generating the sub keys - known as key scheduling is fairly simple:

1. Set the round number R to 1.
2. Split the current 56-bit key, K , up into two 28-bit blocks, L (the left-hand half) and R (the right-hand half).
3. Rotate L left by the number of bits specified in the table below, and rotate R left by the same number of bits.
4. Join L and R together to get the new K .
5. Apply Permuted Choice 2 (PC-2) to K to get the final $K[R]$, where R is the round number we are on.
6. Increment R by 1 and repeat the procedure until we have all 16 sub-keys $K[1] - K[16]$.

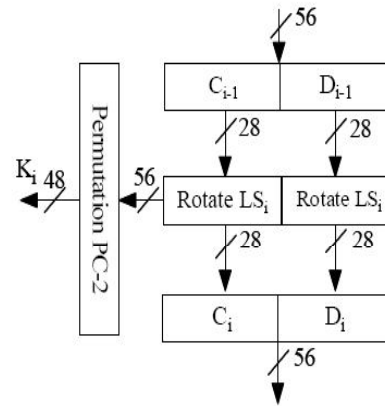


Figure 3. Key Scheduling

5. PIPELINED DES

The figure4 shows the pipelined DES architecture. There are four registers, each 16-bit, to store 64-bit input. Pipelined DES will load 4 different inputs from ISA bus for the first four rounds. This will fill the pipeline with 4 different inputs. After the first four rounds, the result from the fourth segment will be fed back to the first segment and so on. 4 different inputs will be encrypted in the architecture. At the 16th round, the first data is encrypted, and 17th the second data is obtained and so on [6].

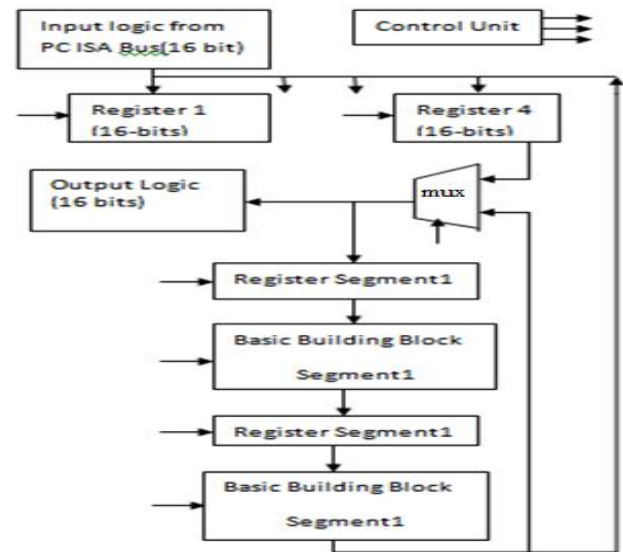


Figure 4. Architecture of Pipelined DES

Control unit is in charge of coordinating operations of components and data flow. The flow chart is shown in Figure 5.

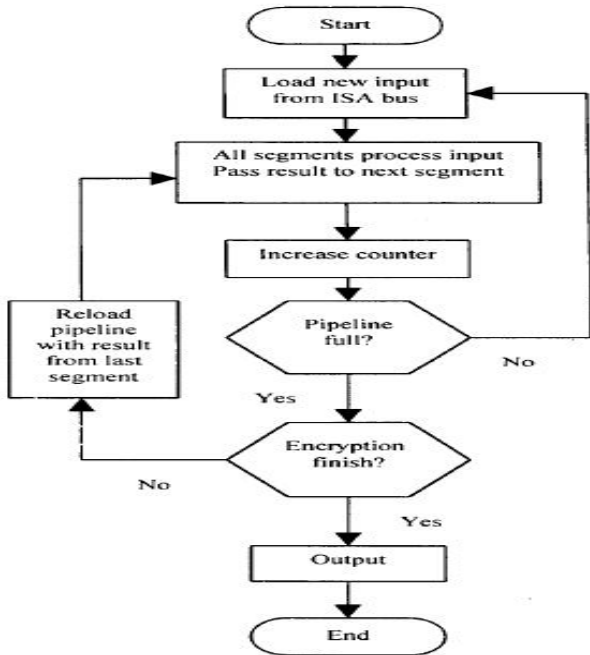


Figure 5. Flow Chart of Pipelined DES.

6. TRIPLE DES

The following figure 6 shows the triple DES where three keys are used to encrypt the data. The same DES is repeated thrice by using three different keys K1, K2, K3. The plain text is encrypted using key K1 and it is again decrypted using key K2 and again encrypted using K3.

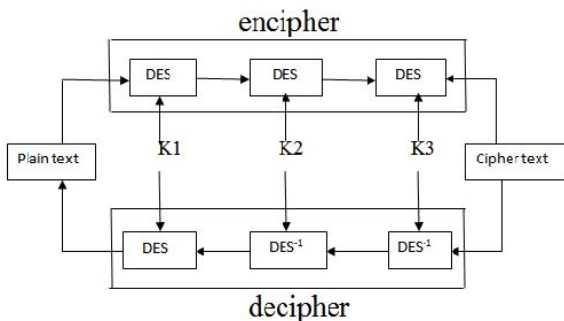


Figure 6. Block Diagram of Triple DES

7. APPLICATIONS

Cryptographic services are required across variety of platforms in a wide range of applications such as secure access to private networks, electronic commerce and health care. The security of conventional encryptions depends on several factors. DES can be used in intensive cryptographic computer application. Applications such as electronic commerce, internet banking and

electronic fund transfer, secure and private communication require better performance cryptographic system.

8. RESULTS

Implementation of DES algorithm was accomplished using Xilinx 8.1 as simulation tool. The design was coded in Verilog. The design achieves a frequency of 111.882 MHz It takes 16 clock cycles latency first time only then encrypts one data block (64-bits) per clock cycle. Initially let the key be $K = 00010011\ 00110100\ 01010111\ 01111001\ 10011011\ 10111100\ 11011111\ 11110001$. From this key, the sub-keys are generated. The sub-keys generated are shown in fig 7:

KEY1[48:1]	42b	42b0001101100000101101111110001100001110010
KEY2[48:1]	42b	42b011100110101101110110011011011110010011100101
KEY3[48:1]	42b	42b0101010111111010001010000010110011110011001
KEY4[48:1]	42b	42b011001
KEY5[48:1]	42b	42b0111101110110000001111101010101000110101000
KEY6[48:1]	42b	42b01100011101001010011110010100001110110010111
KEY7[48:1]	42b	42b1101010000101010110111110100011000110011100
KEY8[48:1]	42b	42b11101110010100011010110101000010011011111001
KEY9[48:1]	42b	42b110000110101011010111010111010101100111000001
KEY10[48:1]	42b	42b101000111100101000111011010100010010010111
KEY11[48:1]	42b	42b01000010101111101010110111010011100110
KEY12[48:1]	42b	42b0110101011000101000111101010110100110100100001
KEY13[48:1]	42b	42b10101111000101101001111010101101001000100001
KEY14[48:1]	42b	42b010111101000110101111010111001100110011010
KEY15[48:1]	42b	42b101111100100011000110100111010011110001010
KEY16[48:1]	42b	42b1101011011101100010100011100010111110101

Figure 7. Output of Key-Generation

After the generation of sub-keys, the initial text (here $M=1221210128FEDCBA$) is divided into two halves and then the right half is applied to the s-box. The outputs are xored with function and then the right and left halves are swapped and finally applied to the inverse permutation. The final output i.e. the cipher text is shown in fig 8.

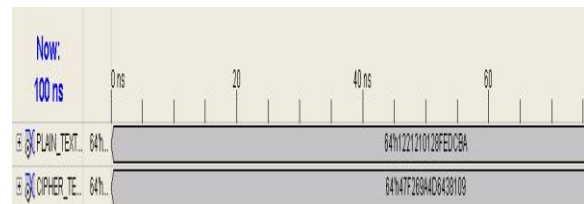


Figure 8. ENCRYPTED DATA

So the initial data is encrypted. Now the data encrypted is again decrypted at the receiver side to obtain the original data being transmitted. So in order to obtain the original data, we should again repeat the same algorithm with the key16 for the round1, key15 for the round2 and so on. The cipher text for the different inputs is shown in the figure9.

Plain Text	Cipher Text
64'h12212101 28FEDCBA	64'h47F269A4 D6438109
64'h01234567 89ABCDEF	64'h85E81354 0F0AB405.
64'h97530281 9A7D4B7C	64'hF8C675DA 6904AB21

Figure 9 Various Plain and cipher text obtained by DES

The pipelined data with 4 segments is implemented. The output of sbx1 is shown in the figure10.

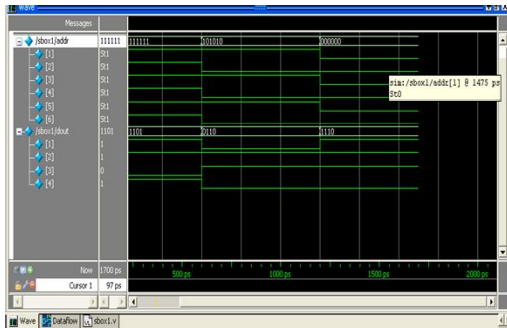


Figure 10. Pipelined output of Sbox1

First input is first fed into segment 1. After being processed by basic building block of segment 1, the output is fed into register of segment 2. At the same time, the second input is fed into segment 1. There are two inputs now, first input in segment 2, second input in segment 1, in pipeline now. The final pipelined output is shown in Fig 11.

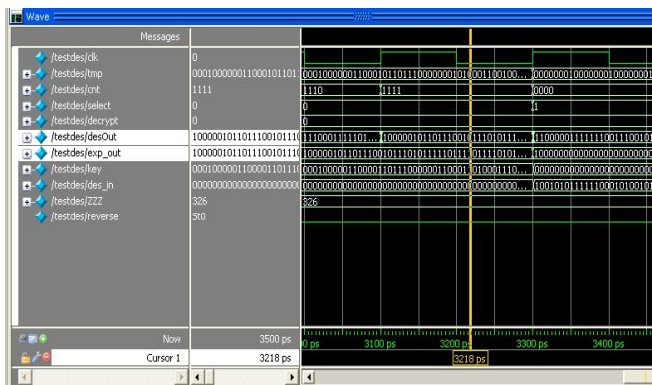


Figure 11. Output of Pipelined Des

During the triple DES, the text is encrypted using K1 and then decrypted using K2 and again encrypted using K3 as shown in figure 12.

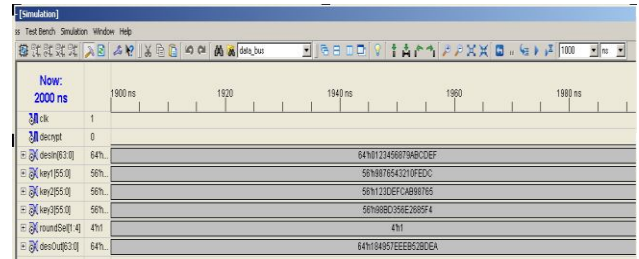


Figure 12. Output of Triple DES

For triple DES, at step 1, the input to DEA1 is “The quic” i.e., P1= 5468652071756663, and the output of DEA1 is “5D0946FFEB52ECFB”. At step 2, the input to DEA2 is the output of DEA1, and the output of DEA2 is “4454515014415400”. At step 3, the input to DEA3 is the output of DEA2, and the output of DEA3 is “484D02BFAA52A9BA”. The output of DEA3 is the cipher text C1. The tabular representation of the outputs at various stages of encryption is shown in the figure 13.

	Input	Output
DEA1 - FK ₁	5468652071756663	5D0946FFEB52ECFB
DEA2 - IK ₂	5D0946FFEB52ECFB	4454515014415400
DEA3 - FK ₃	4454515014415400	484D02BFAA52A9BA

Figure 13. Tabular output of Triple DES

The comparison of the straight forward architecture and triple DES is shown in the figure 14. The time taken for encryption is less in pipelined DES.

Architecture	Straight DES	Triple DES
Key Length	56 bits	158 bits
Possible Keys	2 ⁵⁶	2 ¹⁵⁸
Flip Flops Used	64	192
LUT used	878	2636
Max. Freq	525.216MHz	446.730MHz
Min. Time Period	1.768ns	2.254ns

Figure 14. Comparison of DES and triple DES

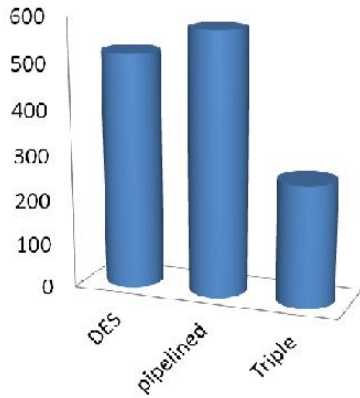


Figure 15. Maximum Freq comparison

The above graph in figure 15 portrays the comparison between the maximum operating frequency and throughput of the existing work and the results obtained. As known pipelining increases the operating frequency as well as the area but the triple DES increases the security of the system.

9. CONCLUSION

Developed a Straight forward DES architecture using the 64-bit key. The sub- key is generated and then the data which is to be sent is encrypted and converted into cipher text. Later, the data is encrypted using a 4 segment pipelined DES and the encrypted time for both is observed & the triple DES is implemented using three independent keys.

The pipelined DES consumes less hardware resource than fully pipelined DES does, and provides more throughput than practical DES. The Triple DES consumes more hardware and also the frequency of operation is low compared to straight DES. But the basic advantage of using triple DES is that it is more secured than that of DES and pipelined DES.

10. REFERENCES

- [1] Teo Pock Cheung, "Implementation of Pipelined Data Encryption Standard (DES) Using Altera CPLD" in Proc. IEEE Trans circuits syst vol.74, no.13, 1759-1763, 2007.
- [2] The DES Algorithm Illustrated by J. Orlin Grabbe.
- [3] Schneier, B. "Applied Cryptography, Protocols, Algorithms, and Source Code" in Proc. IEEE Int Symp circuits syst (ISCAS), 2005 pp.592-595,2003.
- [4] Ahmed Zure Sha'meri, "DES Cryptographic System for Information Security" in Proc. IEEE Trans circuits syst vol.53, no.11,1165-1169, 2002.

- [5] Wong, K., Wark, M., Dawson, E.: A Single-Chip FPGA Implementation of the Data Encryption Standard (des) Algorithm. In: IEEE Globecom Communication Conf., Sydney, Australia (2002) 827–832.
- [6] FPGA implementation of des using pipelining concept with skew core key-scheduling By vishwanath patel, r. c. joshi, a. k. saxena.
- [7] J Wilcox, D., Pierson, L., Robertson, P., Witzke, E.L., Gass, K.: A DES basic suitable for network encryption at 10 Gbs and beyond. In: CHES 99, LNCS 1717 (1999) 37–48.
- [8] Kaps J, Fast DES implementations for FPGAs and its application to a Universal key-search machine. In: Proc. 5th Annual Workshop on selected areas in cryptography.



Cultural, Technological and Informational Based Knowledge Management

Varick D. Love
Strayer University
Henrico Campus, USA

Ify S. Diala
Contributing Faculty
Walden University
Minneapolis MN, USA

Abstract: The goal of this research was to compare and contrast each protocols role in knowledge management. These protocols includes: Cultural, Informational and Knowledge management. Cultural information lays out the norms and beliefs of the corporate environment, informational knowledge management looks at how individuals processes information's and solves problems on a daily bases while technology is a tool that presents information to the users. Cultural, informational and technology based knowledge management are so close but so far away in how they use information.

Keywords: knowledge management; cultural management; informational management; technology management; tacit; explicit

1. INTRODUCTION

Knowledge management is an organizational management concept of resource allocation that includes employees. Corporations that focus on knowledge management spend a great deal of time and resources identifying the knowledge they are trying to harness, nurture and preserve for future use. The correlation or comparison of the knowledge management is based on the unique categories of Tacit and Explicit data.

2. PURPOSE OF THE STUDY

This research will analyze knowledge management from perspectives of cultural based knowledge management, technological based knowledge management and information-based knowledge management. The study will look at how the knowledge-based protocols are compared and contrasted with each other, and the role of computer technology in capturing, analyzing and using the knowledge protocols.

3. DISCUSSION

3.1 Business Outlook

Business organizations understand the need to invest in resources that improve the intellectual knowledge of employees. Corporations understand that by developing and investing in the knowledge based resources, they will go a long way towards ensuring the growth and future success of the organization. The competitive business environment will result in strategic planning tools for profit enhancement and product or service differentiation to meet the needs of the consumer [1].

3.2 Cultural, Informational and Technological

When comparing culture, technical and informational knowledge management, it is important to decide what type of information that the researcher is looking for. Has the information already been created or must it be created, is it Tacit or Explicit? Wallach [2] stated that Tacit culture consist

of three groups bureaucratic, innovative and supportive. The correlation or comparison of the knowledge management is based on the unique categories of Tacit and Explicit data that this research will compare. Knowledge flow information is broken down into two categories (a) Tacit and (b) Explicit. Tacit knowledge or information is that information or knowledge that is known only to the individual, like cultural tacit knowledge, informational tacit knowledge cannot be touched or heard. Each individual person has their own unique set of habits and norms that make them different from everyone else. How an individual solves a problem using their intuition or habit is that individual's Tacit knowledge, Tacit knowledge is not written down, nor can it verbalized [3]. As teams are formed each member will have their own unique tacit habits, but for the teams to be successful each team member will have to find a method of articulating their expertise, by turning their tacit information into explicit information.

