JOURNAL
OF
APPLIED
QUANTITATIVE
METHODS

STRUCTURAL MODELING AND ANALYSIS OF INTELLIGENT MOBILE LEARNING ENVIRONMENT: A GRAPH THEORETIC SYSTEM APPROACH

Nitin UPADHYAY[1]
Computer Science & Information Group, BITS-PILANI, Goa Campus, India

E-mail: upadhyay.nitin@gmail.com, nitinu@bits-goa.ac.in

Vishnu Prakash AGARWAL[2]
PhD, Mechanical Engineering Group, BITS-PILANI, Goa Campus, India

Abstract: *This paper presents a new methodology using graph theory and matrix algebra to analyze software architecture based on systems engineering approach. It proposes a set of analytical tool to capture the notion of structural model as the basis to analyze characteristics of software architecture. In the present work, architecture (structure) modeling and analysis of intelligent mobile learning environment (iMLE) are presented that describe characteristics of performance, quality and reliability.*

Key words: *intelligent mobile learning environment; mobile agent; m-learning; agent; intelligent tutoring system; system structure; graph theory; matrix approach; variable permanent function (VPF)*

1. Introduction

Much work has been done during the last two decades in modeling and analyzing software architecture for various characteristics such as- performance, quality, reliability etc. Systems engineering has evolved as a novel approach to model software architectures. It is proposed that the structural/system modeling technique **[Saradhi M., 1992]** acts as a framework through which components, attributes, inter-relationship, and inter-dependencies within and across the system are expressed. It has been shown by researchers that the overall performance of a system depends upon the interaction/interdependence of its systems and subsystems **[Maes et al., 1998; Gray, 1997; Papaionnou and Edwards, 1998; Nick et al., 2000; Maes and Guttman, 1998]**.

The work reported here addresses the fundamental issue of how to analyze software architecture based on systems engineering approach. Our contribution is to propose a mathematical model, which analyzes without loss of generality all the flow, information, control, semantics, static and dynamic behaviors of systems and sub-systems using graph theory, matrix algebra, and permanent function. The present work, deals with the modeling and analysis of intelligent mobile learning environment. The methodology proposed here can work for all other software (system) architecture as well. The methodology is so strong that it can analyze all aspects of ***iMLE*** architecture without losing any information and optimize the characteristics associated with it.

2. Literature survey & related work

The process of learning has undergone revolutionary changes. The system of education has now crossed its geographical and time limit only because of the availability of high bandwidth infrastructure (such as 3G, GPRS and UMTS networks), advances in wireless technologies **[Chen and Nahrstedt, 2000; Chiang, et al., 1998; Johnson and Maltz, 1996; Chen and Lai, 2000; Lin and Liu, 1999]** and acceptance of handheld devices [**Microsoft, 2001**]. Now, e-learning system is moving from first generation to second generation. The integration of artificial intelligence and e-learning is identified as second generation learning or ***ITS*** **[Upadhyay, 2006]**. Integrating mobile computing with e-learning has given rise to new promising field known as mobile learning (m-learning) **[Upadhyay, 2006; Lehner and Nösekabel, 2002]**. In order to improve efficiency and performance of education systems various architectures have been developed and deployed.

In most of the analysis of intelligent m-learning education systems the researchers mainly consider the optimization of the characteristics of education systems from the aspects of autonomous behavior, quality and security. This may be a time bound solution. But in the long term, the performance of other sub-systems will affect the performance of the ***iMLE*** as a whole and hence whatever has been optimized may not be good. Therefore, an appropriate systems approach is best for identifying a permanent solution over the expected life cycle of the ***iMLE.***

The authors are not aware of any study that integrates all the subsystems and system of ***iMLE***. Researchers have identified that the performance of any system is a function of its basic architecture (i.e. layout and design). The understanding of systems architecture and its connectivity and interactions between different systems and down to component level is useful for estimating the contribution of different attributes of the performance of the system. The performance of complete ***iMLE*** (e.g. intelligence, adaptability, quality, availability, reliability) depends upon the performance of its macro level systems and interconnections in an integrated manner. Currently no effective mathematical model is present for studying these aspects in relation with each other or independently.

An attempt is made in this paper to represent the architecture of intelligent mobile learning environment mathematically and a methodology to model complete structure of ***iMLE*** consisting of its macro systems. This is achieved with the help of graph theory, matrix-algebra, and permanent function. This tool has so far been used by various authors to study a sub system for a particular attribute of the performance of a system in thermal power plant **[Mohan et al., 2003]**, nuclear plant **[Sacks et al., 1983],** selection of rolling elements of bearings **[Seghal et al., 2000],** maintainability index **[Gandhi et al., 1991],** but so far it has not been used to model and analyze software architectures with special emphasis to an intelligent mobile learning environment.

For rapid software development, software designers are encouraged to integrate commercial-off-the-shelf (COTS) components in their software systems. Component-based software engineering, in particular, **[Cai et al., 2000; Kozaczynski W., and Booch G., 1998]** has drawn tremendous attention in developing cost-effective and reliable applications to meet short time-to-market requirements. Performance, reliability, quality and other characteristics of software architectures are mostly analyzed and measured only at the time of implementing artifacts. It has been identified by industry and academia that investing in architecture design in the early phase of lifecycle is of paramount importance to a project's success **[Bosch J., 2000; Clements et, al., 2002; Kruchten P., 1995; Shaw M., and Garlan D., 1996]**. The basic structure of the system (software architecture) contributes approximately 30% value to various attributes associated to it. For instance, in order to obtain the value of performance of software architecture following formula can be evaluated:

Performance = (structure, $\boldsymbol{a}_i$)

Where $\boldsymbol{a}_i$ (i = 1,…,n; Load balancing, Priority, Assignment, Scheduling etc.,) are the attributes other than the basic structure of the software architecture which can affect it.

Most of the research is done in optimizing these attributes **[Goel A.L., and Okumoto K., 1979; Jelinski, Z. and Moranda, P. B., 2001; Littlewood, B.A., and Vernall, J.L., 1973; Musa J.D., and Okumoto K., 1984;]**. The optimization of characteristics such as reliability and testing is based on software development and testing rather than on complete software structure is addressed in **[Lyu et. al., 2002]**. Some research point out that software reliability and performance cannot be assessed at the architectural level **[Medvidovic N., and Taylor R., 1998]**. In structural statistical software testing (SSST) model reliability issues are evaluated by considering components independently **[Lyu M. R., 1996; May J. H. R., and Lunn A. D., 1995; May J. H. R., and Lunn A. D., 1995]**. Issues such as reliability, safety, security and availability comprise software dependability **[Littlewood B., and Strigini L., 2000; Randell B., 1995]**. However, there is no standard representation for dependability in model driven architecture thus lacking in complete optimizing of software architecture characteristics. The reliability estimation is also proposed using modular approach **[Woit D., 1997]**. It supposed that software system can be divided into independent components and each component has associated reliability as provided by the vendor. The overall system reliability can be calculated using well know Markov analysis techniques in software system. However the approach does not take into account the reliability of interactions between pair of components.

Our mathematical model preserves all inter-relationships, inter-dependencies, interactions within and across the systems and sub-systems in a single multinomial function. This model also permits us to evaluate various characteristics such as-performance, quality, reliability etc., associated with software architectures. In the previous works researchers were mainly concern about the evaluation of systems/sub-systems (components) and interactions independently. The limitation to these approaches results in not fully optimizing the overall system characteristics as the approaches do not analyze or evaluate all information (components and interactions) together. Our contribution is the major break through in optimizing the overall system characteristics by giving special emphasis to evaluate and analyze structural modeling aspects of software architectures based on system engineering approach. No study deals with the aspect of modeling and analyzing characteristics of software architectures concurrently but our mathematical model does it efficiently.

3. Identification of system

A top-level system is viewed as a combination of various systems and subsystems. The structure of ***iMLE*** is dependent on the elements contained in the boundary and their interconnections. In order to perform complete designing and analysis of ***iMLE***, we also have to consider contributing factors other than the main physical sub-systems and their interconnections. A subsystem is a system in itself.

To define an intelligent mobile learning environment engineering process, an outline of the necessary tools and procedure to support it is required. Initially, system requirement is identified which is broken down for further analysis, generating its own set of requirements. The whole process is repeated containing more detailed view of the system and sub-systems, until the component level is reached. The prime objective of system approach is to facilitate through evaluation and proper accommodation of new concepts and technology in ***iMLE*** design. On the basis of critical review **[Fabiano et al., 2003; Capuano et al., 2000 ; Oana et al**., **2005; Pesty and Webber, 2004; Tang and Wu, 2000]**, different sub-systems are identified which are further combined to produce five generic sub-systems as shown in Figure 1.

1. Intelligent Tutoring System (***ITS***).
2. Multiagent Intelligent System (***MIS***).
3. Mobile Dimension System (***MDS***).
4. Environment and Human Aspect System (***EHAS***).
5. Mobile Agent System (***MoAS***).

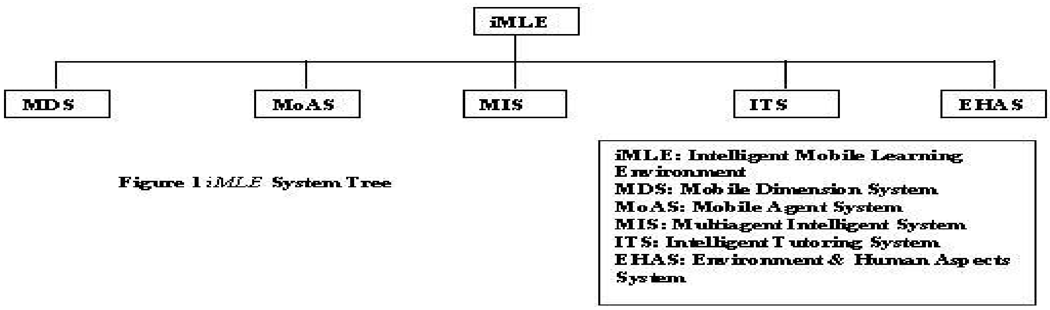

Figure 1 *iMLE* **System Tree**

Mobile learning application industry is free to identify a different set of subsystems as per its requirements, aims and objectives. Interaction and interdependency of various subsystems from the point of view of business, researches, maintenance etc. is the basis to understand the function and performance of ***iMLE***.

Figure 1 does not show interactions between sub-systems. In real application interactions are present among these sub-systems. An attempt is made to identify different types of interactions/interdependencies or information flow between these sub-systems under different situations. For better understanding the system tree diagram, Figure 1 is modified to include all the interactions and is shown as block diagram, Figure 2.

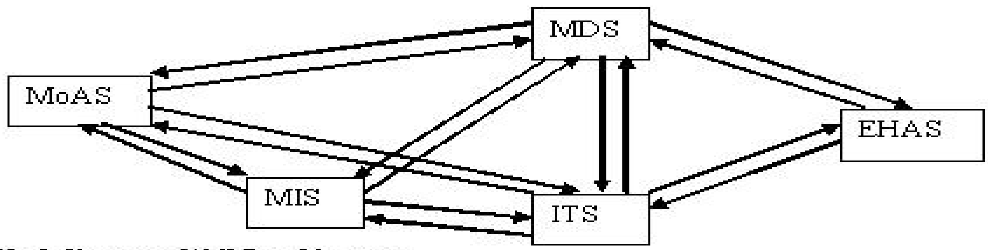

Figure 2 Block diagram of *iMLE* architecture.

3.1. Composition of subsystems and sub-subsystems of imle

The following sub-sub systems of ***iMLE*** sub-systems are proposed:

3.1.1. iMLE Subsystem - Intelligent Tutoring System (*ITS*):

Three major subsystems characterize the ***ITS*** **[Upadhyay, 2006],** Figure 3 - the **Student Model**, the **Domain Model** and the **Pedagogical Module**. A new subsystem **Education Model** adds functionality for the teacher.

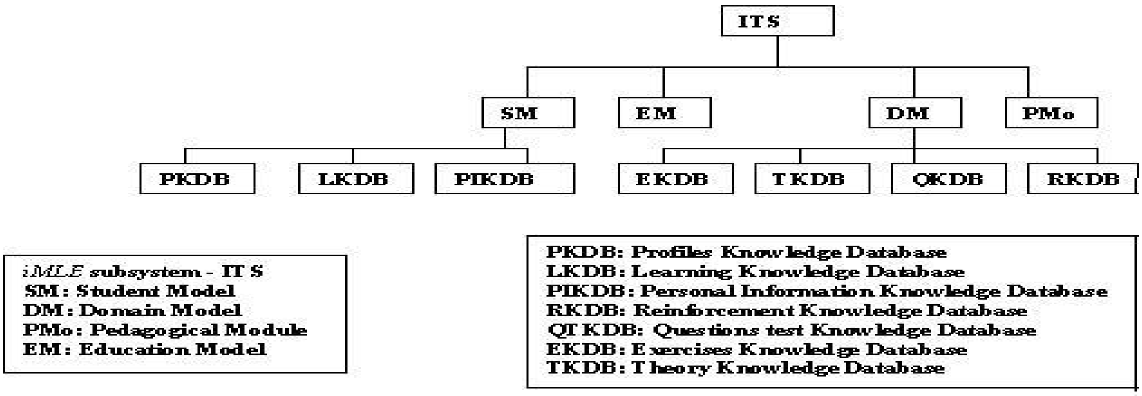

Figure 3 *subsystems* of Intelligent Tutoring System

Student Model

In Student Model, knowledge about the students is maintained, which is obtained by means of their profile and interaction with the system. It consists of three subsystems - knowledge databases (KDBs):

Personal Information KDB: It maintains personal identification, which allows access control to learning perspective through system.

Profile KDB: It manages student level (beginner, intermediate and advanced) and presentation styles (font style, color, size, background etc.)

Learning KDB: It maintains information about students learning history such as page visited, scrolls performed, number of hits/clicks, exercise and tests attempted so far etc.

Domain Model

In Domain Model, knowledge about the contents to be taught is stored. It consists of four subsystems i.e. knowledge databases:

Content KDB: It manages content pages to be used for teaching purpose.

Test KDB: It maintains test questionnaire on the specific contents for different levels.

Exercise KDB: It maintains exercises on the specific contents for different levels.

Reinforcement KDB: It maintains information to be shown to students for better learning. This is done by analyzing information from pedagogical module.

Pedagogical Module

In Pedagogical module, critical analysis is done for effectively presenting the subject matter to the student. It performs three main tasks:

- It provides learning guidelines.
- It updates domain model statistics.
- It keeps record of reinforcement information in learning KDB.

Education Model

In Education Model, functions necessary for teaching are managed. Using this model teacher can change the contents of the subject matter on the basis of information obtained from the Student Model and Domain Model. For effective teaching, teacher can change preferences (presentation styles, color, background etc.), give reinforcement to students, obtain statistics and consult the subject matter.

3.1.2. iMLE Subsystem – Multiagent Intelligent System (MIS)

The MIS comprises four subsystems, Figure 4, as follows:

Exercise agent: The exercise agent looks after the exercise that a student has to deal with depending upon student level of understanding and the content that student has covered. The exercise agent by its own means (pro-active) also provides link to the subject content pages relevant to the proposed exercises.