3.2.1 Tacit

Bureaucratic Tacit work cultured environments are regimented; rules will be followed to the letter with little deviation from the assigned directions for completing the task, any deviations from these guidelines could result in repercussion for the employee. Tacit Innovative work culture is the exact opposite of bureaucratic culture; innovative culture encourages creative thinkers to express themselves by coming up with innovative ideas to improve processes and procedures, that could help improve the corporations standing in their respective industry. Because innovators are encouraged to think outside the box, they sometimes take risks, to try out a new innovative idea or inventions, these types of risks would not be encouraged in a bureaucratic culture. Wallach [2] defined a Tacit supportive culture as one where ideas will be encouraged and shared amongst not only to direct team members but to other departments that might see benefit in the ideas being presented [2].

3.2.2 *Explicit*

Explicit information is that information, which can be shared, whereas tacit information is personal to the individual and cannot be touched; explicit information or knowledge is information that can be, heard, read and touched. Explicit knowledge is that knowledge that was once Tacit but has now been transferred to a shareable medium; this shareable medium, could be an instructional manual, a training CD etc. Explicit information gives each team member the opportunity to learn new methods or techniques to resolve an issue or simply familiarize themselves with a new process or technique [8].

3.3 Cultural Knowledge Management

The organizational cultural beliefs instilled in corporate employees ensure that everyone in the corporation has same understanding of the goals the organization is trying to achieve. These organizational beliefs determine the observable organizational norms and practices that consist of rules, expectation, rituals and routines, power structures, and control systems [4]. From a knowledge management perspective Tacit culture decides the norms or beliefs that will determine who will control knowledge as well as who will share the knowledge. Only when Leadership and employees start to share or gain an understanding of each other's cultural ideologies, visions and beliefs will true informational and technological change begin to take place. If there is no understanding, distrust and acrimony could set in delaying the project [5]. Park [6] believed there was a direct correlation between the success and failure of informational and technology based knowledge management integrations and the tacit cultural attributes exhibited by employees of a corporation. Tacit cultural questions or concerns must be addressed before information and technological concerns could be addressed. The proper corporate culture will enable and motivate employees to produce, share and use information that will be of benefit not only to the corporation but also the employee's as well. Schein [7] described culture as a set of rules or beliefs held by members of an organization or group. The cultural rule determines how the group adapts or reacts to it, not only to their environment but also in how they react to stakeholders within its environment. Schein [7] believes that tacit culture exists in correlations with three conceptual levels, basic assumption, values and artifacts. Basic assumption or beliefs are interpretive ideas which individuals give sense to activities and human relations that is the basis of cooperative action. Beliefs or basic assumptions develop through time as groups or corporations deal with the consequences of actions taken, resolve a problem or issue. Values can be described as the rules that define norms so that communication among team members or others in organization can occur. Depending upon the organization or group, values will define what it deems socially acceptable or socially unacceptable behavior, when dealing with a co-worker or dealing with a customer. Although Tacit by category, Artifacts can be explicitly seen, heard and smelled; they are not localized to one individual but it is available to the masses or to those authorized to see or hear the information. Artifacts can be cultural dances, or cultural languages, or it could be a video recorder with specific instructions how to reproduce Einstein theory of relativity [7]. From a Tacit cultural perspective, Artifacts are the first step in gaining a true understanding of the importance of information and the impact it has on Knowledge management. As mentioned previously, before leadership can begin integrating a knowledge management system into the corporate environment they must get a true understanding of the culture.

After the cultural hurdles have been overcome, leadership must then begin the process of determining how to go about increasing the knowledge flow by turning the tacit data into Explicit data, so that everyone can have access to the necessary information.

3.4 Information Knowledge Management

Explicit information is that information, which can be shared, whereas Tacit information is personal to the individual and cannot be touched; Explicit information or knowledge is information that can be, heard, read and touched. Explicit knowledge is that knowledge that was once Tacit but has now been transferred to a shareable medium; this shareable medium, could be an instructional manual, a training CD etc. Explicit information gives each team member the opportunity to learn new methods or techniques to resolve an issue or simply familiarize themselves with a new process or technique [8]. The learning flow categories look at how team members adapt to each other and how they adapt to their surroundings. Just as Tacit information is unique to each individual, how each person adapts to new surrounding and each other will be unique as well. Conventional needs to be questioned, if it is determined that the Explicit information for a process or procedures is no longer relevant. It is imperative that the team members do everything necessary to get more updated Explicit information, If is determined that the updated information is still in a Tacit state, then leadership and the team must, devise a new plan of action. If individuals on the team insist on using the out dated information, the project or task will assuredly move in direction that will result in eventual failure. Lastly, Behavioral changes focuses on how the individual's team members react when being given Tacit or Explicit information in the form of feedback or direction, by superiors or other teammates. Do the teammates get defensive or are they receptive when the information is negative, or do they take it as information that is used to improve. Behavioral scientist theorizes that learning is a change in an individual or corporations behavior; brought on by past or present experiences and how those experiences affected the decision making process of the organization or individual [9]. Individuals have to ask themselves did the internal or external feedback from those experience assist them in making better decision going forward or did it have such a negative impact that the individual refuse to address the feedback given and keep using old information that caused the experience to occur as a barometer in their decision making process?

3.5 Technology Knowledge Management

Place Nonaka [10] stated that for Tacit information to be turned into explicit technological information, individuals or groups would need to show a common interest in resolving a unique problem. After the common interest groups are formed, the team members will have meetings to discuss their common interest in resolving the issue, and to get updates on the status up to this point on how close they believe they are in coming to a resolution. The common interest groups are not necessarily localized to a common college or town; the members could be in different states, or different countries. Four team members could be in Australia, while two could be in Japan and six could be in Nigeria, but they all came together to resolve a common issue. Because the geographical hindrance in face to face meetings would not be a feasible means of collaboration. The team members will need an agreed upon technological tool to conduct their meeting, such as email, chat rooms, video conferences such as Skype or WebEx sessions where they view each other's computer

screen [11]. The goal of Technology based explicit knowledge or information is to make tacit information accessible. Explicit knowledge or information is presented to users through artifacts, such as a computer, CD, network storage appliance or chat room. The Explicit data could be used by teammates working on a project, or a customer looking to get information on a new product, that is due to be released. Explicit information is meeting through manuals, electronic bulletin boards, and text through phones etc. or in some cases using Customer Resource management technology applications.

4. SHARING OF INFORMATION

Without the sharing of information, projects or job functions will suffer. Information must be shared between teammates or coworkers relevant to their designated role in the organization or to the project; the team members are currently assigned to. Lytras & Pouloudi, [9] proposed three information dynamic flows to improve information flow between groups or individuals: Knowledge flow, Learning Flow and Behavioral change. Knowledge flow is based upon the team building concept e.g. everyone coming together for a common cause or goal. For projects or other tasks to be successful, it is imperative that everyone on the team has the correct information. If it is found that team members are not sharing information or giving required feedback, or it is determined that the data is out of date, then knowledge or information flow will become stagnant and project will suffer. Knowledge flow information is broken down into two categories (a) Tacit and (b) Explicit. Tacit knowledge or information is that information or knowledge that is known only to the individual, like cultural tacit knowledge, informational tacit knowledge cannot be or heard. Each individual person has their own unique set of habits and norms that makes them different from everyone else. How an individual solves a problem using their intuition or habits which is that individual's tacit knowledge, tacit knowledge is not written down, nor can it be verbalized [3]. As team are formed each member will have their own unique tacit habits, but for the teams to be successful each team member will have to find a method of articulating their expertise, by turning their Tacit information into Explicit information. Tacit Information is derived from a person's head to resolve a particular issue for a customer or coworker. Once that information is documented, it is then shared with others on the team, once it is shared with the team it can be used as a tool or a resource.

5. CONCLUSION

Cultural, informational and technology based knowledge management are so close but so far away in how they use information. The first thing that stood out to this researcher is how each relied upon different aspects of information to be successful. The knowledge protocols compared to each other because they each need information to be successful. From the leadership perspective, the primary difference between information, technology and culture knowledge, is that information and technology management both have physical attribute that are easily accessible to a user; versus cultural knowledge that would be abstract to the naked eye [12]. Technology by definition is an artifact that presents information to a user [7]. Information by nature starts out in a tacit state; it does not become usable until it is presented in an explicit state by a technology artifact. The Tacit information could be a theory in a user's head on how to splice DNA, but until someone documents the processes and procedures and puts them on a shared media, it is unusable [11]. Unlike Technology or informational-based knowledge management,

an artifact does not represent cultural knowledge. Cultural knowledge management is based upon beliefs, values exhibited groups, or organizations, a culture that is conducive to change will ensure that knowledge management integration will be a success. However, cultural knowledge management has no impact on the actual information or technology integrated into an organization. Cultural Tacit information could be based on many variables, such as the country the integration will take place, the religious beliefs of people working in the organization, the age of the employee's in the office. However, those variables have no impact on tacit informational knowledge management or Technology knowledge management being deployed within an organization. Just as Tacit, information from a cultural perspective is based upon beliefs or norms and is purely abstract. Information and Technology knowledge management are based upon known truths, experience or ideas [4].

6. REFERENCES

- [1] Shohan, S., & Perry, M. (2009). Knowledge management as a mechanism for technological and organizational change management in Israeli universities. *Higher Education*, 57(1), 227-246.
- [2] Wallach, E. J. (1983). Individuals and corporations: the cultural match. *Training and Development Journal*.
- [3] Mohamed, M., Stankosky, M., & Murray, A. (2006). Knowledge Management and information technology: Can they work in perfect harmony. *Journal of Knowledge Management*, 10(3), 103-116.
- [4] Bloor, G., & Dawson, P. (1994). Understanding professional culture in organizational context. *Organizational studies*, 15(2), 275-295.
- [5] Tierney, W. (1992). Cultural leadership and the search for community. *Liberal Education*, 78(5), 2-21.
- [6] Park, H., Ribiere, V., & Schulte Jr, W. D. (2004). Critical attributes of organizational culture that promote knowledge management technology implementation success. *Journal of Knowledge Management*, 8(3), 106-117.
- [7] Schein, E. H. (1985). *Organizational culture and leadership*. San Francisco, CA: Jossey-Bass.
- [8] Hannabuss, S. (2000). Narrative knowledge: eliciting organisational knowledge from storytelling. *Aslib Proceedings*, 52(10), 402-417.
- [9] Lytras, M. D., & Pouloudi, A. (2006). Towards the development of a novel taxonomy of knowledge management systems from a learning perspective: an integrated approach to learning and knowledge infrastructures. *Journal of Knowledge Management*, 10(6), 64-80.
- [10] Nonaka, I., & Konno, N. (1998). The concept of Ba: building a foundation for knowledge creation. *California Management Review*, 40(3), 1-15.
- [11] Marwick, A. D. (2001). Knowledge management technology. *IBM Systems Journal*, 40(4), 814-830.
- [12] King, W. R. (2008). Questioning the conventional wisdom: culture-knowledge management relationship. *Journal of Knowledge Management*, 12(3), 35-47.

- [12] King, W. R. (2008). Questioning the conventional wisdomL: culture-knowledge management relationship. *Journal of Knowledge Management*, 12(3), 35-47.



Design Analysis of Autonomous Air Traffic Flight Control System

D. Vasumathi

Department of Computer
Science and Engineering,
JNTU Hyderabad,
India.

P. Rajarajeswari

Department of Computer
Science and Engineering,
Madanapalle Institute of
Technology and Science,
Madanapalle, India

A. Ramamohan Reddy

Department of Computer
Science and Engineering,
S.V.University, Tirupathi, India

Abstract: Software architectural design, also known as top-level design, describes the software top-level structure and organization and identifies the various components. The concept of an automated air traffic flight control system which controls airplanes requires a high degree of operational integrity and availability. One possible solution to alleviate air travel congestion could be the automation of air traffic control and allowing it to have direct control over airplane flight paths. Such a system would, in theory, reduce the workload of the flight crew and the air traffic controllers, as well as increase traffic flow. This paper presents several analyses of such a conceptual system from a “net-centric” perspective. First, the system’s operation is described from the context of a flight, to provide a basis for the discussion of various system models and views. Spiral development model stages as well as related events which occur during system design give an idea of how the system would be developed incrementally. Formal methods can be used to improve software security but can be costly and also have limitations of scale, training, and applicability. To compensate for the limitations of scale, formal methods can be applied to selected parts or properties of a software project, in contrast to applying them to the entire system. The concept of object-oriented development (OOD) has gradually matured from being presented. The OOD can still be regarded as one of the mainstream development models. UML includes a standardized graphical notation used to create an abstract model of a system, referred to as a UML model. We describe AATFCS system with UML modeling techniques. AADL is an extensible and allows us to introduce new properties; we can define a set of properties specific to the data state variable. In this paper we present the AADL language for AATFCS system.

Keywords: Software Architecture, Autonomous Air traffic Flight control system, Spiral development, UML modeling analysis, Architecture analysis design language.

1. INTRODUCTION

The architectural design allocates requirements to components identified in the design phase. Architecture describes components at an abstract level, leaving their implementation details unspecified. Some components may be modeled, prototyped, or elaborated at lower levels of abstraction. Top-level design activities include the design of interfaces among components in the architecture and can also include database design.

Formal methods are the incorporation of mathematically based techniques for the specification, development, and verification of software. The OOD can still be regarded as one of the mainstream development models. Obviously we have approaches to describe software architecture according to such concept. As we know, in software engineering, the famous Unified Modeling Language (UML) (Booch, 2005) is a non-proprietary specification language based on the concept of OOD for object modeling. The UML is an effort to create a standard, generic, graphical modeling language for software systems, as a general-purpose modeling language, UML includes a standardized graphical notation used to create an abstract model of a system, referred to as a UML model. A

software designer can describe the system architecture employing UML and kinds of models.

Air traffic congestion is rapidly becoming one of the major commercial transportation challenges at the start of the 21st century as more people take to the skies for their travel needs. “Forecasts indicate a significant increase in demand, ranging from a factor of two to three by 2025.... In short, U.S. competitiveness depends upon an air transportation system that can significantly expand capacity and flexibility, in the presence of weather and other uncertainties, while maintaining safety and protecting the environment”[1].

We describe Autonomous Air traffic flight control System in section 2. In section 3 we provide AATFCS System Architecture Modeling and Analysis. We present the design of architecture for an autonomous air traffic control system using Uml modeling techniques in section 4. In section 5 we provide AADL for an AATFCS system. We presented conclusions in section 6.

2. Autonomous Air Traffic Flight Control System Description

Although the system description of the AATFCS System description provide the information is included here in order to give clarity and context for the analyses of this system.

2.1 AATFCS System Overview

The Automated Air Traffic flight Control System consists of two primary system element types: ground stations and airplanes. The two types are connected via an air-to-ground wireless network and are in constant communication with the other nodes in the network. Each system element type also communicates with other network members of its own type: ground stations within the vicinity of an airport are linked to each other and airplanes communicate with other airplanes within range. Ground stations have additional interfaces with secondary system elements such as external data sources. Airplanes possess their own internal networks which connect on-board subsystems to flight control computers. Each element and its architecture and interfaces are described in further detail in this section. A top-level diagram of the system is shown in Figure 1.

The pilot is assumed to take control at this point for Free Flight during cruise for the reasons previously mentioned in the Background section. However, the pilot could decide to allow the automatic model to continue computing the flight vector and fly the plane based on the last valid commands received and its current position, with updates provided by any “waypoint” ground stations it connects to and authenticates with en route. The airplane does not attempt to connect with another airplane in an ad-hoc air-to-air network until it reaches its destination. As the airplane enters the airspace of the destination airport, it once again connects to and authenticates with the local air-to-ground and air-to-air networks.

The system performs the same actions as during take-off, though in reverse. The pilot, if in command, relinquishes control of the airplane after the data from the local networks has been validated. The airplane then automatically slots itself for approach and landing, in accordance with the ground station’s instructions. After landing, the airplane taxis off the runway and transitions back to pilot control before reaching the gate.

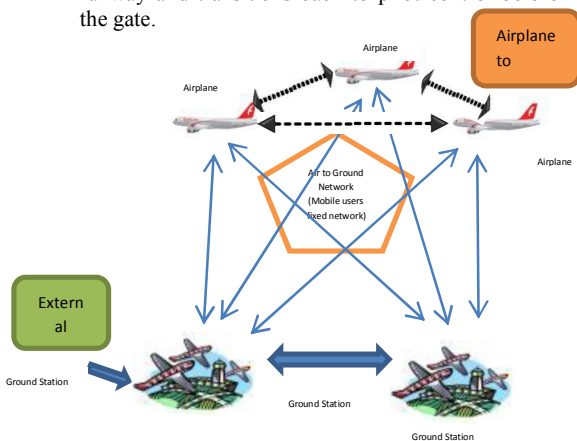


Fig. 1 – Automated Air Traffic Flight Control System

3. AATFCS System Architecture Modeling Analysis

Architectural modeling is an important enabler for the understanding and comprehension of a complex system because it can provide unambiguous representations or views of the system’s architecture and behavior. One definition of a model is “a virtual or physical representation of an entity for purposes of presenting, studying and analyzing its characteristics such as appearance, behavior or performance for a prescribed set of operating environment conditions and scenarios.” Any system can be modeled from numerous points of view which are essentially projections of the system onto one or more operational domains. It should be noted, however, that although models may adequately represent the system for the purpose of further design and implementation, they are still a finite set of projections which limit their ability to exhaustively describe the system and its behavior because of the heuristic which states that “a model is not reality.” It is just as important to be aware of the models’.