Preference agent: The preference agent is responsible for maintaining the student (user) choice state of interaction as compatible with MUI.

Account agent: The accounting agent perceives the interaction between user and the MUI when the student accesses content page. This agent keeps track of the scroll and time spent on each page of content. When the student shifts to some other content then account agent stores all parameters in learning KDB.

Test Agent: The test agent is responsible for proposing test as per student level. The test agent by its own means (pro-active) works for the designing of test for the particular topic/content. The test is shown to the student in the form of questionnaires. The test and exercise agent both work synchronously.

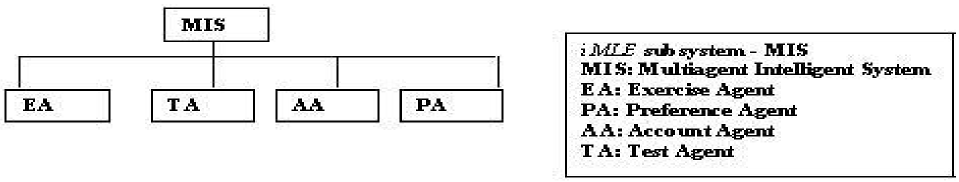

Figure 4 subsystems of Multiagent Intelligent System

3.1.3. iMLE Subsystem – Mobile Dimension System (MDS)

The critical subsystems of MDS, Figure 5, are:

Multimodal User Interface (MUI)

For desktop/PC applications, use of keyboard, mouse and monitor have been widely accepted. But mobile application needs additional mode of interaction such as voice user interfaces, smaller displays, stylus and other pointing devices, touch screen displays, and miniature keyboards.

Platform (PF)

The scalability issue in mobile devices leads to manufacturing of small size mobile devices. Theses devices are composed of less hardware in comparison to PC/Desktops. It is advisable to write program/application for different compatible platforms only if not needed for specific one for some performance reasons.

Device Capability (DC)

The physical size limitation imposes boundaries on volatile storage, non-volatile storage, and CPU on mobile devices. Storage and processing issues are largely addressed by the various operating systems and platforms on the mobile devices. Limited power supply results by putting constraints on limited size and usage on batteries instead of AC power supply.

Active Behavior (AB)

The two main subsystems of active behavior are:

Synchronous system: These behaviors are time dependent transactions. Here transaction is used in data storage and other systems to indicate boundaries for roll-back and committing of a series of actions that must be executed successfully,` in some predefined manner, for the completion of transactions.

Asynchronous system: These behaviors are time independent transactions.

Wireless Environment (WE)

Whether wired or wireless connectivity is used, mobility means loss of reliability in network connectivity. In the case of wireless network connectivity, physical conditions can significantly affect the quality of service (QoS). For example bad weather, solar flares, and a variety of other climate-related conditions can negate QoS.

Context Awareness (CA)

Context awareness consists of various subsystems as follows:

Location awareness: It deals with the sensing of desired location service in mobile applications.

Environment awareness: Collects information related to environmental conditions and variations such as humidity scale etc.

Situation awareness: It is responsible for specific situation such as light condition, sound and orientation of display.

User recognition awareness: It manages the automatic identification of user login.

Personalization awareness: It perceives the personal information of the user for example font style and color, theme, background color and avatar.

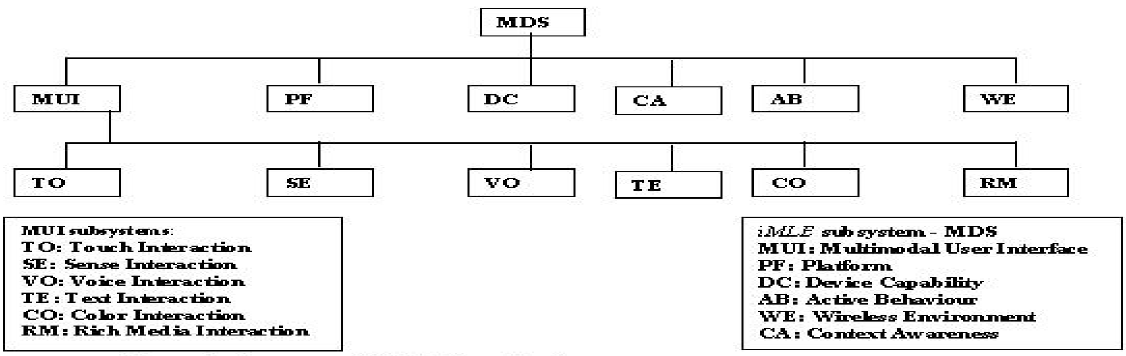

Figure 5 subsystems of Mobile Dimension System

3.1.4. iMLE Subsystem - Environment & Human Aspects System (EHAS)

Two main subsystems have been identified, Figure 6, as **EC**: Environment Condition and **HH**: Human Handling systems. Both theses subsystems affect the functionality of ***iMLE*** architecture. To ensure quality, performance and reliability these two subsystems have to be perfectly sound and fixed from all internal and external disturbances.

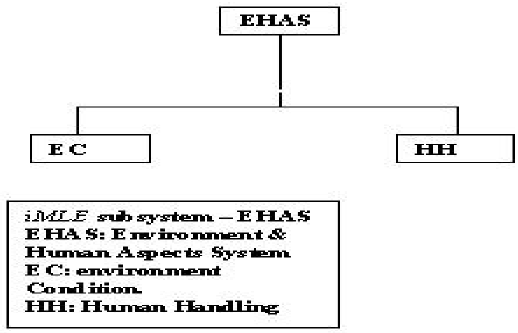

Figure 6 *sub systems* of Environment & Human Aspects System

3.1.5. iMLE Subsystem - Mobile Agent System (MoAS)

Two types of agents as shown in Figure 7: **agent wrappers** and **mediation agents.** They are the subsystems identified **[Bee-Gent and Plangent, 2003]** for **MoAS.** The functionality of both is as follows:

Agent Wrappers: They are used to incorporate agents on existing application. Wrapper agents manage the states of applications. They are also responsible for invoking the application as required.

Mediation Agents: They are responsible for all sorts of inter-application communication among applications. They migrate from an application site to another where they interact with remote agent wrappers.

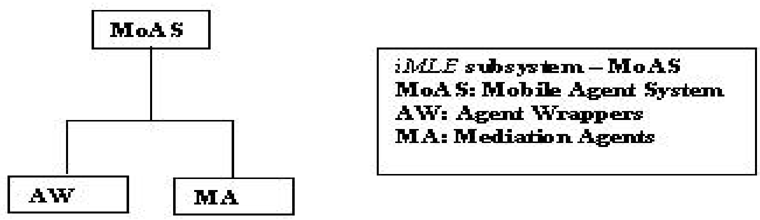

Figure 7 *sub systems* of Mobile Agent System

4. Hierarchical tree structure of iMLE

To compute overall designing and analysis of ***iMLE*** system, a "top-down" approach is used. In this, systems, sub-systems, sub-sub-systems etc are identified up to the component level. This tree structure allows all the parts to be designed from components level to the system level in the hierarchical order by using "bottom-up" approach. This helps to ensure design and geometric compatibility in the system. In general, the hierarchical tree structure may have (n+1) levels as given below:

Level – 0: Complete ***iMLE*** (system)
Level – 1: Sub systems (s-systems)
Level – 2: Sub sub systems (ss-systems)
Level – 3: Sub sub sub systems (sss-systems)
|
|
Level – n: Component level (Component)
A four level tree structure of typical ***iMLE*** system and ITS sub system is proposed in Figure 3 as:
Level – 0: Complete ***iMLE*** (system)
Level – 1: ***ITS, MoAS, MIS*** etc., sub systems (s-system)
As an example for ***ITS*** subsystem (Figure 3)
Level – 2: ***Student Model, Domain Model*** etc., sub sub systems (ss-system)
Level – 3: ***Profile KDB, Learning KDB*** etc., is components' level (Components)
Similarly level 2 and 3 are developed for remaining four subsystems as shown in Figure 4-7.
The hierarchical trees of ***iMLE*** structure may differ depending upon the choices of the distinct systems up to the component level. Identification of tree structure helps in full understanding of ***IMLE*** system engineering process and acts as an asset in improving efficiency, quality, maintainability and reliability of the system.

5. Graph theoretic modelling of system architecture

A system graph $\mathbf{G_s} = [\mathbf{S, E}]$ is used to model system architecture by applying graph theory using linear graph. Let each of the five systems of ***iMLE*** be represented by $\mathbf{S_i}$ (i=1,…,5) and interconnections between them ($\mathbf{S_i}$, $\mathbf{S_j}$) as edge set **E** by edges $\mathbf{e_{ij}}$ (i,j = 1,…,5) connecting the two vertices $\mathbf{S_i}$ and $\mathbf{S_j}$. The graph theoretic representation [**S, E**] of vertex and edge sets of the five-system of ***iMLE*** is called the ***iMLE*** system structure graph. Various types of edges and weights can differentiate the type of connections and interconnections. The undirected edges show the connectivity between (sub) systems or components and the directed edges represent the flow of information or interaction.

The system structure graph (**SSG**) of ***iMLE*** is shown in Figure 8. The five nodes represent respective systems of ***iMLE*** and edges corresponding to the connections/interactions between the subsystems. Connectivity, interdependence and interactions between systems are shown by undirected, directed and dashed edges is shown. If the two systems are interdependent on each other then the relation is shown by opposite arrow edge. If one system is influencing the other then directed edge characterizes this influence. Physical connectivity is represented simply by undirected edge. Dashed edge represents weak or indirect connection.

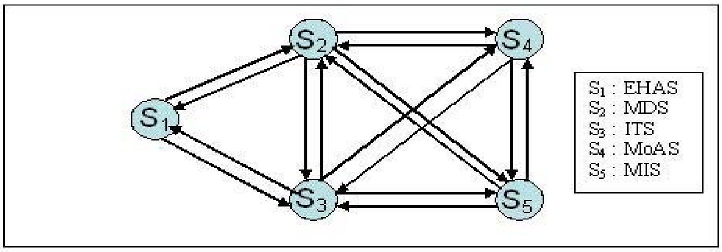

Fig. 8(a). Directed Graph

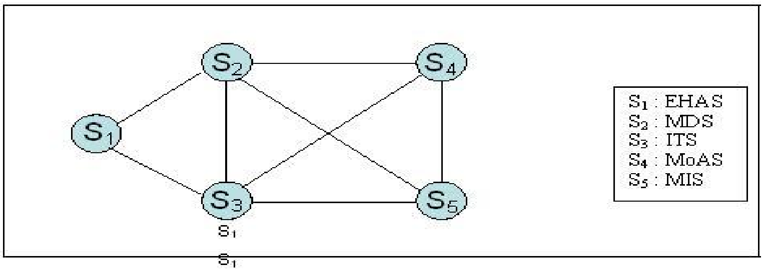

Fig. 8(b). Undirected Graph

Figure 8. System Structure Graph of ***iMLE***

These systems of the ***iMLE*** are also connected physically or indirectly at the level of their sub-systems. Graph theoretic architecture model is used to represent direct, undirected or hybrid interactions between subsystems. A real life ***iMLE*** is represented graphically by directed graph Figure 8a and undirected graph Figure 8b. The connectivity may be directed or undirected depending upon the structural, functional or performance considerations. The ***iMLE SSG*** is capable of updating, modifying and deleting of systems or sub-systems based on different design aspects as per real life situation. The proposed ***SSG*** representation is suitable for understanding and visual analysis, but not appropriate for computer processing. If the number of systems is more, then the overall system becomes more complex for understanding and visual analysis. Moreover, changing of labels of vertices/systems results into new ***SSG***. In view of this, we present computer efficient representation. Many matrix representations are available in the literature **[Deo, 2004; Upadhyay, 2004],** for example, adjacency and incidence matrices. The adjacency matrix is a square matrix and used for this purpose. Using this ***iMLE*** is represented in matrix form.

6. Matrix models

The adjacency matrices of the ***SSG*** are defined to find out which matrix is more suitable to represent ***iMLE***. The matrix should be flexible enough to incorporate the structural information of subsystems and interconnections between them.

6.1. System structure matrix [adjacency matrix] (VAM- iMLE) of iMLE

An incidence matrix can be used to understand the number of connections and how these connect the sub systems. As the resultant matrix is non-square matrix, its further use for system analysis or its derivatives is not very useful. An alternative to incidence matrix, adjacency matrix representation is used to show the connectivity and graph representation. The adjacency matrix **[Deo, 2004; Jurkat and Ryser, 1996]** of a graph **G** = [**V, E**] with '**n**' nodes is an '**n**' order symmetric binary (**0, 1**) square matrix, and e_{ij} representing the connectivity between systems i and j such that:

$\mathbf{e_{ij}}$ = 1, if the sub system '**i**' is connected/interacted to the sub system '**j**' and

= 0, otherwise.

However, $\mathbf{e_{ii}}$ = 0, as subsystem is not connected to itself. In a case where it is connected to itself $\mathbf{e_{ii}}$ = 1. This implies a self-loop at node '**i**' in the graph.

In the (0, 1) adjacency matrix each row and column of the system structure matrix corresponds to a subsystem. The off-diagonal elements $\mathbf{e_{ij}}$ in the matrix represent connection between systems i and j. In this matrix, $\mathbf{e_{ij} = e_{ji} = 1}$ as only connections between systems are considered. The adjacency matrix for a graph as shown in Figure 8 (b) is given below as:

$$A = \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \text{ Subsystems} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array} \qquad (1)$$

The (**0, 1**) adjacency matrix does not contain properties/attributes characterizing different interactions/ connections between different subsystems. It only represents the system connectivity. As the matrix is a square matrix its evaluation is possible. In order to get information about the structural characteristics of the system, we associate variable with the elements of adjacency matrix.

In order to show connectivity/interconnection/interdependence between different systems '**i**' and '**j**' of the ***iMLE***, let off-diagonal elements be represented by a symbol $\mathbf{e_{ij}}$ whose function will depend upon type of connection/interconnection. Adjacency matrix **A** = [$\mathbf{a_{ij}}$] will be (**0,** $\mathbf{e_{ij}}$) instead of (**0, 1**) matrix. The $\mathbf{e_{ij}}$ also provides information about the flow from one subsystem to the other. Variable adjacency matrix (VAM- ***iMLE***) of the system shown in Fifure 8 is proposed below assuming $\mathbf{e_{ij} = e_{ji}}$ as:

$$V_A = \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \text{ Subsystems} \\ \begin{bmatrix} 0 & e_{12} & e_{13} & 0 & 0 \\ e_{12} & 0 & e_{23} & e_{24} & e_{25} \\ e_{13} & e_{23} & 0 & e_{34} & e_{35} \\ 0 & e_{24} & e_{34} & 0 & e_{45} \\ 0 & e_{25} & e_{35} & e_{45} & 0 \end{bmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array} \qquad (2)$$

The $\mathbf{e_{ij}}$, of (VAM- ***iMLE***) apart from representing connectivity also represents influence of structural performance characteristics of **i**$^{\text{th}}$ subsystem on **j**$^{\text{th}}$ sub system, change of **i**$^{\text{th}}$ subsystem affecting the structural performance of **j**$^{\text{th}}$ sub system etc., according to the particular analysis of ***iMLE***. Hence, this is the complete representation of interconnection/interdependence of ***iMLE***. As this matrix also does not infer anything about the characteristic features of the systems a new matrix called '***characteristic system structure matrix***' is defined.