Irrespective of their limitations, it is important to develop system models early in the development phase of a program in order for stakeholders – people with an interest in the development or outcome of the design – to develop a common understanding of what the system will look like and how it will operate. Without this, errors from misinterpreting or misunderstanding the system’s characteristics creep into the design and create nontrivial problems (often very big problems) in terms of schedule and cost when the errors are discovered and need to be fixed. In fact, poor communications has been cited as the number.

A picture is worth a thousand words’ is a classic heuristic and a good set of system models can be worth their development cost by preventing errors which, if undiscovered, can propagate to later design, implementation and verification phases.

3.1 Spiral Development Model Stages

“The spiral model is a software development process combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts.” [11] This approach allows for the iterative risk assessment of the design at various stages along the development path and “promotes quality assurance through prototyping at each stage in systems development.” [12] Each loop of the spiral represents a single iteration and each quadrant represents one of four stages of design: determining objectives, alternatives and Constraints; identifying and resolving risks; development testing and planning the next. As the spiral progresses outward from the origin, each successive loop builds on the previous iteration and provides incremental functionality and risk reduction prior to the next loop. The horizontal axis is labeled ‘review’ to indicate the point in the spiral where a review is required before proceeding into the next loop and the vertical axis is labeled ‘cumulative cost’ to show the accumulating cost per

loop.

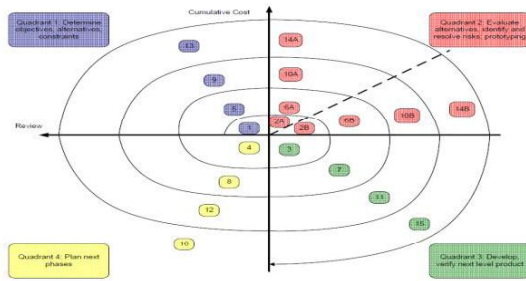


Fig. 2 – AATFCS Spiral Development Model

Figure 2 shows the spiral development model for the AATFCS. The spiral does not start at the origin but instead starts already established in quadrant 1. This is to indicate that the first task to be done in the development of the system is an initial review and determination of objectives, alternatives and constraints at the very top level. The dashed radial line in quadrant 2 is the dividing line between risk (left) and prototype development (right) for a given loop. This shows that the risks must be assessed and a “phase gate” type of evaluation must be passed inured to allow development to continue for that iteration or phase. If the evaluation does not meet its pre-determined criteria, development can be halted or terminated.

4. Design of architecture for an autonomous air traffic flight control system

System architecture is a set of design decisions. These decisions are technical and commercial in nature. To meet the functional and nonfunctional requirements of the above said ATFC system it is necessary to model the complete AATFC system by the use of UML. Different types of diagrams are Request departure clearanceDepartGrant departure clearance designed and described below in brief: UML is perhaps the most well-known commercial industry modeling language today. The unified Modeling Language is a method by which one can “describe a complex system rigorously and unambiguously...such that the integrated system design can be tested and verified to meet requirements before generating any code or designing any hardware,” for the reasons mentioned previously. System modeling takes place in task 3 of the spiral development model, which is early enough to provide assurance that the mission requirements, the overall system architecture, and the subsequent hierarchical decomposition are communicated among and understood by the program stakeholders.

The UML system architecture diagrams presented in this section are described from the use case perspective of an airplane’s approach and landing. However, in order to model the system correctly, a brief discussion to provide understanding of the mission-level operations for this use case shall first be presented.

4.1 Airplane Approach and Landing Description

Some of the high-level description of an airplane’s fault-free approach and landing has been previously mentioned in the discussion of the AATFCS system’s operation. Additional

details which describe the order of events can also be used to help establish the context for modeling the system properly using UML. It should be noted that understanding the system’s fault response, which can be based on a system-level FMECA, is also required to generate a more complete model of the system.

Use Case Diagram

The use case diagram “provides a tool for organizing system requirements in order to understand interactions between:

- “Actors” that make a request, and
- “Activities” made in response by the system

The AATFCS use case diagram in Figure 3 shows the “fault-free arrival” use case for the AATFCS.

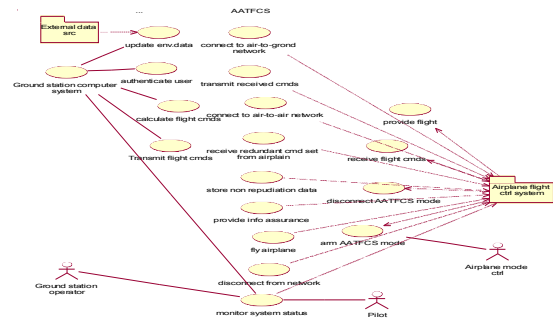


Fig.3 – AATFCS Fault-Free Airplane Arrival Use Case Diagram

The actors and activities in the use case diagram back to the steps in the airplane landing sequence. Note that the Ground Station Operator actor, External Data Source external system and the Update Environmental Data and Provide Information Assurance.

Class Diagram

UML class diagrams “show the static structure of the system at an abstract level” [15].In object oriented programming; classes are abstract representations of software objects, which are, in turn, instantiations of the class. The class diagram in Figure 4 shows representations of the two primary object templates in the AATFCS, the AATFCS subsystem and the network. The highest, most abstracted level of the class hierarchy for net-centric systems would generally depict only a generic object type in the system and possible connection methods (i.e., the network(s)). Each class in the diagram has three fields; from top to bottom, they are the name of the class, the attributes associated with the class, and the methods associated with the class. Additionally, the connections in the hierarchy depict aggregation, inheritance and multiplicity. A closed (filled) diamond indicates that the parent class is comprised of N child level items, with N being a specific or unspecified number or range of numbers such as 1, 1..3 (1 to 3), N, or 1..* (1 or more). The example shows that the AATFCS is comprised of 1 to 5 network types (air-to-air, air-to-ground, ground station network, AATFCS data bus network, or actuation data bus network).

There are three types of lower-level classes called ground control system, SWIM system and airplane connected to the subsystem class. These inherit the attributes and methods listed in the respective fields in their parent class; the

inheritance is shown by the open (unfilled) triangular arrows pointing up from the child classes to the parent class. Each of the three inherited classes has continuing levels of decomposition which are left out of the diagram for clarity except for a few key examples. Similarly, the network class has the five child classes which inherit characteristics from it, as previously described. The empty fields of the child classes indicate that they are not instantiable – they are the equivalent of abstract classes in object-oriented programming.

The SWIM system class is the only class which is decomposed in greater detail in this example; the other classes at this level all decompose to one or more levels further down in the overall hierarchy. The SWIM system class is shown to be comprised of SWIM flight data and system status. The flight data class contains attributes of weather data, airplane flight plan data and pilot authentication data, which are all “inputs” to the class and are annotated as private data (the minus sign preceding the name). Private attributes are not exposed to other classes. The fourth attribute, SWIM data, represents the outgoing message to the ground station and is considered public data (the plus sign preceding the name) because it would be exposed to other objects as part of the transmission process allocated to the public method “provideFlightPlanData ()”. In the system status class, the internal status methods and the data attributes are private and the message attributes, along with the display message method, are public.

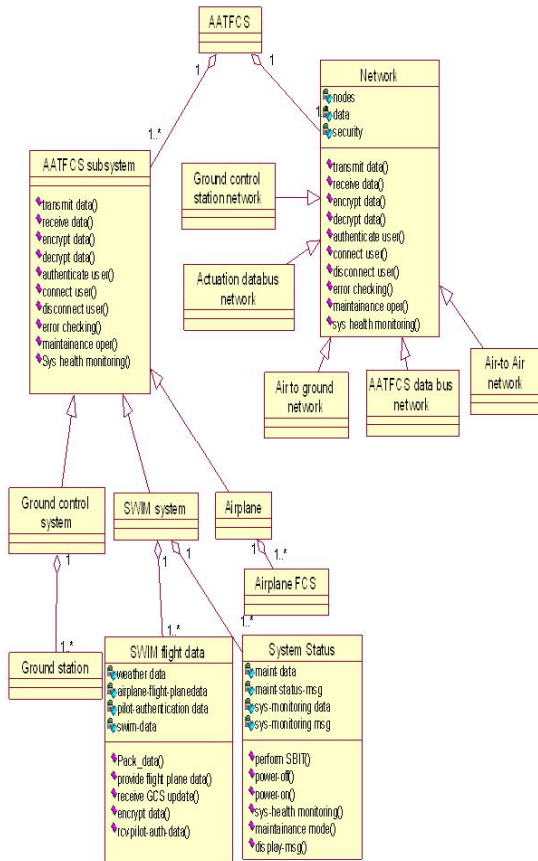


Fig. 4 – AATFCS Top-Level Class Diagram

By continuing this process for all objects, we can derive a hierarchical representation of each object in the system which describes not only the data but also the actions performed on that data.

Sequence diagram

A UML sequence diagram will “model logic flow within a system in a visual manner, especially dynamic modeling of system behavior [14].The sequence diagram does this by showing interactions between objects at various points in time, in sequential order. Time is represented increasing from top to bottom and each entity (e.g., object, actor, etc.) will have a dashed vertical line beneath it which indicates its lifetime within the sequence. Objects in particular are shown as instantiations of their class; they are specified as OBJECT: CLASS. Object lifelines turn into a wider bar (an ‘activation’) upon instantiation and return to the dashed lifeline when the object has been removed from the sequence (i.e., destroyed or de-allocated). “The activation represents an execution of an operation the object carries out. Each activity line contains the name of the message associated with the source object, along with data that is passed to the destination object,

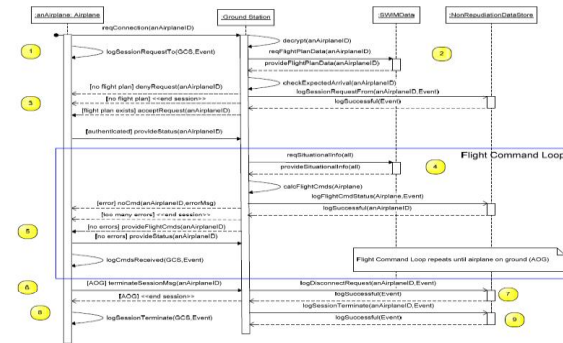


Fig. 6 – AATFCS Airplane Arrival Sequence Diagram Example

5. Architecture Analysis and Design Language for an Autonomous Air traffic Flight control system

A common way of modeling such meta-information in AADL is to associate AADL properties with the item in question and record information about the item. For example, the measurement unit and confidence of data may be recorded in properties. Since AADL is extensible and allows us to introduce new properties, we can define a set of properties specific to the data state variable. In some cases, this Meta information is communicated explicitly with the data and is checked by the application at runtime. In this case, the Meta information is declared to be part of the data representation, either just reflected in the increased size of the data type, or explicitly as a data subcomponent in a data component implementation declaration. State variables are communicated between Ground and Flight systems via telemetry. The data transport mechanism uses State Variables and State Variable Proxies. A State Variable represents the location in the deployment where the state is being locally estimated, and a Proxy State Variable represents a remote location that intends to utilize state variable content remotely. This deployment is shown in Figure 5.

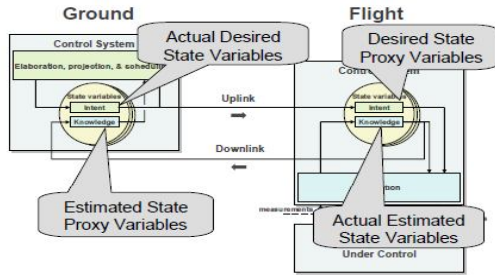


Fig 5 Deployment of State Variables

The deployment of these data is such that Estimators in a deployment update their corresponding State Variables (SV). The data transport mechanism occasionally collects the value histories stored in these SVs and transports these histories to appropriate Proxy SVs in other deployments. The same data transport mechanism is used to transport measurement histories and command histories between deployments (from Basis Hardware Adapters to Proxy Hardware Adapters). Systems engineers specify what information needs to be transported between deployments, and the regularity of proxy updates.

The telemetry transport mechanism is used, then, to update the proxies with actual values with a specified periodicity or on demand. At a high level of abstraction of the AADL model, the state variable proxy notion can be encapsulated in the protocol used by the telemetry (Space Link) bus component. It is the responsibility of the protocol to distribute the state to the out data ports of components to other components. For data port connections across the Space Link bus, a different protocol is used to provide the desired caching strategy of the state variable proxy. The application model is agnostic to this proxy/caching scheme.

If it is desirable to explicitly model the proxy scheme, we can do so in two ways. We can model an implementation of the proxy/caching protocol of the telemetry bus component as a separate AADL model that is associated with the Space Link bus by property. This property is interpreted by the instance model generator to refine the bus abstraction by its implementation. Alternatively, we can model the proxies explicitly as application components (i.e., as threads that receive the original data port content by executing at a specified rate and make it available locally). In this case, users need to modify the model by inserting or removing the proxies as components that are migrated between flight system and ground [16].

```

Package Control Software
Public
-- This type is refined for a AATFCS instance by
refining the
-- Classifiers of the features to be instance specific
Thread group controller
Features
State Estimates In: port group AATFCS Data::State
Estimates In;
Estimate History In: port group Value Histories:
Estimate History Inv;
Control Goals In: port group AATFCS Data::X goals
in;
    
```

```

Commands Out: port group AATFCS
Data::Commands Out;
End controller;
Thread group implementation controller. Basic
End controller. Basic;
    
```

Fig6 Example package of AATFCS system

5.1 Operating System Thread Model

Hardware adapter, estimator, controller, planner, goal executive, and goal monitor are represented by logical threads, each with an execution rate, a deadline, and a worst-case execution time. Some of this functionality may be distributed between flight system and ground or may be distributed within the flight system or ground system. The latter distribution may occur due to a multiprocessor configuration or in anticipation of using multi-core chip architectures in a spacecraft.

Distribution decisions regarding ground or flight system are localized to changes in processor binding property values in the AADL model, unless state variable proxies are modeled explicitly as part of the application system. The collection of logical threads bound to the ground processor the flight processor is then grouped into rate groups. Each member of a rate group is executed by an operating system thread at the period of the rate group. Note that such rate group optimization must take into account execution order requirements between threads of the same rate or of different rates that require data to be communicated mid-frame (i.e., within the same execution cycle).

```

Property set Rate Groups is
Rate Groups : type enumeration ( EstimatorRateGroup,
ControllerRateGroup, PlanExecutionRateGroup,
PlanningRateGroup, HWARateGroup);
AssignedRateGroup : inherit RateGroups::RateGroups
applies to (thread, thread group, process, system);
end RateGroups;
    
```

Fig 7: Rate Group Modeling by Properties

Rate group optimizations can be represented within the current version of AADL using the property mechanism. We can introduce a property type Rate Groups that is an enumeration of rate groups in a particular application and a property to specify the rate group that a thread is assigned to, as illustrated in Figure 7. The enumeration literals are an ordered set. AADL V2 introduces the concept of virtual processor to model hierarchical schedulers. The operating system threads, which execute the tasks of a rate group, act as schedulers that dispatch these tasks as a cyclic executive. Therefore, we represent each of them as a virtual processor to which the application AADL threads are bound. Each of these virtual processors is defined as a subcomponent of a given processor or is defined separately and bound to a processor.

5.2 Binding to Hardware

AADL supports modeling the computer platform of the embedded system. In Figure 8, we illustrate how flight system and ground system computer platforms can be modeled. The flight system consists of a processor, memory, and a flight system bus. In addition, the flight processor has access to a

device bus that is also accessible by devices representing the sensors and actuators outside the MDS computer hardware system component. The ground system consists of a processor, memory, and a ground system bus. The two computer platforms are interconnected via a Space Link bus that represents the downlink between the spacecraft and the ground station. Without having to model the internal details of the hardware, we can use properties to specify characteristics relevant to the analysis of embedded systems.