6.2. Characteristic system structure matrix (CSSM- iMLE)

By defining characteristic system structure matrix '**C**', realization of the presence of different systems (based upon system structure) can be done. The ***iMLE*** characteristic system structure matrix (CSSM- ***iMLE***) corresponding to the systems graph in Figure 8 is given below:

C = {**SI – A**}

$$\mathbf{C} = \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \quad \text{Subsystems} \\ \begin{bmatrix} S & -1 & -1 & 0 & 0 \\ -1 & S & -1 & -1 & -1 \\ -1 & -1 & S & -1 & -1 \\ 0 & -1 & -1 & S & -1 \\ 0 & -1 & -1 & -1 & S \end{bmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array} \qquad (3)$$

Where ***I*** is the identity matrix and ***S*** is used as a variable to represent systems characteristic features of the basic structure. This matrix is similar to the characteristic matrix defined in graph theory **[Deo, 2004]**. The characteristic of a system can be reliability, security, availability etc. It can be inferred from the matrix that, it is capable of representing the presence of systems and interconnection between them. It does not include information about the attributes of the connections among subsystems. The determinant of CSSM- ***iMLE*** is called characteristic system structure polynomial (**CP- s**). The **CP-s** of the matrix is shown below:

Det (C) = $\mathbf{S^5 - 8S^3 - 10S^2 - S + 2}$

The **CP-s** of the matrix is invariant of the system **[Deo, 2004]** as it does not change by modifying labeling of systems (vertices) and is the characteristic of the systems structure. It can be inferred that CSSM- ***iMLE*** is not an invariant of system, as new matrix can be obtained by changing labels of systems. Also, diagonal elements show that identical systems are present in the basic structure. This is one of the reasons that make the **CP-s** of CSSM- ***iMLE*** non-unique and incomplete representation of any real system. It has been identified in literature that many graphs belong to the same family known as co-spectral graphs on the basis of having same **CP-s**. To present distinct information of different systems and interconnections between them, a matrix called a ***variable characteristic system structure matrix*** (VCSSM- ***iMLE***) is proposed.

6.3. Variable characteristic system structure matrix (VCSSM- iMLE)

A variable characteristic system structure matrix $\mathbf{V_c}$ is defined by taking into consideration distinct characteristics of subsystems and their interconnections defined by ***SSG***. Let the off-diagonal elements matrix **F** consists of $\mathbf{e_{ij}}$ rather than **1** to represent interaction/connectivity (system '**i**' is connected to system '**j**') and also $\mathbf{e_{ij} = e_{ji}}$. Let us also define diagonal matrix **D** with its variable diagonal elements $\mathbf{S_i}$ (i = 1, 2…., 5) representing the characteristic structure features of five distinct systems.

The VCSSM- ***iMLE*** $\mathbf{V_c}$ = [**D – F**] is written as:

$$\mathbf{V_c} = \begin{array}{c} \\ \\ \end{array}\begin{array}{l} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \quad \text{Subsystems} \\ \begin{bmatrix} S_1 & -e_{12} & -e_{13} & 0 & 0 \\ -e_{12} & S_2 & -e_{23} & -e_{24} & -e_{25} \\ -e_{13} & -e_{23} & S_3 & -e_{34} & -e_{35} \\ 0 & -e_{24} & -e_{34} & S_4 & -e_{45} \\ 0 & -e_{25} & -e_{35} & -e_{45} & S_5 \end{bmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array} \tag{4}$$

The determinant of this (VCSSM- ***iMLE***) is known as variable characteristic multinomial and is written as VCM- ***iMLE***, the variable characteristic multinomial of the ***iMLE***.

Det($\mathbf{V_c}$) =

$$\begin{aligned} & S_1S_2S_3S_4S_5 - e_{12}^2S_3S_4S_5 - e_{13}^2S_2S_4S_5 - e_{23}^2S_1S_4S_5 - e_{24}^2S_1S_3S_5 - e_{25}^2S_1S_3S_4 - e_{34}^2S_1S_2S_5 - e_{35}^2S_1S_2S_4 \\ & -e_{45}^2S_1S_2S_3 - 2e_{12}e_{13}e_{23}S_4S_5 - 2e_{23}e_{24}e_{34}S_1S_5 - 2e_{23}e_{25}e_{35}S_1S_4 - 2e_{24}e_{25}e_{45}S_1S_3 - 2e_{34}e_{35}e_{45}S_1S_2 \\ & -2e_{23}e_{34}e_{25}e_{45}S_1 - 2e_{23}e_{35}e_{24}e_{45}S_1 - 2e_{24}e_{35}e_{25}e_{34}S_1 - 2e_{12}e_{24}e_{13}e_{34}S_5 - 2e_{12}e_{25}e_{13}e_{35}S_4 + e_{13}^2e_{45}^2S_2 \\ & +e_{24}^2e_{35}^2S_1 + e_{25}^2e_{13}^2S_4 + e_{24}^2e_{13}^2S_5 + e_{23}^2e_{45}^2S_1 + e_{25}^2e_{34}^2S_1 + e_{12}^2e_{45}^2S_3 + e_{12}^2e_{34}^2S_5 + e_{12}^2e_{35}^2S_4 + 2e_{12}^2e_{34}e_{35}e_{45} \\ & +2e_{45}^2e_{12}e_{23}e_{13} + 2e_{13}^2e_{24}e_{25}e_{45} - 2e_{12}e_{24}e_{13}e_{35}e_{45} - 2e_{12}e_{25}e_{13}e_{34}e_{45} \end{aligned} \tag{5}$$

The VCM- ***iMLE*** multinomial contains terms both of positive and negative signs. It is the comprehensive tool for analysis in symbolic form. While calculating VCM- ***iMLE*** value for ***iMLE*** analysis, some information about system, sub-systems, components and their connectivity is lost. This is due to the cancellation of some terms and subtraction operation in the process of computing VCM- ***iMLE***. In order to avoid loss of information during structural analysis and structural performance evaluation in critical cases, we propose a new matrix function, which will retain all the multinomial terms with no subtraction operation and hence preserve information about the system, sub systems, components and their interconnectivities, i.e. permanent/permanent function of matrix **[Mohan et al., 2003; Luo and Huang, 2005]**.