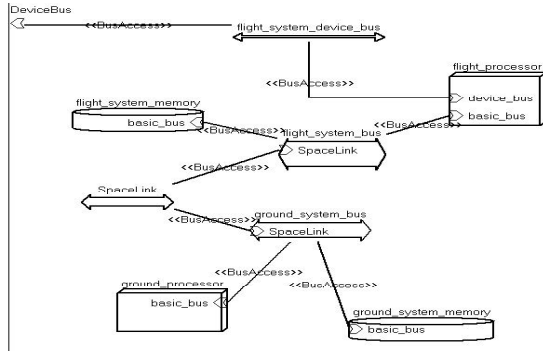


Figure 8 Flight and Ground Processing Systems

The binding of embedded software applications to the computer platform is also accomplished through properties. The Allowed_Processor_Binding property places constraints on the binding to processors. The binding may be constrained to a processor type or to a set of processors. Binding constraints are taken into consideration when a resource allocation tool makes its allocation decisions; the Actual_Processor_Binding property records the actual binding decisions shows the use of Allowed_Processor_Binding for the AATFCS architecture. This property is declared with the top-level system. Implementation allowing the property declaration to refer to the processor as the reference value and to the application component to which the property applies.

```

Package Complete AATFCS system::Camera
Public
System Complete AATFCS system
Extends Complete AATFCS system::Complete
AATFCS System
End Complete AATFCS system;
System implementation Complete AATFCS system.
Camera
Extends Complete AATFCS system::Complete
AATFCS system. Basic
Subcomponents
AATFCS Control System: refined to process
AATFCScontrolSystem::Camera:: AATFCS
ControlSystem.camea;
Controlledsystem: refined to system
SystemUnderControl::Camera::system_under_control.c
amera;
AATFCSPlatform: refined to system
ExecutionHardware::Camera:: AATFCS
Hardware.camera;
flows
TemperatureResponse: end to end flow
    
```

```

AATFCS systemUnderControl.Tempflow ->
SystemtoControllerConn -
AATFCS ControlSystem.ControlFlow ->
ControllertoSystemConn ->
AATFCS systemUnderControl.HeaterCmdFlow
{ Latency => 50 ms;};
properties
Allowed_Processor_Binding =>
reference mdsplatform.ground_processor applies to
AATFCS ControlSystem.OperatorConsole;
Allowed_Processor_Binding =>
reference mdsplatform.ground_processor applies to
AATFCS ControlSystem.GoalElaborator;
Allowed_Processor_Binding =>
reference mdsplatform.flight_processor applies to
AATFCS ControlSystem.GoalExecutive;
Allowed_Processor_Binding =>
reference mdsplatform.flight_processor applies to
AATFCS ControlSystem.StateEstimation;
Allowed_Processor_Binding =>
reference mdsplatform.flight_processor applies to
AATFCS ControlSystem.DeviceControl;
Allowed_Processor_Binding =>
reference mdsplatform.flight_processor applies to
AATFCS SystemUnderControl.Hardware Adapters;
    
```

Fig 9: Modeling of Processor Bindings

6. Conclusions

- This paper has presented an overview of the automated air traffic Flight control system and has performed analyses using different systems architectural modeling methods. A description of the system was provided along with the many architectural views which are indicative of the Complex nature of the system.
- Several of the analyses presented in this paper are exemplary of the ability of models to reduce complexity and increase comprehension of the system architecture in an unambiguous manner. Models also serve to reduce miscommunication and can potentially reduce overruns in development cost, especially if the modeling activity is done sufficiently early in the program, as demonstrated by the spiral development model.
- The architectural analyses highlighted primary areas of interest in defining the AATFCS. The UML analyses presented examples of the system architecture from an object-oriented perspective: the use case, the class hierarchy diagram, and the sequence diagram.
- We presented AADL language for AATFCS system.

7. References

- [1] "NASA &The Next Generation Air Transportation System (NextGen)" (n.d.), from www.aeronautics.nasa.gov/docs/nextgen_whitepaper_06_26_07.pdf.retrieved October 5, 2008.

- [2] “Free flight (air traffic control)” retrieved from Wikipedia(FreeFlight):[http://en.wikipedia.org/wiki/Free_flight_\(air_traffic_control\)](http://en.wikipedia.org/wiki/Free_flight_(air_traffic_control)), retrieved October 21, 2008.
- [3] Cureton, K. SAE 574 Lecture #1: Net-Centric Systems Architecting and Engineering. University of Southern California. (Aug. 26, 2008).
- [4] Cureton, K. SAE 574 Lecture #1: Net-Centric Systems Architecting and Engineering. University of Southern California. (Aug. 26, 2008).
- [5] “Fact Sheet – System-Wide Information Management (SWIM)”, (May 2, 2006), retrieved from Google (System-Wide Information Management):http://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=7129 October 23, 2008
- [6] “OEP Plan Reference Sheet NNEW” (June 19, 2007) from Google(NNEW):http://www.faa.gov/about/office_org/headquarters_offices/ato/publications/oep/version1/reference/nnew/ retrieved October 23, 2008.
- [7] Spitzer, C., The Avionics Handbook. CRC Press LLC. (Ed.). (2001).
- [8] Wasson, C. System Analysis, Design, and Development: Concepts, Principles and Practices. New Jersey : John Wiley & Sons, Inc (2006).
- [9] Rehtin, E. (1991). System Architecting: Creating and Building Complex Systems. New Jersey: Prentice-Hall, Inc. (1991).
- [10] Hines, J. *Systems Engineering Theory and Practice, SAE 541, Session 1*. University of Southern California. (June 2, 2008)
- [11] “Spiral model” retrieved from Google (spiraldevelopmentmodel):http://en.wikipedia.org/wiki/Spiral_model, Dec. 6, 2008
- [12] “Spiral model” retrieved Dec. 6, 2008 from Google (spiraldevelopmentmodel):http://en.wikipedia.org/wiki/Spiral_model Dec. 2, 2008.
- [13] Cureton, K. *SAE 574 Lecture #6: Architecture Modeling Concepts*. University of Southern California. (Oct. 14, 2008).
- [14] Cureton, K. SAE 574 Lecture #7: Architecture Modeling Concepts. University of Southern California. (Oct. 28, 2008).
- [15] Schuller, J. Sams Teach Yourself UML in 24 Hours, Third Edition, Sams Publishing. (2004).
- [16] Society of Automotive Engineers (SAE). “The Architecture Analysis and Design Language (AADL).” Society of Automotive Engineers (SAE) Standard AS-5506 (November 2004) Revised in January 2009 as AS-5506A.
<http://www.sae.org/technical/standards/AS5506A>.



An Efficient Gravitational Search Algorithm Based Optimal Web Service Selection for Composition in SOA

D.Palanikkumar
Department of CSE
Anna University of Technology
Coimbatore

P.Anbuselvan
Department of CSE
Anna University of Technology
Coimbatore

M.Kathiravan
Department of CSE
Anna University of Technology
Coimbatore

Abstract: The Web services model has emerged as a standard for representation, discovery, and invocation of services in a distributed environment. During service composition, Selection of appropriate services for composition is the chief task. A series of tasks are tied together to form a composite web service. Such composition is a challenging task because a number of Web services with the same or similar functions are increasing rapidly. QoS plays a vital role in the problem of selecting the most appropriate web service for composition. It is a measure for how well the composite web service serves the requester. Particle Swarm Optimization (PSO) algorithm and Gravitational Search Algorithm can be used to resolve this problem of optimal service selection. To compose individual web services dynamically according to users' requirements in functional aspects as well as in non-functional preferences. In order to meet the QoS requirements of consumers, this paper presents the QoS calculation of non-functional requirements such as cost, availability, reliability and execution time. To verify the effectiveness in latency of Web Services selection the above two algorithms are compared. Results indicate the Gravitational Search Algorithm improves the latency over the Particle Swarm Optimization algorithm.

Keywords: PSO, GSA, Service Selection.

1. INTRODUCTION

The Web services model has emerged as a standard for representation, discovery, and invocation of services in a distributed environment. During service composition, Selection of appropriate services for composition is the chief task. Many sophisticated tools are required for a web consumer to search for the best service that satisfies his needs. Several web services may exist that provide the same functionality. In such a case, Quality of Service (QoS) is the decisive factor in distinguishing the functionally similar services. QoS-aware Web Service composition is defined as the selection of Web Services maximizing the QoS of the overall Web Service composition, taking into account preferences and constraints defined by the user. Web service composition is gaining a considerable momentum as an approach to the effective integration of distributed, heterogeneous, and autonomous application[12].The process is to search for the optimal set of services that can be composed to create a new service, result in the best QoS with user constraints [2, 18]. Web service has three basic underlying components which form its basic platform; WSDL, UDDI and SOAP [17, 12]. These components are used to select the suitable web services based on client's request. WSDL (Web Service Description Language) is an XML based language. It is used to describe the web services with its functionalities and provides an end point to invoke the web service. A web service is described in the standard XML format of service description using WSDL. It has similar purpose of IDLs (Interface Definition Language) [11, 5].

UDDI have enabled service providers and requesters to find the web service through UDDI Business Registries [20, 4]. In order to select a suitable web services that are published and made available to the clients, every web service must initially publish in some UDDI registries [1, 5].SOAP defines how to communicate the information using XML in web service selection. SOAP specifies the message format and description of message should be transported using HTTP and SMTP [3, 5]. Recently Service selection algorithms are mainly genetic algorithm, Ant colony algorithm, linear programming, and Particle Swarm algorithm and so on [16]. An efficient method is PSO-based service Selection which can be performed by considering user's functional and QoS constraints. Several algorithms exist to perform this chief task of Service selection based on QoS. Each service that exists for accomplishing a specific task can be considered as a particle. So several particles may exist, thus forming a population.PSO can then be applied to find the best solution among the candidates to form an optimal composition. An optimization algorithm called Particle Swarm Optimization (PSO) can be applied to solve this problem of Service selection for composition. Gravitational Search Algorithm is used in an NP hard problem area in a large power system. This optimization deals to find the best location of SVC (Static Var Compensator). The optimization is made on two parameters: location and size [6]. The problem of web service selection for composition can be defined as follows, "In an algorithm based approach of web service selection, which evolutionary search and optimization technique can obtain the best result".

This paper presents a comparative approach to Service selection for Service Composition based on QoS with the user's constraints. Gravitational Search Algorithm (GSA) is one of the optimization algorithms based on law of gravity [19, 3].

2. WEB SERVICE COMPOSITION

Web Service composition aims at selecting and interconnecting Web Services provided by different partners according to a business process. Thus, Web Service compositions can be seen as workflows based on Web Services. There is a workflow model that consists of abstract tasks describing the required functionality (e.g. invoking a credit rating) of a specific workflow step. One of the main issues hereby is the selection of appropriate Web Services that form the execution plan for a Web Service composition. The functionality of each task can be provided by different candidate Web Services. Web Services that provide similar or identical functionality are grouped in the same category [7]. Web Services within the same category may have different non-functional attributes. Fig.1 explains Web Services has many service oriented application. It is one of the web service compositions and selects the best service in the value added services. Existing technologies of web services are extended to give value added customized services to customers through composition. If no single Web service can satisfy the functionality required by the user, there should be a possibility to combine existing services together in order to fulfill the request. The basic web service model consists of three entities: Service Provider, Service Registry and Service Requester [12]. Service Provider is used to provide the service to the Service Registry [5]. Service Requester is the requester who retrieves the information from service registry to find a suitable service provider and publish the web service. The Service Registry contains the information about Service Provider and Service Requester [10].

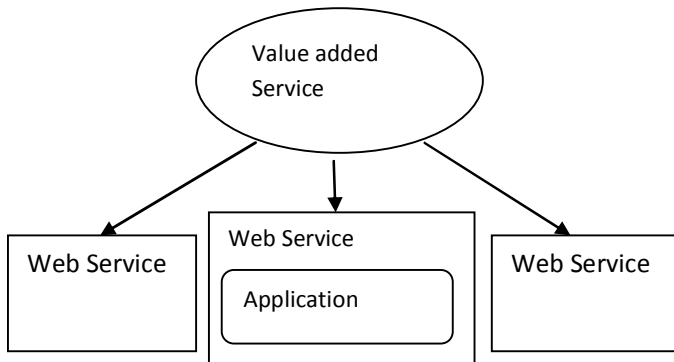


Fig.1 Selection of Web Service Application

Execution cost: The execution cost of an operation of a service is the fee that a service requester has to pay for invoking the service and executing its operation.

Response Time: The response time measures the maximum delay in seconds between the moment when a request is sent and the moment when the results are received by client-side view.

Reliability: The reliability is a measure of trustworthiness of a service. It measures the degree of compliance between providers claimed value with the actual value.

Availability: The availability is the probability that the service is accessible. It is the quality aspect of whether the service is available for immediate use.

2.2 The web services selection problem

Motivating Example: In this section, we present a travel plan domain such as hotel, airline and taxi reservation web service that receives information regarding the type of customer wishes to reserve and makes a reservation to customer needs. Each web service have four individual Services such as Service1, Service2, Service 3, and Service 4. Using the both algorithm we can select the best Services in the Input Details. Fig.2 has shown the diagram for this model. This study describes how Particle Swarm Optimization Algorithm and Gravitational Search Algorithm can be applied to the optimization problem of optimal web service selection and compares the performance of both the algorithms.

3. WEB SERVICE SELECTION BASED ON PSO ALGORITHM

PSO Algorithm introduced by Kennedy and Eberhart in 1995[1]. The Particle Swarm Optimization Algorithm based on user Preferences in web service selection [7]. In this Paper, we mainly solved service selection problem in the web service composition. The PSO-based web service selection method (PSOWSS) to resolve dynamic web services selection with global QoS constraints, considering various QoS attributes, such as response time, cost, and availability etc.

Algorithm

Input: target function;
 Output: Pareto solutions

- 1) Initialization set up parameters;
- 2) In accordance with population size set, randomly generated paths of service composition to meet the constraint. Each path is encoded as a particle, and all particles form the initial particle population;
- 3) Implementing the disturbance moving of particles;
- 4) Randomly selecting current number of mutation particles(PN mutation) from external population, and updating those;
- 5) Updating the values of pbest and gbest;
- 6) According to the evolution times of fitness function to judge whether meeting the end conditions or not.

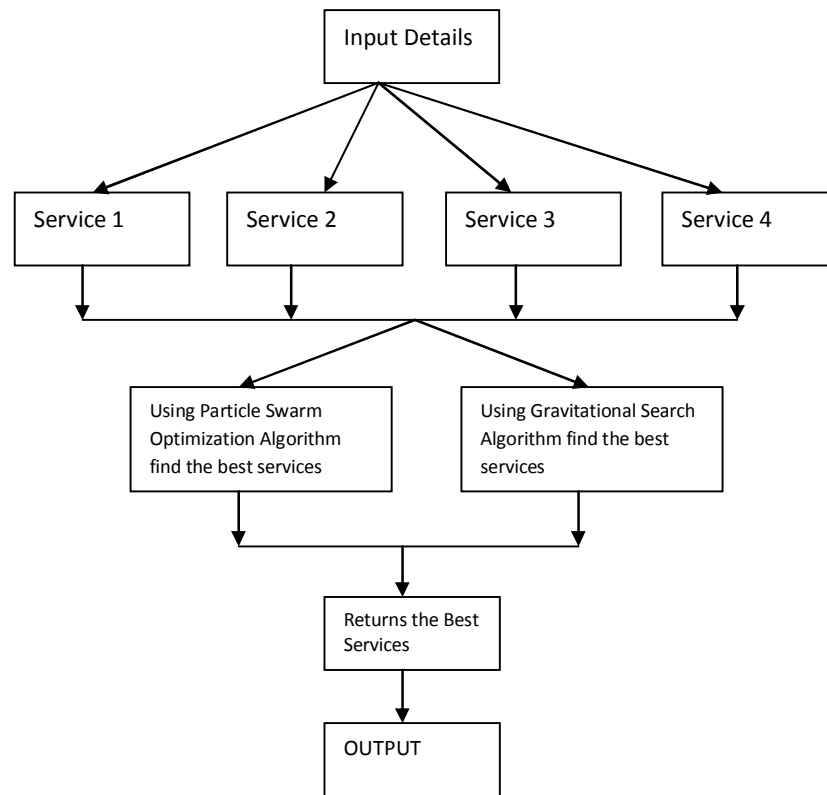


Fig.2 Travel plan domain

Generating Initial Particle Population

Input: population size (N);

Output: initial particle population (SN)

- 1) Set SP to empty;
- 2) Add the sets that are generated by Constr(RanService(WS)) into SP |SP|=|SP| +I;
- 3) If |SP|=N then goto 2 else ending;
- 4) Where WS is the set of service candidates; RanService(WS) is randomly service path generating method and its range is service selection scenarios; Constr() is service selection method based on constraints.

4. WEB SERVICE SELECTION BASED ON GRAVITATIONAL SEARCH ALGORITHM

Gravitational Search Algorithm is based on the law of gravity. It Consider agents as objects and their performance measured by their masses.

Now, consider the travel plan domain created, which is a web service application for flight, hotel and taxi Reservation. The application uses non functional requirements such as price, duration, reputation, rate and availability. Each service has four candidate services.

The fitness value of each service is calculated in web application using the non functional parameter and stored into the new table.

Gravitational and inertial are calculated by the fitness evaluation. The gravitational and inertial masses are updating by the following equations:

$$Mg_i = \frac{(fit(t) - Worst(t))}{(Best(t) - Worst(t))}$$

Here fit(t) calculated by the non functional requirement parameters and represents the fitness value of the each services at time t.