6.4. Variable permanent system structure matrix (VPSSM- *iMLE*)

In order to describe proper characterization of ***iMLE*** systems as derived from combinatorial considerations, a permanent matrix **P**, is proposed. The matrix function/permanent **Per(P)** of **VPSSM- *iMLE*** is capable of describing whole ***iMLE*** system i.e. system graph in a single multinomial equation **[Jurkat and Ryser, 1996]**. Let the complete permanent matrix of five-subsystem ***iMLE*** system with all possible interactions present be defined as

$$\mathbf{P} = \begin{array}{l} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \quad \text{Subsystems} \\ \begin{bmatrix} S_1 & e_{12} & e_{13} & e_{14} & e_{15} \\ e_{12} & S_2 & e_{23} & e_{24} & e_{25} \\ e_{13} & e_{23} & S_3 & e_{34} & e_{35} \\ e_{14} & e_{24} & e_{34} & S_4 & e_{45} \\ e_{15} & e_{25} & e_{35} & e_{45} & S_5 \end{bmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \end{array} \tag{6}$$

A variable permanent system structure matrix (VPSSM- ***iMLE***) '$\mathbf{V_p}$' of **SSG** with $\mathbf{e_{ij}} = \mathbf{e_{ji}}$ in Figure 8(b) is written as:

$\mathbf{V_p} = \{\mathbf{D} + \mathbf{F}\}$

$$
\begin{array}{cccccl}
1 & 2 & 3 & 4 & 5 & \text{Subsystems} \\
\end{array}
$$

$$
\mathrm{V_p} = \begin{array}{c} \left[\begin{array}{ccccc}
S_1 & e_{12} & e_{13} & 0 & 0 \\
e_{12} & S_2 & e_{23} & e_{24} & e_{25} \\
e_{13} & e_{23} & S_3 & e_{34} & e_{35} \\
0 & e_{24} & e_{34} & S_4 & e_{45} \\
0 & e_{25} & e_{35} & e_{45} & S_5
\end{array}\right] \end{array}
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\tag{7}
$$

It is a complete representation of ***iMLE***, as it does not contain any negative sign. This means that it preserves all the structural information about dyads, loops of systems, or system attributes such as reliability, availability etc even in numerical form. The only difference between VCSSM- ***iMLE*** and VPSSM- ***iMLE*** is in the signs of off-diagonal elements. The VPF- ***iMLE*** for matrix is written as:

Per($\mathbf{V_p}$) =

$$
\begin{aligned}
&S_1S_2S_3S_4S_5 + [e_{12}^2S_3S_4S_5 + e_{13}^2S_2S_4S_5 + e_{23}^2S_1S_4S_5 + e_{24}^2S_1S_3S_5 + e_{25}^2S_1S_3S_4 + e_{34}^2S_1S_2S_5 + e_{35}^2S_1S_2S_4 + e_{45}^2S_1S_2S_3] \\
&+[2e_{12}e_{23}e_{31}S_4S_5 + 2e_{23}e_{34}e_{42}S_1S_5 + 2e_{23}e_{35}e_{52}S_1S_4 + 2e_{24}e_{45}e_{52}S_1S_3 + 2e_{34}e_{45}e_{53}S_1S_2] \\
&+\{[2e_{23}e_{34}e_{45}e_{52}S_1 + 2e_{23}e_{35}e_{54}e_{42}S_1 + 2e_{24}e_{43}e_{35}e_{52}S_1 + 2e_{12}e_{24}e_{43}e_{31}S_5 + 2e_{12}e_{25}e_{53}e_{31}S_4] + [e_{13}^2e_{45}^2S_2 + e_{24}^2e_{35}^2S_1 \\
&+e_{25}^2e_{13}^2S_4 + e_{24}^2e_{13}^2S_5 + e_{23}^2e_{45}^2S_1 + e_{25}^2e_{34}^2S_1 + e_{12}^2e_{45}^2S_3 + e_{12}^2e_{34}^2S_5 + e_{12}^2e_{35}^2S_4]\} \\
&+\{[2e_{12}^2e_{34}e_{45}e_{53} + 2e_{45}^2e_{12}e_{23}e_{31} + 2e_{13}^2e_{24}e_{45}e_{52}] + [2e_{12}e_{24}e_{45}e_{53}e_{31} + 2e_{12}e_{25}e_{54}e_{43}e_{31}]\}
\end{aligned}
\tag{8}
$$

It can be inferred that the terms present in VCM- ***iMLE*** and VPF- ***iMLE*** are the same but they differ in the signs. In VCM- ***iMLE*** terms consist of both positive and negative sign. But VPF- ***iMLE*** only contains terms of positive sign.

The above equation (multinomial) uniquely represents the ***iMLE*** of Figure 2 irrespective of labeling of subsystems. Every term of these equations represents a subset of the ***iMLE*** system. It is possible to write these equations simply by visual inspection of the ***iMLE*** system of Figure 8 as every term corresponds to a physical subsystem of thee complete system. To achieve this objective, the permanent function of Equation (8) is written in a standard form as (N + 1) groups. All these distinct combinations of subsystems and interactions of the macro system are shown graphically in Figure 9. The multinomial, i.e., the permanent function when written in (N + 1) groups, presents an exhaustive way of analysis of ***iMLE*** at different levels. It helps in identifying different critical components and links to improve reliability, fault tolerance, performance, quality, security, autonomy and availability of system.

On critical analysis of permanent function (8) it is inferred that this multinomial contains only distinct subsystems – $\mathbf{S_i}$, dyads - e_{ij}^2 and loops – $\mathbf{e_{ij}}\ \mathbf{e_{jk}}$ …… $\mathbf{e_{ni}}$. A complete permanent function has been written in a systematic manner for unambiguous and unique interpretation. In short it can be represented as:

Per ($\mathbf{V_p}$) = g ($\mathbf{S_i}$, e_{ij}^2, $\mathbf{e_{ij}}\ \mathbf{e_{jk}}\ \mathbf{e_{ki}}$ etc) { if $\mathbf{e_{ij}} = \mathbf{e_{ji}}$}

= g (Vertices, dyads, loops)
= g (structural components)

Per (**V_P**) = g′ ($S_{i,}$, $e_{ij}e_{ji}$, e_{ij} e_{jk} e_{kl} e_{li} , e_{ij} e_{jk} e_{kl} e_{lm} e_{mi}) { if $e_{ij} \neq e_{ji}$}
= g′ (Vertices, 2-vertex loops, loops)
= g′ (structural components)

The terms of the permanent function **Per (V_P)** are arranged in (n+ 1) groups in the decreasing order of number of vertices/sub-systems **S_i** present in each term. The first group contains terms with (n – 1) **S_i’s.** Second group will contain terms with (n – 2) **S_i’s** and remaining as dyad e_{ij}^2 or **$e_{ij}e_{ji}$** and so on. The last group does not contain any **S_i** in its terms. It contains only terms such as e_{ij}^2 , **e_{ij} e_{jk} e_{ki}** , etc.

Group 1: The first term (grouping) represents a set of **N** unconnected ***iMLE*** subsystems, i.e., **S_1**, **S_2**, …, **S_n**.

Group 2: Group is absent as a particular subsystem has no interaction with itself (absence of self-loops) i.e. any of the subsystem ***MDS, MoAS, ITS, MIS* or *EHAS*** is not connecting itself.

Group 3: Each term of the third grouping represents a set of two-element ***iMLE*** system loops (i.e.,**S_{ij} S_{ji}**)and is the resultant ***iMLE*** system dependence of characteristics **i** and **j** and the ***iMLE*** system measure of the remaining (**N-2**) unconnected elements/subsystems. Group has eight terms, each term is a set of one dyad, e_{ij}^2 or a two-subsystem loop i.e. **$e_{ij}e_{ji}$** and three independent subsystems (dyad is a system of two subsystems **i** and **j** , considered as one entity).

Group 4: Each term of the fourth grouping represents a set of three-element ***iMLE*** subsystem interaction loops (**e_{ij} e_{jk} e_{ki}** or its pair **e_{kj} e_{ji}**) and the composite system measure of the remaining (**N-3**) unconnected elements. Group has (2*5) 10 terms in all. Each term has a set of one 3-subsystem loop (**e_{ij} e_{jk} e_{ki}**) and independent subsystems. The three-subsystem loop is a system, to be considered as one entity.