Fit t

$$= \frac{(Qprice + Qduration + Qreputation + Qrate + Qavailability)}{Qtime}$$

Where fit(t) represent the fitness value of the agents i at time t, worst(t) and best(t) are defined as follows:

Best (t) = Max fit (t)

Worst (t) = Min fit (t)

At a specific time 't', the fitness value is calculated by:

$$Fit t = \frac{(gt * Mgi)}{(Rij t + es)} * Fit(t)$$

Here gt is gravitational constant at time t, es is small constant. fitness is fit (t) value of each services.

$$R_{ij}(t) = (euclidian1 + euclidian2)^2$$

R_{ij}(t) is the Euclidian distance between two services. Here we can calculate the Euclidian1 using the flight and hotel services and Euclidian2 using the Euclidian1 and car services.

$$Euclidian1 = (flight service - hotel service)^2$$

$$Euclidian 2 = (flight service - taxi service)^2$$

Execution #	Parameters				Latency(ms)	
	Availability	Reliability	Cost	Reputation	PSO	GSA
1	{0,1}	{0,1}	Varied	Ranked	375	156
2	{0,1}	{0,1}	Varied	Ranked	140	109
3	{0,1}	{0,1}	Varied	Ranked	109	125
4	{0,1}	{0,1}	Varied	Ranked	125	93
5	{0,1}	{0,1}	Varied	Ranked	375	109

Table 1 Comparison between PSO and GSA

5. EXPERIMENTAL RESULTS

5.1 Comparing the Fitness Functions

In this paper we compared the fitness functions of Particle Swarm Optimization algorithm and Gravitational Search algorithm. In PSO algorithm updating is performed without considering the fitness value. In GSA, fitness value is important around the search space. In GSA fitness is reversely proportional to the distance between solutions. Particle Swarm Optimization algorithm Calculates the fitness values using Group of iteration, then compute the best values from the global values of iteration. So the time consumption is more. But, in GSA, choose fitness as random iteration of web service for calculate the fitness value. So the time consumption much lower compared to Particle Swarm Optimization algorithm

5.2 Comparing PSO with GSA

The comparison results shown in Fig.3. For the particular time t, Gravitational Search algorithm performs better than PSO algorithm. Gravitational Search algorithm solving the NP-hard problem like web service selection better than PSO algorithm. This proves all the particle be likely to converge to the best solution quickly compared to Particle Swarm Optimization algorithm. This is represented in the figure. (Refer Table 1)

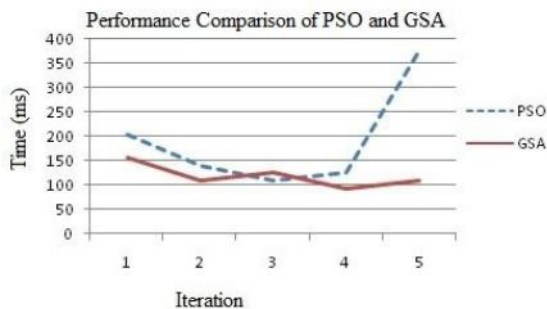


Fig. 3 Comparison of PSO vs GSA

6. CONCLUSION

This Paper explains the web service selection problem by considering a travel plan domain as an example. The algorithm can generate a set of optimal plan to meet user constraints. We have presented an algorithmic approach for solving the optimal service selection optimization problem by considering two optimization algorithm, PSO and GSA algorithm. PSO-based web service selection selects the optimal Web Services for each task so that the overall QoS and cost requirements of the composition are satisfied. A travel agency application is considered to facilitate dynamic service composition, selection and adaptation in order to compose and to adjust the travel process to meet tourists' needs regardless of their locations, platforms and/or hardware

speeds. The proposed Gravitational Search Algorithm based on user preference in the Web Service Selection is obtaining better results. The future work may be achieve towards incorporations other evolutionary Computing Algorithm the same.

REFERENCES

- [1]. Al-Masri E., Mahmoud Q. H.,(2007) "Discovering the best web service", (poster) 16th Intl. Conference on World Wide Web (WWW), 2007, pp. 1257-1258. (For QWS Dataset Version 1.0 or QWS Dataset Version 2.0).
- [2]. Canfora G.,(Penta M., Di,Esposito R., and Villani M.,(2004) "A lightweight approach for QoS-aware service composition", In Proc. of the 2nd Intl. Conference on Service Oriented Computing, New York, USA. 2004.
- [3]. Demian Antony D'Mello, Ananthanarayana. "A Review of Quality of Service (QoS) Driven Dynamic Web Service Selection Techniques" ICIIS 2010, Jul 29-Aug 01,2010, India.
- [4]. Demian Antony D' Mello, V.S.Ananthanarayana., (2009) "A Tree Structure for Web Service Composition".@2009.
- [5]. Dmytro Zhovtobryukh., "A Petri Net-based Approach for Automated Goal-Driven Web Service Composition". Simulation 2007; 83; 33.
- [6]. Esmat Rashedi, Hossien Nezamabadi-pour, Saeid Saryazdi, Malihe M. Farsangi. (2010) "Allocation of Static Var Compensator Using Gravitational Search Algorithm".IEEE Conference on Web Service 2010.
- [7]. Junsun, BinFeng, Wenboxu, "Particle swarm Optimization with particles having Quantum Behavior".
- [8]. Joyce El Haddad, Maude Manouvrier, and Marta Rukoz, (2010) "TQoS: Transactional and QoS-Aware Selection Algorithm for Automatic Web Service Composition", IEEE Transactions on Services Computing, Vol. 3, No. 1.
- [9]. Kennedy, I. and Eberhart, R. "Particle swarm Optimization", Pmc.IEEE int.Conf. On Neural Network, 1995: 1942-1948.
- [10]. Prashanth, B. and Narahari, Y. "Efficient Algorithm for Combinatorial Auctions with Volume Discounts Arising in Web Service Composition".
- [11]. Manoharan, R. Archana, A. and Siddhika Cowlagi (2011) "Hybrid Web Services Ranking Algorithm". IJCSI, Vol 8, Issue 3, No.2, May 2011.

- [12]. Ming Chen, Zhen-wu Wang, (2007) "An Approach for Web Services Composition Based on QoS and Discrete Particle Swarm Optimization", ACIS Intl. Conference on Software Engineering, AI, Networking, and Parallel/Distributed Computing, 2007, vol. 2, pp. 37-4.
- [13]. Martine De Cock, Sam Chung, Omar Haffez, (2007) "Selection of Web Services with Imprecise QoS Constraints", Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence.
- [14]. Rashedia, E. Nezamabadi-pour, H. and Saryazdi, S. (2009) "GSA: A Gravitational Search Algorithm", Elsevier, New York, NY, USA, June 2009, pp. 2232-2248.
- [15]. Su Myeon Kim, Marcel-Catalin Rosu., (2004) "A Survey of Public Web Services". www 2004, May 17-22, 2004, New York, USA.
- [16]. Shuzhi LI, Ping SHEN, Shux in YANG, "A Grouping Particle Swarm Optimization Algorithm for web service based on user Preference".
- [17]. W3C Working Group. Web services architecture. <http://www.w3.org/>.
- [18]. Wei-Hua Ai, Yun-Xian Huang, Hui Zhang, and Ning Zhou, (2008) "Web Services Composition and Optimizing Algorithm Based on QoS", 4th Intl. Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08, Oct. 2008.
- [19]. Zibanezhad, B. Zamanifar, K. Nematbakhsh, N. Mardukhi, F. (2009) "An Approach for Web Services Composition Based on QoS and Gravitational Search Algorithm". Whitepaper Microsoft copyrights@2009.
- [20]. Zhang Liang-Jie, Li Bing, and Chao Tian et al., (2003) "On demand Web services-based business process composition", In Proc. of the IEEE Intl. Conference on System, Man, and Cybernetics, Washington, USA, 2003.



BLOOM FILTERS & THEIR APPLICATIONS

Saibal K. Pal
Defence R & D Organization
SAG, Metcalfe House
Delhi, India

Puneet Sardana
Department of Computer Science
University of Delhi
Delhi, India

Abstract: A Bloom Filter (BF) is a data structure suitable for performing set membership queries very efficiently. A Standard Bloom Filter representing a set of n elements is generated by an array of m bits and uses k independent hash functions. Bloom Filters have some attractive properties including low storage requirement, fast membership checking and no false negatives. False positives are possible but their probability may be controlled and significantly lowered depending upon the application requirements. There are many variants of the standard Bloom Filter – counting BF, variable increment BF, compressed BF, scalable BF, generalized BF, stable BF and Bloomier Filter. Bloom Filters are increasingly finding applications in fast and approximate search, encrypted search in the cloud, routing and controlling of network traffic, network intrusion detection and differential database and file updating. This paper explores the typical properties of Bloom Filters, their variants and their suitability for use in present day applications.

Keywords: Bloom Filter, Variants, Set Membership, Hashing and Encrypted Search.

1. INTRODUCTION

A Bloom Filter is a space efficient probabilistic data structure which is used to represent a set and perform membership queries [1] i.e. to query whether an element is a member of the set or not. The Bloom Filter data structure was introduced by Burton H. Bloom [2] in 1970. A Bloom Filter occupies negligible space compared to the entire set. Space saving comes at the cost of false positives but this drawback does not affect the processing of information if the probability of an error is made sufficiently low. Bloom Filters typically find applications in situations that involve determining membership of an element for a sufficiently large set in small amount of time. Today, Bloom Filters are used in wide variety of applications including spell checking, network traffic routing and monitoring, database search, differential file updating, distributed network caches, and textual analysis. In this paper we will describe bloom filter, its variants and its applications in different areas of computer science.

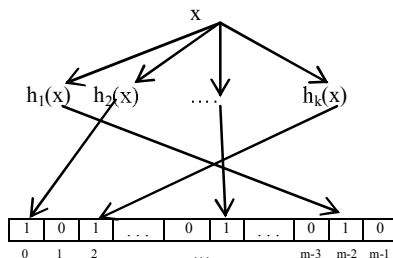


Fig. 1 Set Operation in a Standard Bloom Filter

1.1 Standard Bloom Filter

A Bloom Filter for representing a set $S = \{s_1, s_2, \dots, s_n\}$ of n elements is described by an array of m bits, initially all set to 0. A Bloom Filter uses k independent hash functions h_1, h_2, \dots, h_k with range $\{1, \dots, m\}$. Each hash function maps every item to some random number over the range $\{1, \dots, m\}$. For each element $x \in S$, the bits $h_i(x)$ are set to 1 for $1 \leq i \leq k$.

To check if an item y is in S , we check whether all $h_i(y)$ are set to 1. If any of $h_i(y)$ is 0 then clearly y does not belong to S . If all $h_i(y)$ are set to 1 then y may belong to S .

Following are the properties of a Standard Bloom Filter:

- The amount of space needed to store the Bloom Filter [3] is very small as compared to the entire set.
- The time needed to check whether an element is present or not is independent of the number of elements present in the set.
- False negatives are not possible. False positives are possible but their probability can be significantly lowered.
- Bloom Filters can be easily halved in size allowing applications to shrink a Bloom Filter.
- Bloom Filters can also be used to approximate the intersection between two sets.
- If two Bloom Filters represent sets S_1 and S_2 with same number of bits and same number of hash functions then a Bloom Filter representing the union of these two sets can be obtained by taking OR of the two bit-vectors of the original Bloom filters [4].

The remaining part of the paper is organized as follows. Section 2 describes different variants of the bloom filter. Different applications of bloom filters in area of approximate / encrypted search, network security and database updating are

explained in Section 3. Concluding remarks are mentioned in Section 4.

2. VARIANTS OF BLOOM FILTER

In this section we explore and describe variants of Bloom Filter [5] built on the Standard Bloom Filter data structure.

2.1 Counting Bloom Filter

The Standard Bloom Filter works fine when the members of the set do not change over time. Addition of elements only requires hashing the additional item and setting the corresponding bit locations in the array. However, deletion is not possible in the Standard Bloom Filter since it will require setting 0's in the array to already set 1's that was result of hashing another item which is still a member of the set. To overcome this deficiency of Standard Bloom Filter, Fan et al. [6] introduced the idea of Counting Bloom Filter. In Counting Bloom Filter, bits of the array are replaced by a small counter. When an element is inserted, the corresponding counters are incremented; when the element is deleted, the corresponding counters are decremented. The value of the counter gives the number of items hashed to it. Since each counter size is limited, the n-bit counter will overflow if it reaches a value of 2^n . The figure below shows the structure and set operation in a Counting Bloom Filter. Analysis carried out by Fan et al shows that a 4-bit counter is adequate for most applications.

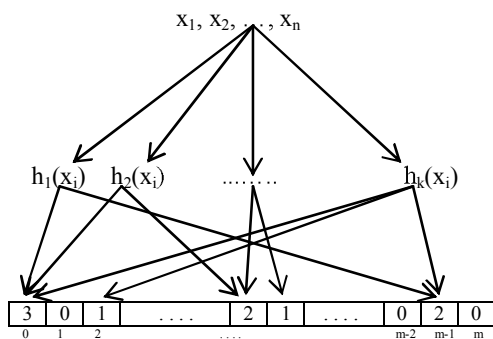


Fig. 2 Set Operation in a Counting Bloom Filter

2.2 Variable Increment Bloom Filter

The Variable Increment Counting Bloom Filter (VI – Bloom) [7] is a generalization of the Counting Bloom Filter that uses variable increments to update each entry. In this structure, a set of possible variable increments are defined. For each counter update by an element we hash the element into the variable increment set and use it to increment the counter. Similarly, to delete an element we decrement by its hashed value in the variable increment set. To determine if an element is part of the set, we check in each of its counters if its hashed value in the variable increment set could be part of the sum. If this be the case in at least one counter the element definitely does not belong to the set. Otherwise, the element may belong to the set with some probability of false positive. VI – Bloom can be used in Approximate Concurrent State Machine [8], Counter Braids [9] and Fingerprint-based schemes.

2.3 Compressed Bloom Filter

Using a larger but sparser Bloom Filter can yield the same false positive rate with a smaller number of transmitted bits. The resulting bloom filter is called a Compressed Bloom Filter [10]. By using Compressed Bloom Filters networking

protocols reduce the number of bits broadcast, the false positive rate and the amount of computation per look up. Costs involved are larger computation time for compression and decompression.

2.4 Scalable Bloom Filter

A Scalable Bloom Filters consist of two or more Standard Bloom Filters, allowing arbitrary growth of the set being represented. When one Bloom Filter gets filled due to the limit on the fill ratio, a new filter is added. Querying an element involves testing the presence in each filter. Each successive bloom filter is created with a tighter maximum error probability on a geometric progression [11].

2.5 Generalized Bloom Filter

Generalized Bloom Filter [12] uses hash functions that can set as well as reset bits. In Generalized Bloom Filter, the initial value of the bits of the array is not restricted to zero anymore. For each element $x_i \in S$ bits corresponding to the positions $h_1(x_i), h_2(x_i), \dots, h_k(x_i)$ are set and the bits corresponding to the positions $g_1(x_i), g_2(x_i), \dots, g_k(x_i)$ are reset. In case of collision between function h_j and g_j the resulting bit is always reset. To check if element belongs to the set we check whether bits corresponding to h_j are all set and bits corresponding to g_j are all reset. If at least one bit is inverted then the element does not belong to the set with high probability. If no bit is inverted, then the element belong to set with high probability.

2.6 Bloomier Filters

Bloomier Filters [13] associate a value with each element that had been inserted thereby implementing an associative array. These structures achieve a small space overhead by accepting a small probability of false positives. In this type of Bloom Filter, a false positive is defined as returning a result when the key is not in the map. The map will never return the wrong value for a key that is in the map.

2.7 Stable Bloom Filter

This variant of Bloom Filter is particularly useful in data streaming applications. In these applications when more and more elements arrive, the number of 1's in the array of bloom filter will increase significantly, finally reaching the limit where every distinct element is reported as duplicate indicating that bloom filter can no longer be used. In Stable Bloom Filters [14], this state is avoided by eviction of some information. In this approach a random deletion operation is incorporated in the Bloom Filter so that it does not exceed its capacity.

3. APPLICATIONS OF BLOOM FILTER

3.1 General Applications

3.1.1 Spell Checkers

Bloom Filters are particularly useful in spell checking software. They are used to determine if the word is a valid word in its language. This is done by creating Bloom Filter of all possible words of that language and checking a candidate word against that Bloom Filter. Suggested corrections are generated by making all single substitutions in rejected words and then checking if these results are members of the set [15].

3.1.2 Longest Prefix Matching

Bloom Filters are used for longest prefix matching algorithms [16] as these are typically used for efficient exact match searches. The basic idea is to create a hash table consisting of prefixes of various lengths that have to be potentially matched against a given string with the goal of finding the longest possible match. By using Bloom Filter one can avoid

unnecessary look up into a hash table when the corresponding prefix does not exist in the table.

3.1.3 Refining Web Search Results

Bloom Filters are extremely useful in refining search results [17] returned by search engines. Most of the top search results returned by search engines contain similar contents. This technique involves removing or grouping all near-duplicate documents in the results presented to the user. Bloom Filter is also used for similarity detection of text documents. For finding similar documents, Bloom Filters are compared by using bit wise AND operation. In case the two documents share large number of 1's after applying bit-wise AND to their bloom filter, the documents are assumed to be similar.

3.2 Networking Applications

3.2.1 Routing

If the network is in the form of a rooted tree with nodes holding resources and a node receives a request for resource, it checks its unified list to ascertain if it has a way of routing that request to the resource [18]. False positives in this cause may forward the routing request to an incorrect path. In such a case backtracking of the tree is necessary. Another similar application needs to verify if the requested file has a replica nearby and in such cases the request may be routed efficiently along the shortest path [19, 20]. Each node in the network keeps an array of Bloom Filter for each adjacent edge i.e. The k^{th} Bloom Filter in the array keeps track of the files reachable via k hops from the node in the network.

Bloom Filters are also used for geographic routing system for mobile computers [21]. In this scheme each node contains a Bloom Filter representing the list of mobile hosts reachable through itself or through its three siblings at each level.

3.2.2 Loop Prevention

Normally, packets trapped in the network loop are detected using the IP Time-To-Live field but these are not of much help if the loops are small. A small Bloom Filter can be used which can be carried in the packet header and which keeps track of the set of nodes visited [22]. Each node has a mask that can be ORed into the Bloom Filter as it passes; if the filter does not change there is a loop.

3.2.3 IP Traceback and IP Multicast

Bloom filter is also used to trace the route that a packet traversed in a network [23]. Bloom Filters reduce the amount of information [24] that needs to be stored in order to summarize the set of packets seen. A router mistakenly identifying a packet as having been seen would be treated as a false positive.

Bloom filters are also used as alternative of interface lists that the router associates with multicast addresses to send packets through a multicast tree [25]. There can be Bloom Filter of addresses associated with each interface. When a packet with multicast addresses arrives on one interface, the Bloom Filters of all other interfaces are queried to check if packets with that address should be forwarded along that interface. This helps in significant space savings.

3.2.4 Network Traffic

Bloom filters are widely used to reduce network traffic. Bloom filters are used in caching proxy servers [26] on the World Wide Web (WWW). Bloom filters are used in Web caches to efficiently determine the existence of an object in cache. Use of web caches help to reduce the network traffic.

Bloom filter are also used as cache digest. A cache digest contains information of all cache keys with lookup capability. By checking a neighbor cache, a cache can determine with certainty if a neighboring cache does not hold a given object [27]. This allows in reduction of the cache directory size while keeping the number of collisions low.

Bloom Filters find applications in network traffic measurement and detection of heavy flows inside a router [28]. The basic idea is to hash each packet entering into the Bloom Filter. A counter is associated with each location in the Bloom Filter that records the number of packet bytes that have traversed the router associated with that location. The counter is incremented by the number of bytes in the packet. If a minimum count associated with a packet is above some threshold, the corresponding flow is marked as heavy flow.

3.3 Applications in Security & Database Management

3.3.1 Intrusion Detection

Network Intrusion Detection and Prevention Systems (IDS/IPS) use string matching to scan Internet packets for malicious content. Bloom Filters are particularly useful for searching large number of strings efficiently. The basic idea is to find substrings (or commonly known as signatures) at high speed [29, 30]. A common approach is to separate signatures by length and use Bloom Filter for each length allowing parallel processing. If the Bloom Filter detects a match, a hash table is queried to determine if exact match has occurred. If the queried signature is exact match, the malicious content can be blocked and the network administrator is informed. Google Chrome uses Bloom Filters to make preliminary decision whether a particular web site is malicious or safe. Bloom filters are also used in virus scanning [31], worm detection [32], Denial of Service (DoS) prevention [33] and network forensics [34].

3.3.2 Encrypted Search

Bloom filters are extremely useful for searching in encrypted text. At the client end, user first creates the Bloom Filter of the document, encrypts the document using an encryption algorithm and then sends both the encrypted document as well as its corresponding Bloom Filter to the server. When the client needs to search the document, it sends keyword to the server and the server checks the document Bloom Filter for presence of the keyword. If presence of the keyword is established, the encrypted document is returned to the client which is decrypted with the key (used earlier to encrypt the document).

3.3.3 Database Applications

Bloom Filters have been frequently used for management of databases. Bloom Filters were used to estimate the size of joins in databases [35, 36] and to speed up semi-join operations. This is particularly useful in distributed databases. In these applications, one host sends the other host information in the form of Bloom Filter to reduce the overall communication load between two hosts.

Bloom filters can also be used to maintain differential files [37]. A differential file keeps track of all changes to a database that occurred during the day or within a specific time period. Instead of keeping a list of all records that are being changed, one can replace this list with Bloom Filters of the records that have been changed.

4. CONCLUSION

The significance of Bloom Filters has been highlighted in this paper. Variants of Standard Bloom Filter were explored, which have been modified according to the requirement of different applications. We have also explained various applications of Bloom Filters. This simple data structure is gaining significance particularly for applications related to searching of documents, databases and encrypted content on the cloud. Applications related to network traffic management, database management and cloud security are also being addressed using Bloom Filters. The Standard Bloom Filter may be modified according to the needs of the application so that more power can be derived from this data structure. In the future, we are interested in applications related to cloud security and encrypted search and would like to modify this data structure to make it more suitable for these applications.

5. REFERENCES

- [1] Peter Brass, *Advanced Data Structures*, Cambridge University Press, 2008, pp. 402-405.
- [2] Burton H. Bloom, Space/time trade-offs in Hash Coding with Allowable Errors, *Communications of the ACM*, Volume 13, Issue 7, 1970, <http://portal.acm.org/citation.cfm?doid=362686.362692>.
- [3] Jing Chi, Research and Application on Bloom Filter, *Applied Computing, Computer Science and Advanced Communication, First International Conference on Future Computer and Communication, FCC 2009, China, June 2009*, pp. 30-33.
- [4] Andrei Broder and Michael Mitzenmacher, *Network Applications of Bloom Filters*, *Internet Mathematics Vol. 1*, pp. 492-505.
- [5] Graham Cormode and Marina Thottan, *Algorithms for Next Generation Networks*, Springer, 2010, pp. 185-189.
- [6] L. Fan, P. Cao, J. Almeida and A. Z. Broder, Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol. *IEEE/ACM Transactions on Networking*, 2000.
- [7] Ori Rottenstreich and Issac Keslassy, *The Variable-Increment Counting Bloom Filter*, Technion, Israel.
- [8] F. Bonomi, M. Mitzenmacher, R. Panigrahy, S. Singh and G. Varghese, Beyond Bloom Filters: from Approximate Membership Checks to Approximate State Machines. *SIGCOMM*, 2006.
- [9] Y. Lu, A. Montanari, B. Prabhakar, S. Dharmapurikar, and A. Kabbani, Counter Braids: a Novel Counter Architecture for Per-slow measurement, *SISMETRICS*, 2008.
- [10] Michael Mitzenmacher, Compressed Bloom Filters, *IEEE/ACM, Transactions on Networking*, 2002, pp. 604-612.
- [11] Almeida, Paulo; Baquero, Carlos; Pregoica, Nuno; Hutchison, David (2007), Scalable Bloom Filters, *Information Processing Letters*, pp.255–261.
- [12] Rafael Laufer, Pedro B. Velloso, and Otto Carlos M. B. Duarte, A Generalized Bloom Filter to Secure Distributed Network Applications, *Computer Networks*, vol. 55, no. 8, pp. 1804-1819, June 2011.
- [13] Chazelle, Bernard; Kilian, Joe; Rubinfeld, Ronitt; Tal, Ayellet, The Bloomier Filter: an Efficient Data Structure for Static Support Lookup Tables, *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 30–39, 2004.
- [14] Deng, Fan; Rafiei, Davood, Approximately Detecting Duplicates for Streaming Data using Stable Bloom Filters, *Proceedings of the ACM SIGMOD Conference*, pp. 25–36, 2006.
- [15] James K. Mullin and Daniel J. Margoliash, A tale of three spelling checkers, *Software, Practice and Experience*, pp. 625- 630, June 1990.
- [16] S. Dharmapurikar, P. Krishnamurthy and D.E. Taylor. Longest prefix matching using Bloom Filters, *IEEE/ACM Transactions on Networks*, pp. 397-409, 2006.
- [17] Navendu Jain, Mike Dahlin and Renu Tewari, Using Bloom Filters to Refine Web Search Results, *Eighth International Workshop on the Web and Databases*, 2005.
- [18] S. Czerwinski, B. Y. Zhao, T. Hodes, A. D. Joseph and R. Katz. An Architecture for a Secure Service Discovery Service. In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 24-35, ACM Press, 1999.
- [19] S. C. Rhea and J. Kubiatowicz. Probabilistic Location and Routing. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, Volume 3, pp. 1248-1257, IEEE Computer Society, 2002.
- [20] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A Scalable Content-Addressable Network. *ACM SIGCOMM Computer Communication Review, Proceedings of the 2001 SIGCOMM Conference*, 2001.
- [21] P. Hsiao, Geographical Region Summary Service for Geographical Routing. *Mobile Computing and Communications Review*, 2001, pp. 25-39.
- [22] A. Whitaker and D. Wetherall. Forwarding without Loops in Icarus. In *Proceedings of the Fifth IEEE Conference on Open Architectures and Network Programming (OPENARCH)*, pp. 63-75, Los Alamitos, CA, IEEE Computer Society, 2002.
- [23] R. P. Laufer, P. B. Velloso, D. d. O. Cunha, I. M. Moraes, M. D. D. Bicudo, M. D. D. Moreira, O. C. M. B. Duarte, Towards stateless single-packet IP traceback, *Proceedings of the 32nd IEEE Conference on Local Computer Networks*, IEEE Computer Society, Washington, DC, USA, 2007, pp. 548–555.
- [24] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer. Hash-Based IP Traceback. *ACM SIGCOMM Computer Communication Review, Proceedings of the 2001 SIGCOMM Conference*, (2001).
- [25] B. Gronvall, Scalable Multicast Forwarding, *Computer Communication Review* 32, 2002.
- [26] Jia Wang. A survey of web caching schemes for the internet. *ACM SIG-COMM Computer Communication Review*, 1999.
- [27] James Blustein and Amal El- Maazawi, Bloom Filters – A Tutorial, Analysis, and Survey, 2002.
- [28] C. Estan and G. Varghese. New Directions in Traffic Measurement and Accounting. *ACM SIGCOMM Computer Communication Review, Proceedings of the 2002 SIGCOMM Conference*, 2002, pp. 323-336.

- [29] S. Dharmapurikar, P. Krishnamurthy, T.S. Sproull and J.W. Lockwood. Deep packet inspection using Parallel Bloom Filters, IEEE Micro, pp. 52-61, 2004.
- [30] S. Dharmapurikar and J.W. Lockwood, Fast and Scalable Pattern Matching for Network Intrusion Detection Systems, IEEE Journal on Selected Areas in Communications, 2006.
- [31] O. Erdogan and P. Cao, Hash-AV: Fast Virus Signature Scanning by Cache-resident Filters, in IEEE Globecom 2005, St Louis, MO, 2005.
- [32] G. Gu, D. Dagon, X. Qin, M. I. Sharif, W. Lee, and G. F. Riley, Worm detection, early warning, and response based on local victim information, in In Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC 2004), Tucson, Arizona, 2004.
- [33] Y. Kim, W. Lau, M. C. Chuah, and H. J. Chao, Packetscore: statistical based overload control against distributed Denial-of-Service, in 23rd Annual IEEE Conference on Computer Communications (INFOCOM), Hong Kong, 2004.
- [34] K. Shanmugasundaram, H. Bronnimann, and N. Memon, Payload attribution via hierarchical Bloom Filters, in 11th ACM Conference on Computer and Communications Security, Washington, DC, 2004.
- [35] James K. Mullin. Optimal semijoins for distributed database systems. IEEE Transactions on Software Engineering, May 1990.
- [36] James K. Mullin. Estimating the size of a relational join. Information Systems, 1993.
- [37] L. L. Gremilion. Designing a Bloom Filter for Differential File Access. Communications of the ACM 25, 1982, pp. 600-604.



A Comparative Study between LSB and Modified Bit Replacement (MBR) Watermarking Technique in Spatial Domain for Biomedical Image Security

Koushik Pal
Institute of Radio Physics and
Electronics, University of
Calcutta, Kolkata, 700009,
India

Goutam Ghosh
Institute of Radio Physics and
Electronics, University of
Calcutta, Kolkata, 700009,
India

Mahua Bhattacharya
Indian Institute of Information
Technology and Management,
Gwalior, 474010, India

Abstract: The current paper proposes a biomedical image watermarking technique through modified bit replacement algorithm in spatial domain. In this algorithm, multiple copies of the same information are hidden in the biomedical images. Detection is performed by applying bit majority algorithm which can reconstruct the hidden information from different recovered sets subjected to some common attacks. Keys for watermark recovery are also embedded in the cover image. The superiority of the proposed modified bit replacement scheme over the LSB technique is indicated through evaluation of image quality metrics that shows much enhancement in the visual and statistical invisibility of hidden data.

Keywords: Biomedical image watermarking, spatial domain, data security, image quality metrics.

1. INTRODUCTION

The rapid growth and advancement of digital networks and multimedia technologies has created immense interest in the areas of tele-diagnosis tele-medicine, tele-radiology, tele-surgery and remote patient monitoring. The digitization of patient information, such as electronic patient records (EPR), clinical and diagnostic images, has provided significant flexibility and more accuracy in medical diagnosis. Patient records are required to be secure and information confidentiality maintained. For biomedical images, modifications are not allowed during data transfer over networks for obvious legal reasons. Digital watermarking can embed messages without changing the image size and without violating the DICOM format maintaining the following necessary conditions:

- I. There should be minimal perceptible changes in the watermarked image. The watermarked image should visually be the same as the original image [1].
- II. The watermarking technique should be reversible. This means that the watermarked image should revert back to its original form on removal of the water mark [2].
- III. There should be no impact on the stored images in the PACS server due to introduction of the watermark [3].
- IV. Modification of the watermarked image may lead to unsuccessful verification. So the proposed watermarking scheme should not change the amount of data that needs to be transferred.
- V. The watermarking technique for authentication should be applied while transferring image data in DICOM format over the network [4].

Digital watermarking is one of the safest and popular methods to enhance medical data security [5]. It is the process of embedding information into a digital image with an imperceptible form for the human visual system such that the hidden information or the watermark can be extracted or recovered afterwards [6-9]. However, medical image watermarking requires extreme care when embedding additional information because the additional information should not degrade the medical image quality.

If a medical image is illegally obtained and the content is changed, it may lead to wrong diagnosis. Watermarking of medical images and authentication of legal documents face extremely hostile environments where the most harmful attacks remove the embedded watermarks [10, 11].

A biomedical image watermarking technique can be characterized by the following four features: imperceptibility, robustness, security, and capacity.

Imperceptibility: Imperceptibility refers to the perceptual transparency of the watermark. Ideally, no perceptible difference between the watermarked and original image should be perceived by the human visual system.

Robustness: Robustness is the capability of the watermark to survive unwanted alterations or manipulations known as attacks. A watermark needs only to survive the attacks that are likely to occur when the watermarked signal is being transmitted. Not all watermarking applications require a watermark to be robust enough to survive all attacks. In an extreme case, robustness may be completely irrelevant where fragility is desirable [12].

Security: Watermarking security implies that the watermark should be difficult to remove or alter without damaging the

host image. It is the most important figure of merit for a medical image assuring secrecy and integrity of the watermarked information [13].

Capacity: Watermarking capacity normally refers to the amount of information that can be embedded into the cover image [14].

Watermarks can be applied in the spatial domain or in the frequency domain. Spatial domain techniques, such as the least significant bit method, embed the message by altering the coefficients of the least significant bit of an image. The frequency domain technique embeds the message by modulating the coefficients in the frequency domain, such as in the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) cases. Both domains have their own advantages and disadvantages. Since frequency domain techniques can embed more bits of information and provide ease of compression it is more commonly used. Spatial domain techniques are simple but suffer from the disadvantage that they are not robust enough to overcome several attacks or image alterations.

In this paper the proposed watermarking technique is compared with the simple LSB technique using several image quality metrics that show that the superiority of the proposed algorithm over the latter in terms of authenticity and integrity of DICOM images. Further, the embedded watermark or information logo can be recovered using this algorithm which resembles the hidden image more closely.

2. WATERMARKING USING SIMPLE LSB TECHNIQUE IN SPATIAL DOMAIN

The Least Significant Bit (LSB) algorithm, is very simple, strong, and less perceptible. The embedding of the watermark is performed choosing a subset of image pixels and substituting the LSB of each of the chosen pixels with the watermark bits [15]. This algorithm takes as inputs a cover image and a watermark logo while the output function takes as input the watermarked image and gives the extracted watermark as its output. For a gray scale 8-bit image, we need to read the cover image and information logo and then add the data of the information logo to the least significant bits of each pixel of the cover image, in every 8-bit pixel. The cover image is generally a gray scale image where each pixel is represented by 1 byte. It can represent 256 gray colors between the black which is 0 to the white which is 255. The information logo is generally a binary image and it can be represented by black (0) or white (1). Recovery of the hidden information or watermark is done by extracting the least significant bit of each of the selected image pixels. If the extracted bits match the inserted bits, then the watermark is reconstructed.

Though LSB watermarking schemes have a higher level of invisibility and a less computational overhead, modifications of LSB data is highly sensitive to noise and is easily destroyed. This technique is not resistant enough to image compression and other image processing techniques. Furthermore, image quality may be degraded by the watermark.

This paper describes a new type of biomedical image watermarking technique which is also simple but more effective and efficient than the LSB technique.

3. PROPOSED WATERMARKING TECHNIQUE IN SPATIAL DOMAIN

In the proposed biomedical image watermarking technique, modified bit replacement algorithm in spatial domain is used which is much better than the conventional simple LSB technique. In this scheme, multiple copies of the same information are hidden in several bits of the cover image starting from the lower order to the higher orders. So even if some of the information is lost due to an attack, we can still collect the remaining information and recover the watermark from the cover image using the bit majority algorithm.

3.1. Embedding watermark

The gray scale cover image is divided into several parts according to the size of the information logo and the number of the same information that are to be hidden. Then the lowest pixel value of each sub division is taken as the starting index for the embedding process and this value gradually increases up to the limit equal to the maximum number of information blocks of the watermark. The number of information blocks is the total number of black pixels of the binary logo. This number is also used as a random key which is also sent along with the watermarked image and two other keys which hold the information about the dimensions of the watermark.

3.2. Recovery of watermark

The watermarked image is compared with the original cover image to find the difference in pixel values depending on the three hidden keys. The first key indicates the information block of the information logo and the remaining two indicate the row size and column size of the information block respectively. After getting the three correct keys from the user the different hidden sets of information logo from the watermarked image are found. These different sets are built using the positional information of the hidden pixels and the information obtained from those three keys. The final information logo is then reconstructed by a number of comparisons between these different recovered sets using the proposed bit majority algorithm. This algorithm provides a method to find the closest twin by several comparisons between different sets of data. After recovering the different sets from the attacked watermarked image, the best sets of pixels which are closest to the original information logo, are taken. The rest of the portions of black dots are replaced by white dots. Every set of the recovered logo is checked with one another to find the similarity between the pixels.

4. ATTACKS AND DISTORTIONS

Alterations on the watermarked image made either intentionally or inadvertently, are known as attacks. Different types of attacks can cause image quality degradation. An attack may be performed intentionally on a watermarked document to destroy or degrade the quality of the hidden watermark. These distortions also introduce degradation on the performance of the watermark extraction algorithm [16]. The watermarking technique should be robust enough to survive the attacks so that after extraction the watermark should resemble the original image. Some of the popular attacks mentioned in this paper are (a) *Salt and Pepper Noise*, (b) *Image Compression*, (c) *Gaussian Noise*, (d) *Multiplicative Noise* and (e) *Erosion*.