Group 5: The fifth grouping contains two subgroups. The terms of the first subgrouping consist of two-element ***iMLE*** subsystem interaction loops (i.e., **e_{ij} e_{ji}** and **e_{kl} e_{lk}**) and ***iMLE*** constituent **e_m** .The terms in the second grouping are a product of four-element ***iMLE*** subsystem interaction loops (i.e., **e_{ij} e_{jk} e_{kl} e_{li}**) or its pair (i.e., **e_{il} e_{lk} e_{kj} e_{ji}**)and ***iMLE*** constituent **S_m** . Group has two subgroups: Group 5(i) has ten terms; each term is a subset of two independent dyads (e_{ij}^2, e_{kl}^2) or two-subsystem loops and one independent subsystem. Group 5(ii) has nine terms; each term is a set of 4-subsystem loop (**e_{ij} e_{jk} e_{kl} e_{li}**) and one independent subsystem.

Group 6: The terms of the sixth grouping are also arranged in two sub-groupings. The terms of the first sub-grouping are a product of a two-element ***iMLE*** subsystem interaction loop (i.e., **e_{ij} e_{ji}**) and a three-element ***iMLE*** subsystem interaction loop (i.e., **e_{kl} e_{lm} e_{mk}**)or its pair (i.e., **e_{km} e_{ml} e_{lk}**).The second sub-grouping consists of a five-component ***iMLE*** subsystem interaction loop (i.e., **e_{ij} e_{jk} e_{kl} e_{lm} e_{mi}**)or its pair (**e_{im} e_{ml} e_{lk} e_{kj} e_{ji}**). Group has again two subgroups: Group 6(i) has one 3-subsystem loop and a dyad or two-subsystem loop while Group 6(ii) has three 5-subsystem loops.

By providing/associating proper physical meaning to the VPF-***iMLE*** structural components, appropriate interpretation is obtained:

- e_{ij}^2 is interpreted as a two-system structural dyad, for example, e_{35}^2 represents the dyad of interaction between **ITS** and **MIS** systems.
- $e_{ij}\ e_{jk}\ e_{ki}$ is a three system structural loop, for example, $e_{12}\ e_{23}\ e_{31}$ represents the three system structural loop between ***EHAS, MDS*** and ***ITS*** systems.
- $e_{ij}\ e_{jk}\ e_{kl}\ e_{li}$ is a four system structural loop, for example, $e_{23}\ e_{34}\ e_{45}\ e_{52}$ represents four system structural loop, between ***MDS, ITS, MoAS*** and ***MIS*** systems.

In all, a general 5-subsystem permanent function will have **5!** i.e., 120 terms (subsets) arranged in (N + 1) groups. Figure 9 gives graphical/physical interpretation of terms of different groups for visual understanding, analysis, and improvement of a ***iMLE*** system architecture. It is therefore possible for the system analyst and designer to carry out **SWOT** (strength-weakness-opportunities-threats) analysis of their complete ***iMLE*** system and take strategic decisions to their advantage as per policy.

7. Modular design and analysis of iMLE system

Different terms of permanent function, equation (8) of ***iMLE*** system Figure 8 represent different subsets Figure 9 of the system. As these terms consist of structural terms S_i, $e_{ij}^2/e_{ij}\ e_{ji}$, $e_{ij}\ e_{jk}\ e_{kl}\ e_{li}/e_{il}\ e_{lk}\ e_{kj}\ e_{ji}$ **etc.,** global ***iMLE*** system solution providers offer different alternative solutions for each of these structural modules/subsystems. If there are **n** distinct terms in the permanent function, there are **n** ways of designing and analyzing the ***iMLE*** system. If the system is already in place, the **SWOT** analysis can help in improving the existing system. If the designer is using the proposed methodology to develop optimum design solution at conceptual stage, it can be done in the presence of available standard solutions of modules e.g. subsystems (S_i), dyads (e_{ij}^2) and loops ($e_{ij}\ e_{jk}\ e_{kl}\ e_{li}$) etc. Thus physical representation of Figure 9 helps in analyzing and designing ***iMLE*** system comprehensively using the structural modules. This permanent function and its interpretations become basis of modular analysis and design of ***iMLE*** system.

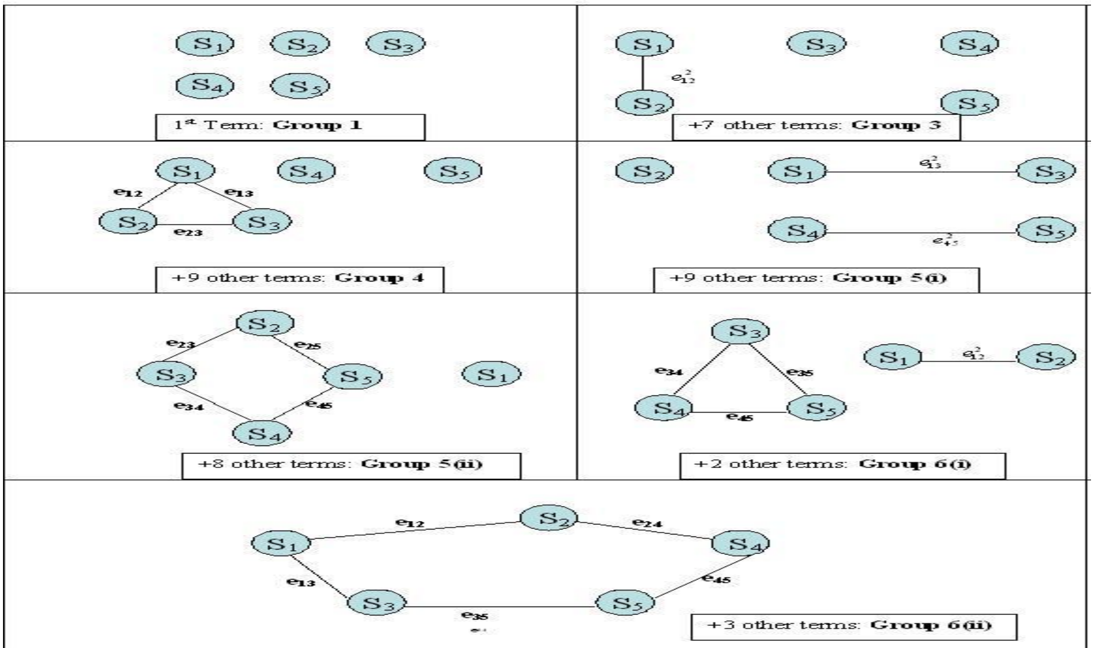

Figure 9 Graphical/Physical representation of Permanent Function Expression for *iMLE*

8. Evaluation of V_p

The diagonal elements of the matrix in equation (**7**) correspond to the five subsystems that constitute a ***iMLE*** system. The values of these diagonal elements S_1, S_2… S_5 are calculated as:

$$\mathbf{S_1 = Per(V_pS_1)} \qquad \mathbf{S_2 = Per(V_pS_2)} \qquad \mathbf{S_3 = Per(V_pS_3)}$$
$$\mathbf{S_4 = Per(V_pS_4) S_5 = Per(V_pS_5)} \tag{8a}$$

Where $\mathbf{V_p}S_1$, $\mathbf{V_p}S_2$, $\mathbf{V_p}S_3$, $\mathbf{V_p}S_4$, $\mathbf{V_p}S_5$ are the variable permanent matrices for five subsystems of the ***iMLE*** system. The procedure for calculating $\mathbf{S_1}$, $\mathbf{S_2}$… $\mathbf{S_5}$ is the same as for calculating **Per**($\mathbf{V_p}$)of equation (**8**).For this purpose, the subsystems of ***iMLE*** system are considered, and the procedure given below is followed:

1. The schematics of these subsystems are drawn separately by considering their various sub-sub-systems.
2. Identify the degree of interactions, interconnections, dependencies, connectivity, etc. between different subsubsytems.