5. IMAGE QUALITY METRICS

Measurement of the quality of a watermarked image and the recovered information logo is very important for the

watermarking technique to indicate its strength and integrity [17, 18].

To measure the amount of visual quality degradation between the original and watermarked images, different types of image quality metrics such as Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measurement (SSIM), Bit Error Rate (BER), Normalized Absolute Error (NAE), Mean Average Error (MAE), Universal Image Quality Index (UIQI) are used while the quality and similarity of the recovered information logo with the original may be measured by determining the BER, SSIM, Normalized Cross Correlation (NCC), Mutual Information (MI), Structural Content (SC), and UIQI.

Higher value of PSNR, SSIM, NCC, MI and UIQI represents image of good quality while lower values of BER, NAE, MAE, SC represent less error and consequently good quality image

6. RESULTS AND DISCUSSION

A comparative study of the quality measurements between the simple LSB watermarking technique and the proposed modified bit replacement (MBR) watermarking technique has been calculated and the results are presented.

In Table 1, four sets of biomedical cover images along with the information logos and the obtained watermarked images are shown for both the LSB and our proposed embedding technique. The calculated value of the quality metrics such as PSNR, SSIM UIQI, BER, MAE and NAE are given to find the image quality after watermark insertion. It is observed that for both the cases the difference between the watermarked image and the cover image by the Human Visual System appear to be identical. In case of the LSB technique we are hiding a single logo but in case of our proposed algorithm we are hiding 8 sets of information logo. So, the chance of deformity of the original image is more in the proposed case. But from Table 1 we observe that the value of SSIM in all the cases of the proposed algorithm is close to 1 and BER is close to 0 which are proof of the similarity between the original and watermarked cover image. The values of the other quality metrics also indicate that the watermarked images are quite similar to the original cover images without any distortions or visual deformities. The value of universal image quality index (UIQI) is also close to 1 which is proof of the good quality of the watermarked image.

In Table 2 through Table 6, the successful recovery of hidden information from the altered watermarked image is shown under several attacks for both the simple LSB and proposed biomedical watermarking. The results are shown for some biomedical MRI images subjected to different types of attacks such as salt and pepper noise, image compression, Gaussian noise, multiplicative noise, erosion and dilation. The recovered logos are visually more similar to the original one in our proposed MBR technique than the LSB technique. Moreover, from the experimental results we can see our proposed technique can resist higher order of attack and still can recover the hidden information whereas the simple LSB technique fails to recover and construct the hidden information in such circumstances. The values of SSIM and BER of the recovered logo using MBR technique indicate that they are much closer to the embedded one and also much better than LSB technique. Moreover NCC and UIQI also prove the quality of the recovered watermark or information logo. In comparison with the LSB and our proposed MBR watermarking technique BER and SC are less and closer to 0. SSIM, NCC, MI and UIQI are higher and closer to 1.

From Table 2 we can see our proposed watermarking technique can recover the watermark more close to the original one than LSB under 40 % of salt and pepper noise. From Table 3 we can see that LSB technique cannot resist JPEG compression whereas the proposed technique can recover information closer to the hidden one from 5 % JPEG compressed image. Results from Table 4 describe that the proposed algorithm is much stronger to recover hidden information from Gaussian noise attack than the LSB technique. Other tables also show that the proposed MBR biomedical image watermarking technique is quite efficient to recover information from other attacked watermarked image like multiplicative noise and erosion.

The value of different quality metrics mentioned in Table 1 to Table 6 indicates that the proposed modified bit replacement (MBR) biomedical watermarking technique is much better both visually and statistically than the simple and conventional LSB watermarking technique.

7. CONCLUSION

The proposed MBR biomedical image watermarking scheme in the spatial domain includes procedures for data embedding, extraction and verification of quality using several quality metrics for both watermarked image and the recovered watermark. Experimental results show that the proposed modified bit replacement watermarking scheme has high robustness, embedding capacity, low distortion and enhanced security than the LSB technique. Moreover it can also resist several moderately strong attacks. It is also observed that the original information logo can be reconstructed by the proposed bit majority algorithm whose integrity can be strictly verified. A number of image quality metrics support the quality, strength and satisfy the high performance of the proposed algorithm.

8. ACKNOWLEDGMENTS

It is my pleasure to express my heartiest thanks to all the faculty members of Institute of Radio Physics and Electronics, University of Calcutta, Kolkata and the Electronics and Communication Engineering Department of Guru Nanak Institute of Technology, Kolkata for their ungrudging support and cooperation.

9. REFERENCES

- [1] Fazli, S. and Khodaverdi, G, Trade-off between Imperceptibility and Robustness of LSB Watermarking using SSIM Quality Metrics, 2010, IEEE DOI 10.1109/ICMV.2009.68
- [2] Siau-Chuin Liew; Zain, J.M., "Reversible medical image watermarking for tamper detection and recovery", 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), vol. 5, pp. 417-420, 2010.
- [3] Cao, F., Huang, H.K. & Zhou, X.Q., "Medical image security in a HIPAA mandated PACS environment", Computerized Medical Imaging and Graphics, vol. 27, no. 2- 3, pp. 185-196, 2003.
- [4] Kobayashi, L.O.M.; Furuie, S.S.; Barreto, P.S.L.M., "Providing Integrity and Authenticity in DICOM Images: A Novel Approach, IEEE Transactions on Information Technology in Biomedicine," vol. 13, Issue: 4, pp. 582-589, 2009.
- [5] G. Coatrieux, H. Maitre, B. Sankur, Y. Rolland, R. Collorec, "Relevance of Watermarking in Medical Imaging", in Proceedings of the IEEE EMBS Conf. on

- Information Technology Applications in Biomedicine, Arlington, USA, Nov., pp. 250-255, 2000.
- [6] "Digital Image Processing Using MATLAB", 2nd edition, by Gonzalez, Woods, and Eddins, Gatesmark Publishing, ISBN 9780982085400.
- [7] Miller, M.L., Cox, I.J., Linnartz, J.M.G. & Kalker, T., "A Review of Watermarking Principles and Practices" in Digital Signal Processing for Multimedia Systems, eds. K.K. parhi & T. Nishitani, Marcel Dekker Inc., New York, 1999.
- [8] G. Langelaar, I. Setyawan, and R. Legendijk. Watermarking digital image and video data. IEEE Signal Processing Magazine, vol. 17, pp. 20-46, 2000.
- [9] C. Podilchuk and E. Delp. Digital Watermarking Algorithms and Applications. In IEEE Signal Processing Magazine, vol. 18, no. 4, July 2001.
- [10] Zain, J.M.; Fauzi, A.R.M., "Evaluation of Medical Image Watermarking with Tamper Detection and Recovery (AW-TDR)", 29th Annual International Conference of the IEEE on Engineering in Medicine and Biology Society, EMBS 2007. pp. 5661-5664, 2007.
- [11] Velumani, R.; Seenivasagam, V., "A reversible blind medical image watermarking scheme for patient identification, improved telediagnosis and tamper detection with a facial image watermark", IEEE International Conference on Computational Intelligence and Computing Research (ICIC), pp. 1-8, 2010.
- [12] C. Rey and JL. Dugelay. A survey of watermarking algorithms for image authentication. EURASIP Journal on Applied Signal Processing, vol. 6, pp.613–621, 2002.
- [13] H-M. Zhao, C-M. Hsu, and S-G Miao, "A data-hiding technique with authentication, integration, and confidentiality for electronic patient records," IEEE Trans. Information Technology in Biomedicine, vol. 6, pp. 46-53, 2002.
- [14] Tian, J. 2003, "High capacity reversible data embedding and content authentication", IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 517- 520, 2003.
- [15] Lee, G. J., Yoon, E. J. and Yoo, K. Y., "A new LSB based Digital Watermarking Scheme with Random Mapping Function", 2008, IEEE DOI 10.1109/UMC.2008.33
- [16] Huang, H.; Coatrieux, G.; Shu, H.Z.; Luo, L.M.; Roux, C., Medical image integrity control and forensics based on watermarking - Approximating local modifications and identifying global image alterations. "Annual International Conference of the IEEE on Engineering in Medicine and Biology Society, EMBC, pp. 8062-8065, 2011.
- [17] M. Eskicioglu and P. S. Fisher, "Image Quality Measures and Their Performance," *IEEE Transactions on Communications*, vol. 43, no. 12, pp. 2959-2965, December 1995.
- [18] H. R. Sheikh, M. F. Sabir and A. C. Bovik, "A Statistical Evaluation of Recent Full Reference Image Quality assessment Algorithms," IEEE Transactions on image processing, vol. 15, no. 11, pp. 3441-3456, November 2006.

Table 1: Cover Image, watermark or information logo and image quality measurements for both LSB and MBR algorithm



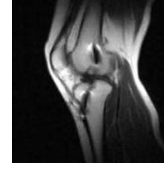
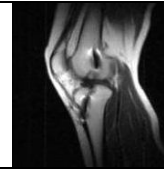
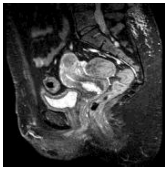

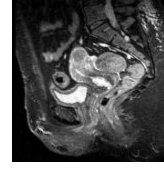
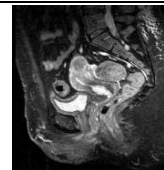




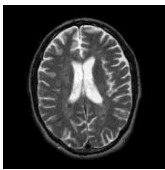



	Cover Image	Information Logo	Watermarking Technique	watermarked image	Image quality measures		
					PSNR	SSIM	UIQI
SET 1		 S Logo	Simple LSB		PSNR	SSIM	UIQI
					51.17	0.99556	0.93536
					BER	MAE	NAE
	Knee Douglass Lucas MRI	No of Information Block 93	Proposed MBR		PSNR	SSIM	UIQI
					42.29	0.97293	0.90468
					BER	MAE	NAE
0.04465	0.33777	0.00666					
SET 2		 E Logo	Simple LSB		PSNR	SSIM	UIQI
					51.1624	0.99757	0.98506
					BER	MAE	NAE
	Pelvis Gynae MRI	No of Information Block 140	Proposed MBR		PSNR	SSIM	UIQI
					40.5349	0.98105	0.96866
					BER	MAE	NAE
0.06819	0.50885	0.00869					
SET 3		 B Logo	Simple LSB		PSNR	SSIM	UIQI
					51.1823	0.99505	0.76853
					BER	MAE	NAE
	Foot Ankle MRI	No of Information Block 130	Proposed MBR		PSNR	SSIM	UIQI
					40.8229	0.96193	0.84927
					BER	MAE	NAE
0.06337	0.47525	0.00753					
SET 4		 KP Logo	Simple LSB		PSNR	SSIM	UIQI
					51.0685	0.99536	0.71935
					BER	MAE	NAE
	Brain MRI (Top View)	No of Information Block 109	Proposed MBR		PSNR	SSIM	UIQI
					41.2924	0.97498	0.86849
					BER	MAE	NAE
0.05563	0.42175	0.00878					

Table 2: Watermarked image, salt and pepper noise attacked watermarked image, recovered logo and image quality measures

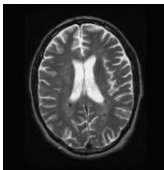
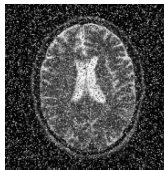


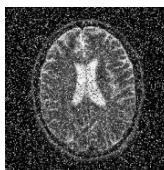


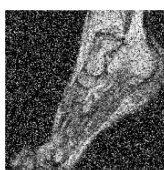




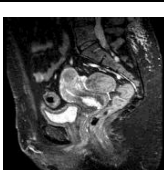
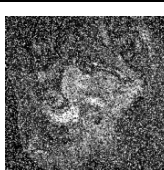


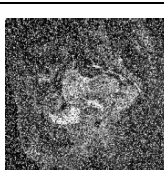

Recovery Technique	Watermarked Image	Amount of attack	Noisy image	PSNR	Original and Recovered logo	Image quality measures		
						BER	SSIM	NCC
Simple LSB		20%		10.99	KP	BER	SSIM	NCC
						0.44531	0.06346	0.5714
						MI	SC	UIQI
						0.00759	1.08889	0.0737
Proposed MBR		20%		10.98	KP	BER	SSIM	NCC
						0.00391	0.99683	1.0000
						MI	SC	UIQI
						0.95027	0.99324	0.9922
Simple LSB		30%		9.24	B	BER	SSIM	NCC
						0.34766	0.22800	0.6428
						MI	SC	UIQI
						0.06792	1.00800	0.2558
Proposed MBR		30%		9.24	B	BER	SSIM	NCC
						0.03906	0.91396	0.9841
						MI	SC	UIQI
						0.77205	0.95455	0.8965
Simple LSB		40%		8.29	E	BER	SSIM	NCC
						1.00000	0.11482	0.7500
						MI	SC	UIQI
						0.09113	0.86740	0.1648
Proposed MBR		40%		8.28	E	BER	SSIM	NCC
						0.14453	0.65472	0.9139
						MI	SC	UIQI
						0.43845	0.88520	0.6596

Table 3: Watermarked image, JPEG Compression attacked watermarked image, recovered logo and image quality measures

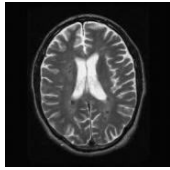
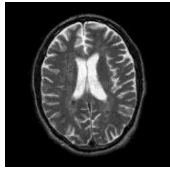






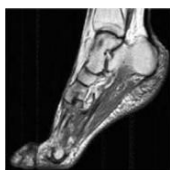















Recovery Technique	Watermarked Image	Amount of attack	Noisy image	PSNR	Original and Recovered logo	Image quality measures for recovered logo		
						BER	SSIM	NCC
Simple LSB		2%		50.43		BER	SSIM	NCC
						0.42578	0.18787	0.6258
						MI	SC	UIQI
						0.01224	1.0068	0.1333
Proposed MBR		2%		54.90		BER	SSIM	NCC
						0.00000	1.00000	1.0000
						MI	SC	UIQI
						0.98405	1.0000	1.0000
Simple LSB		3%		52.01		BER	SSIM	NCC
						0.43359	-0.0208	0.5317
						MI	SC	UIQI
						0.01261	1.05882	0.0377
Proposed MBR		3%		40.46		BER	SSIM	NCC
						0.05469	0.80754	0.9603
						MI	SC	UIQI
						0.69693	0.96923	0.8320
Simple LSB		5%		52.09		BER	SSIM	NCC
						0.36328	0.10754	0.6977
						MI	SC	UIQI
						0.05603	1.04484	0.1705
Proposed MBR		5%		41.92		BER	SSIM	NCC
						0.06250	0.87479	0.9698
						MI	SC	UIQI
						0.66550	0.96399	0.8668

Table 4: Watermarked image, Gaussian Noise attacked watermarked image, recovered logo and image quality measures

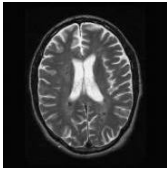
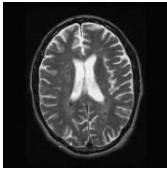


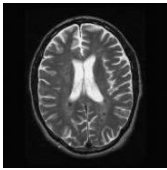
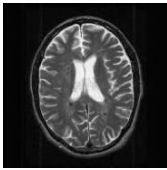


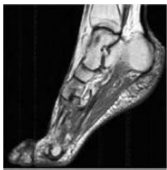
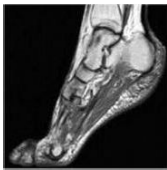






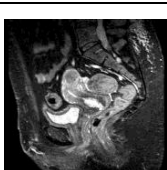



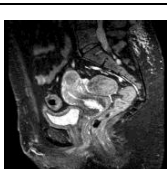



Recovery Technique	Watermarked Image	Noisy image	PSNR	Original and Recovered logo	Image quality measures		
					BER	SSIM	NCC
Simple LSB			46.78		BER	SSIM	NCC
					0.46875	-0.1071	0.5782
					MI	SC	UIQI
					0.00152	1.02797	-0.026
Proposed MBR			49.09		BER	SSIM	NCC
					0.02344	0.95255	0.9863
					MI	SC	UIQI
					0.82424	0.98658	0.9516
Simple LSB			49.15		BER	SSIM	NCC
					0.47656	-0.0432	0.4841
					MI	SC	UIQI
					0.00151	1.0678	-0.013
Proposed MBR			40.18		BER	SSIM	NCC
					0.00391	0.95005	1.0000
					MI	SC	UIQI
					0.96691	0.99213	0.9755
Simple LSB			49.21		BER	SSIM	NCC
					0.50781	0.07570	0.4398
					MI	SC	UIQI
					0.00704	0.99966	-0.019
Proposed MBR			40.02		BER	SSIM	NCC
					0.05469	0.93935	0.9570
					MI	SC	UIQI
					0.72331	0.98272	0.9253

Table 5: Watermarked image, Multiplicative noise attacked watermarked image, recovered logo and image quality measures

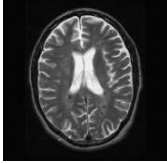
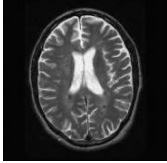


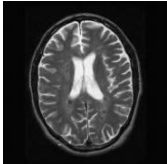
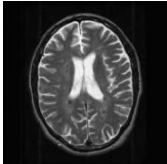






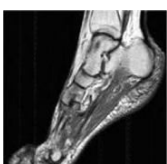
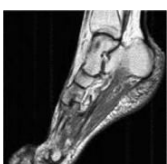



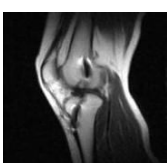






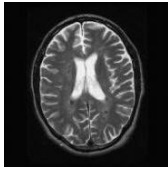
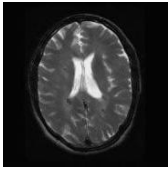


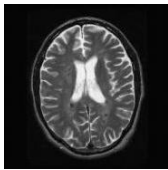
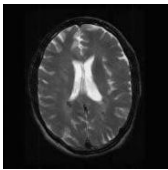


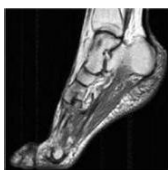
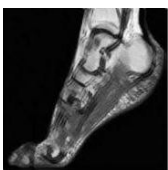



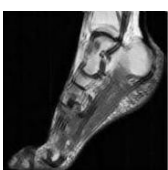



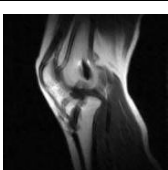






Recovery Technique	Watermarked Image	Noisy image	PSNR	Original and Recovered logo	Image quality measures		
					BER	SSIM	NCC
Simple LSB			43.70	 	0.3984	0.19705	0.6530
					MI	SC	UIQI
					0.02476	1	0.1786
					BER	SSIM	NCC
Proposed MBR			46.55	 	0.01953	0.98918	0.9863
					MI	SC	UIQI
					0.84527	0.99324	0.9702
					BER	SSIM	NCC
Simple LSB			42.33	 	0.30859	0.29201	0.7222
					MI	SC	UIQI
					0.10942	0.93333	0.3486
					BER	SSIM	NCC
Proposed MBR			39.24	 	0.04688	0.90073	0.9682
					MI	SC	UIQI
					0.73051	0.96923	0.8869
					BER	SSIM	NCC
Simple LSB			35.35	 	0.24609	0.66567	0.7964
					MI	SC	UIQI
					0.19081	1.0315	0.5811
					BER	SSIM	NCC
Proposed MBR			34.59	 	0.03516	0.83182	0.9820
					MI	SC	UIQI
					0.78671	0.99356	0.9100
					BER	SSIM	NCC

Table 6: Watermarked image, attacked watermarked image due to Erosion, recovered logo and image quality measures

Recovery Technique	Watermarked Image	Noisy image	PSNR	Original and Recovered logo	Image quality measures		
					BER	SSIM	NCC
Simple LSB			20.53		BER	SSIM	NCC
					0.42188	0.09947	0.4081
					MI	SC	UIQI
					0.03924	1.81481	0.1443
Proposed MBR			20.49		BER	SSIM	NCC
					0.08594	0.71682	0.9659
					MI	SC	UIQI
					0.56838	0.92453	0.8182
Simple LSB			18.49		BER	SSIM	NCC
					0.39453	0.27367	0.5158
					MI	SC	UIQI
					0.03255	1.2000	0.2476
Proposed MBR			18.47		BER	SSIM	NCC
					0.11328	0.81104	0.9285
					MI	SC	UIQI
					0.49918	0.91971	0.7771
Simple LSB			22.39		BER	SSIM	NCC
					0.29688	0.46485	0.7471
					MI	SC	UIQI
					0.12279	1.0448	0.4149
Proposed MBR			22.37		BER	SSIM	NCC
					0.10547	0.67195	0.9450
					MI	SC	UIQI
					0.49788	0.95829	0.7452



Multiple parameter Monitor and Control in Wind Mills

V.Rukkumani D.Angeline Vijula S.Allirani
Sri Ramakrishna Engineering College
Coimbatore, India

Abstract: Generating power through the wind mills is very common and essential now-a-days. In wind mills, designing a proper Condition Monitoring System is a difficult task. The main objective of this paper is a) to monitor the parameters like generator current, temperature, voltage, vibration, turbine speed in wind mills b) to detect fault in the temperature sensor and in the storage unit c) to control the parameters such as speed, temperature, current through LabVIEW. Basically, signal parameters will be detected by sensor units and conditioned with the help of Signal Conditioning Unit (SCU). Generally most of the signals will be analogies, and so it's not convenient to process it further to the PIC Microcontroller. In order to do that, an Analog to Digital Converter (ADC) to be used for digital conversion and the PIC microcontroller has In-Built with ADC. Then, the controller will process those data and sends the appropriate output signal to Personal Computer (PC). Thus the information such as the temperature sensor fault, battery/storage unit fault, fault current etc is received and stored in the PC. The controller is interfaced to the PC through LabVIEW using Ethernet. The measured parameters is displayed through LabVIEW and based upon the data, the faults are rectified. The sequential monitoring of the data is possible through the LabVIEW and any variations that have to be made to maintain the stability of the control system is done with the help of Ethernet at any time. The Ethernet is the advanced communication link that enables us the faster transmission. This paper prevents the replacement of the Microcontroller every time an error occurs as done before and also reduces the total number of Microcontrollers used.

Keywords: Wind mill, LabVIEW, Ethernet, Signal Conditioning Unit, vibration sensor and PIC Microcontroller

1. INTRODUCTION

Recent advances in wind mill technologies have already made enormous contributions in many industrial areas. Automation provides increased efficiency, throughput, accuracy and repeatability. In this era, automation has reached a limit where unplanned automation doesn't provide expected results. In many applications it is difficult and time consuming the process for the entire devices to monitor and control the parameters and to perform the specified task[1],[2]. To overcome this problem we are planned a system that single operator manages the entire system. The concept of the multiple parameter monitoring involves the multiple tasks performed working in coordination with the environment to accomplish a complicated task. When the entire wind mill parameters are monitored as a whole in single PIC Microcontroller, it becomes possible to achieve a complicated task. This device makes optimum utilization of available resources which gives maximum efficiency compared to previous methods used for power generation[4].

This application have been implemented in the wind mills becomes an ideal setup to demonstrate the concept of multiple planning. There are certain situations where overloading, short circuiting, etc creates a problem to the operator and the setup. In those situations, it's apt to have a device that reduces the effort of the operator to modify the parameter when ever needed. By having this device, it adapts to the environment and manages to provide the constant efficient output [5]. Thus this research works aims to reduce the complicated situations and has a constant eye on the parameters and controls it in the unstable situations at less time duration. We propose an approach of neglecting several controllers into a single controller wherein providing the energy efficient at the same time simple and reliable system [6].

2. METHODOLOGY

Total circuit has to be implemented to obtain various parameters in windmills. The generator output current is measured by using the Current Transformer (CT) which is located inside the generator. The generator output voltage is measured by using the Potential Transformer (PT) which is located next to the generator output block. Vibration in the generator is measured by using piezo electric crystal. Piezo electric crystal is kept inside the generator. The output of piezo electric crystal is given to the microcontroller through the signal conditioning unit. The Signal Conditioning Unit (SCU) is used to amplify output signals produced from various sensor blocks (current sensor, voltage sensor, vibration sensor, blade speed sensor). Also it is used to feed the amplified signals to the Analog to Digital Converter (ADC). All the analog signals from various sensor blocks (current sensor, voltage sensor, vibration sensor, blade speed sensor) are converted into digital form. This digital signal is fed into the microcontroller. Fluctuations in the output of the generator are removed by using a capacitor in the smoothing circuit and then it is given to the battery for power storage. Here we are using PIC microcontroller for controlling purpose. It receives all the signals from ADC and based on the inputs given by the user, it controls and monitors all the parameters in the wind turbine. EEPROM is Electrically Erasable Programmable Read Only Memory which is used for store the error data from the microcontroller. This data are read by the main computer through Ethernet after that the information's are erased by the main computer. Driver circuit is used before the control circuit, used to avoid the load

current entering into microcontroller while driving large loads.

A LN2803 driver IC is used. The control circuit consists of relays, temperature controllers, fault current protection. Relays are used for ON-OFF purposes. Temperature controller uses cooling system to control the generator winding temperature. Ethernet is a family of frame-based computer networking technologies for Local Area Networks (LAN). It is used to read the microcontroller's memory directly. It also enables to change the parameters whenever needed. The LM35 is an integrated circuit sensor that can be used to measure temperature with an electrical output proportional to the temperature (in °C). The LM35 generates a higher output voltage than thermocouples and may not require that the output voltage be amplified. All the sensors are fitted inside the generator to monitor all the electrical parameters.

3. PROPOSED METHOD

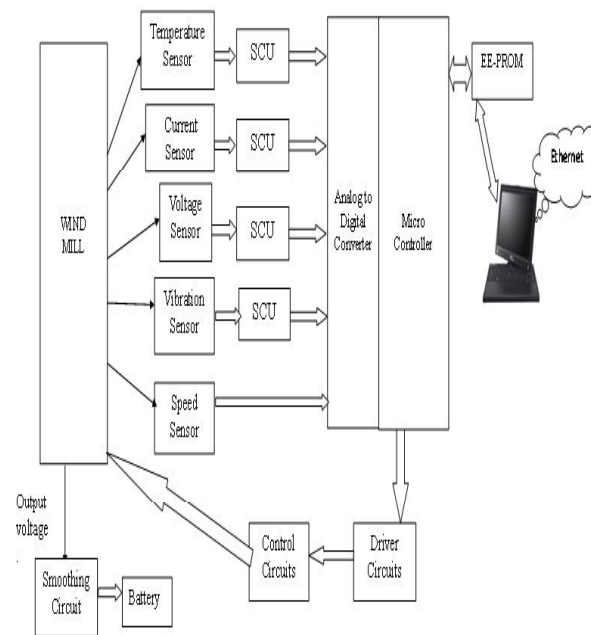


Figure 1. Block Diagram of Multiple Parameter Monitor and Control in Windmills

3.1 Current Sensor

The generator output current is measured by using the Current Transformer (CT) which is located in the generator. A current transformer produces a reduced current accurately proportional to the current in the circuit, which can be conveniently connected to measuring and recording instruments fitted at the output side of the microcontroller.

3.2 Vibration Sensor

When a large amount current/voltage will causes vibrations. The Vibration in the generator is measured by using piezo electric crystal. Piezo electric crystal is kept in the generator. It works on a Piezo-Electric Effect. It produces the electrical output. The output of piezo electric crystal is given to the microcontroller through the signal conditioning unit (SCU).

3.3 Temperature Sensor

The generator winding temperature is measured by LM35 sensor and the output is given to the PIC Microcontroller and also fault detection is carried out using redundant sensor. LM35 is also used as a redundant sensor for temperature measurement.

3.5 Speed Sensor

Turbine consists of large number of blades. The Blade speed is measured by the proximity sensor and it is given to the microcontroller. It is monitored by using LabVIEW.

3.6 Signal Conditioning Unit(SCU)

It is used to amplify output signals produced from various sensor blocks (current sensor, voltage sensor, vibration sensor, blade speed sensor) connected in the generator. It consists of Instrumentation amplifiers circuits. Also it is used to feed the amplified signals to the PIC Microcontroller which contains an in-built ADC.

3.7 Wind Mill

Wind mills are generally called as wind turbines. It consists of turbine connected to the generator with gear box setup. Here we are using synchronous generator to run the turbine. The reason for using synchronous generator is, they are more stable and secure during normal operation, and they do not require an additional DC supply for the excitation circuit. Voltage regulation is possible and provides higher power coefficient and efficiency.

3.8 Smoothing Circuit

Fluctuations in the output of the generator are removed by using the smoothing circuit and then it is used as a battery power storage. The smoothing circuit is a low-pass filter designed to reduce ripples from the direct current obtained from synchronous generator.

3.9 PIC Microcontroller

PIC initially referred to "Peripheral Interface Controller". It is used for controlling purpose. It receives all the signals from the SCU and based on the inputs given by the user, it controls and monitors all the parameters of the wind turbine. PICs are popular with both industrial developers due to their low cost, wide availability, and serial programming (and re-programming with flash memory) capability. It has in-built ADC. It acts as a first level control.

3.10 EEPROM

EEPROM is Electrically Erasable Programmable Read Only Memory which is used to store the error data during the microcontroller operation. It is the part of the Microcontroller. This data are read by the main computer through Ethernet after that the information's are erased by the main computer only limited values stored in EEPROM..

3.11 Ethernet

Ethernet is a family of frame-based computer networking technologies for Local Area Networks (LAN). It is used to

read the microcontroller's memory directly. It also enables to change the parameters whenever needed for turbine operation.

3.12 Driver Circuit

Driver circuit is used before of any control circuit. It is used to avoid the load current entering into microcontroller while driving large loads. LN2803 driver IC is used. A ULN2803 is an Integrated Circuit (IC) chip with a High Voltage/High Current Darlington Transistor Array. It allows you to interface TTL signals with higher voltage/current loads.

3.13 Control Circuit

The control circuit consists of relays, temperature controllers and fault current protection. Relays are used for ON-OFF purposes. Temperature controller used to cool the system and control the generator winding temperature.

3.14 Power Supply

Every unit requires power supply to get enabled. The power supply unit consists of step down transformer, rectifier, filter, voltage regulator. There are many types of power supply. Most are designed to convert high voltage AC mains electricity to a suitable low voltage supply for electronic circuits and other devices. A power supply can be broken into a series of blocks, each of which performs a particular function. Each sensors have a separate power supply sources.

4. IMPLEMENTATION

In the wind mills, the Potential Transformer (PT) and the Current Transformer (CT) is placed to the generator output to measure the voltage and current respectively. The Piezo-electric sensor is placed on the generator to measure the vibrations. The Proximity Sensor is placed near the rotor blade such that the rotations of the blade is measured as speed. The LM35 temperature sensor is placed in the generator to measure the temperature. Then the output of all the sensors except the speed sensor output is given to the respective Signal conditioning Unit and fed to the ports of the PIC Microcontroller and the output from the Microcontroller is given to the driver circuits to control the respective parameters.

5. RESULT

Here the parameters such as voltage, current, speed, temperature and vibration have been measured using the sensors such as potential transformer, current transformer, proximity sensor, LM35 and Piezo-electric sensors respectively placed in the appropriate places to measure all the parameters in wind mill.

Table 1 shows the output measures across all the sensors placed in the wind mill.

6. CONCLUSION

This paper of "multiple parameter monitor and control in wind mills" is very helpful because here we measure and control all the necessary parameters together manually and automatically through LabVIEW. So it enables us to measure and control anytime any parameter. It reduces all the basic needs such as cost, labour power, space, time etc. This feature increases the reliability of the setup with the help of Ethernet. It is possible to manage the setup from anywhere and anyone for a fraction of second. Due to this it is best method

for controlling parameters in windmill. In this research minimum amount of current only measured that cannot be sensed by current transformer, we have to extend a special sensor to measure minimum current also.

Table 1. Measured output parameters

Parameter measured	Name of the sensors Used	range/ type
Voltage	Potential Transformer	230V/12V
Current	Current Transformer	5A/500mA
Speed	Proximity Sensor	NPN
Temperature	LM 35	Up to 300 C

7. ACKNOWLEDGMENTS

We would like to express our deep and unfathomable thanks to our Management of SNR Charitable Trust, Coimbatore, India for providing the Centre of Excellence LabVIEW setup in Sri Ramakrishna Engineering College, Coimbatore to do the proposed work.

8. REFERENCES

- [1] Lehtla, M., Koivastik, L., Moller, T., Kallaste, A., and Rosin, A. (2010) 'Design of renewable micro generation monitoring and control application', in Proc. of the Electric Power Quality and Supply Reliability Conference, Kuressaare., pp.105-110.
- [2] Yanyong Li. (2011) 'Discussion on the principles of wind turbine condition monitoring system', in Proc. of the International Conference on Materials for Renewable Energy and Environment, Vol.1, pp.621 – 624.
- [3] Wang Chuhan. (2011) 'Remote monitoring and diagnosis system for wind turbines based on CAN', in Proc. of the International Conference on Intelligent Computation Technology and Automation, Vol.2, Shenzhen, Guangdong., pp.1152-1155.
- [4] Habash, R.W.Y., Groza, V., Yeu Yang., Blouin, C., and Guillemette, P. (2011) 'Performance testing and control of a small wind energy converter', in Proc. of the sixth IEEE International Symposium on Electronic Design, Test and Application, pp.263 – 268.
- [5] Wenxian Yang, Tavner, P.J., Crabtree, C.J., and Wilkinson, M. (2010) 'Cost-effective condition monitoring for wind turbines', IEEE Transactions on Industrial Electronics, Vol.57, No.1, pp.263-271.

- [6] Yassine Amirat, Choqueuse, Mohamed Benbouzid.(2010) 'Condition monitoring of wind turbines based on amplitude demodulation', IEEE Energy Conversion Congress and Exposition (ECCE), pp. 2417 – 2421.
- [7] Fazli Mehrdad, Talebi Nemat. (2010) 'A new method for uninterrupted operation of wind turbines equipped with DFIGs during grid faults using FCL', in Proc. of the International Conference on Environment and Electrical Engineering , pp.33-36.
- [8] Ramtharan,G., Chris,N., and Ervin,B.(2010) 'Importance of advanced simulations of electrical transients in wind turbines', Proc. in the European Windmill Energy Conference (EWEC) Warsaw, Poland., pp 1-10.
- [9]. Aziz Muthanna,A., Noura, Hassan, and Fardoun Abbas.(2010) 'General review of fault diagnostic in wind turbines',18th Mediterranean Conference on Control & Automation (MED), pp.1302-1307.