Digraph representations (like Figure 7) of five subsystems are drawn first separately to obtain their matrix equations (like Equation (**8**)) i.e. $\mathbf{V_pS_i}$ and then their permanent functions $\mathbf{Per(V_pS_i)}$, $\mathbf{S_i}$, **i** = 1,…,5. The off-diagonal terms $\mathbf{e_{ij}}$ (**i ,j** = 1,2,…,5) of matrix equation (**7**) gives the connections between the systems $\mathbf{S_i}$ and $\mathbf{S_j}$. Depending upon the type of structural analysis, $\mathbf{S_{ij}}$ can be represented as multinomial, graph, and matrix or by some analytical model. To get the exact degree of interactions, interconnections, dependencies, connectivity, etc. between subsystems or subsubsystems we may have to consider the views of technical team experts. A team of experts selected from system analyst, design, software engineering, computer science, information systems etc. to consider all the issues involved from the point of view of engineering, science, technology, and business strategy. The final decision on the values of $\mathbf{S_i}$ and $\mathbf{S_{ij}}$ may be taken on the recommendations of the team. Thus, following the top-down approach and the step-by-step procedure given below will give the complete structural analysis of the ***iMLE*** system.

9. Compact representation of permanent function

The variable permanent function (**VPSSM- *iMLE***) being the characteristic of ***iMLE*** system of any industrial product is a powerful tool for its evaluation and analysis. The **VPSSM- *iMLE*** system expression, which corresponds to the five-factor digraph and matrix, equation (**6**), is written in a compact sigma ($\sum$) form.

VPSSM- *iMLE* = Per($\mathbf{V_p}$)

$$\prod_{1}^{5} S_i + \sum_i\sum_j\sum_k\sum_l\sum_m \left(e_{ij}e_{ji}\right)S_kS_lS_m + \sum_i\sum_j\sum_k\sum_l\sum_m \left(e_{ij}e_{jk}e_{ki} + e_{ik}e_{kj}e_{ji}\right)S_lS_m$$
$$+\left(\sum_i\sum_j\sum_k\sum_l\sum_m \left(e_{ij}e_{jk}\right)\left(e_{kl}e_{lk}\right)S_m + \sum_i\sum_j\sum_k\sum_l\sum_m \left(\left(e_{ij}e_{jk}e_{kl}e_{li}\right) + \left(e_{il}e_{lk}e_{kj}e_{ji}\right)\right)S_m\right)$$
$$+\sum_i\sum_j\sum_k\sum_l\sum_m \left(e_{ij}e_{ji}\left(e_{kl}e_{lm}e_{mk} + e_{km}e_{ml}e_{lk}\right) + \sum_i\sum_j\sum_k\sum_l\sum_m \left(e_{ij}e_{jk}e_{kl}e_{lm}e_{mi} + e_{im}e_{ml}e_{lk}e_{kj}e_{ji}\right)\right) \tag{9}$$

J A Q M

Vol. 2
No. 2
Summer
2007

The above equation is a generalized mathematical expression in symbolic form corresponding to five-factor digraph representation. It ensures an estimate of the ***iMLE*** system of any industrially integrated product. The above equation contains **5!**terms. Each term is useful for a ***iMLE*** designer as each term serves as a test for the effectiveness of the relevant group in **Per**(**V_P**).

10. Generalization of methodology

Suppose a system consists of **N** subsystems in place of proposed five subsystems and is represented as a digraph, then the most general way of matrix representation is shown below. This matrix is also known as the variable permanent matrix (**VPSSM- *iMLE***) corresponding to the **N** subsystems.

$$\begin{array}{c l}
\begin{matrix} 1 & 2 & 3 & . & . & . & . & N \end{matrix} & \text{Subsystems} \\
\begin{bmatrix}
S_1 & e_{12} & e_{13} & . & . & . & . & e_{1N} \\
e_{21} & S_2 & e_{23} & . & . & . & . & e_{2N} \\
e_{31} & e_{32} & S_3 & . & . & . & . & e_{3N} \\
. & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . \\
e_{N1} & e_{N2} & e_{N3} & . & . & . & . & S_N
\end{bmatrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ . \\ . \\ . \\ . \\ N \end{matrix}
\end{array} \tag{10}$$

Permanent for the above matrix, i.e., **Per**(**V_P**)is called variable permanent function (**VPSSM- *iMLE***).The **VPSSM- *iMLE*** for the above matrix is written in sigma form as

$$\begin{aligned}
\mathrm{Per}(V_P) = {} & \\
& \prod_{x=1}^{N} S_x + \sum_i \sum_j \sum_k \sum_l \ldots \sum_N \left(e_{ij} e_{ji}\right) S_k S_l \ldots S_N + \sum_i \sum_j \sum_k \sum_l \ldots \sum_N \left(e_{ij} e_{jk} e_{ki} + e_{ik} e_{kj} e_{ji}\right) S_l S_m \ldots S_N \\
& + \left(\sum_i \sum_j \sum_k \sum_l \ldots \sum_N \left(e_{ij} e_{jk}\right)\left(e_{kl} e_{lk}\right) S_m \ldots S_N + \sum_i \sum_j \sum_k \sum_l \ldots \sum_m \left(\left(e_{ij} e_{jk} e_{kl} e_{li}\right) + \left(e_{il} e_{lk} e_{kj} e_{ji}\right)\right) S_m \ldots S_N \right) \\
& + \sum_i \sum_j \sum_k \sum_l \ldots \sum_N \left(e_{ij} e_{ji} \left(e_{kl} e_{lm} e_{mk} + e_{km} e_{ml} e_{lk}\right) S_n S_o \ldots S_N + \sum_i \sum_j \sum_k \sum_l \ldots \sum_m \left(e_{ij} e_{jk} e_{kl} e_{lm} e_{mi} + e_{im} e_{ml} e_{lk} e_{kj} e_{ji}\right) S_n S_o \ldots S_N \right) \\
& + \ldots
\end{aligned} \tag{11}$$

The number and composition of groups and subgroups will be the same as discussed earlier. So it is possible to write the permanent function of any ***iMLE*** system in (**N + 1**) groups. It may be noted that a permanent function will contain **N!** terms only, provided e_{ij} are not 0. In certain cases, designers and/or developers team may decide that some of e_{ij} are 0 because of insignificant influence of one subsystem over the other subsystem. Substitutions of corresponding **e_{ij}** equal to 0 in general permanent function (equation (**9**)) or in general **VPM** (equation (**8**)) gives the exact number of terms with modified permanent function.

11. Step-by-step procedure

The step-by-step methodology is proposed which can permit industry, university and organizations to modify, extend, and improve quality of their ***iMLE*** products. Various marketing and strategic decisions can also be taken as per the competitiveness of ***iMLE*** products in global market. It will also give an insight to researchers, designers and developers to identify, select and create critical systems integration process. A generalized procedure for the complete design and analysis of ***iMLE*** system architecture is summarized below:

Step 1: Consider the desired *iMLE* product. Study the complete *iMLE* system and its subsystems, and also their interactions.

Step 2: Develop a block diagram of the ***iMLE*** system Figure 2, considering its sub-systems and interactions along with assumptions, if any.

Step 3: Develop a systems graph of the ***iMLE*** system Figure 8 with sub-systems as nodes and edges for interconnection between the nodes.

Step 4: Develop the matrix equation (10) and multinomial representations equation (11) of ***iMLE*** system.

Step 5: Evaluate functions/values of diagonal elements from the permanent functions of distinct sub-systems equation (8a) of the composite and repeat Steps 2 – 4 for each sub-system.

Step 6: Identify the functions/values of off-diagonal elements/interconnections at different levels of hierarchy of the ***iMLE*** amongst systems, sub-systems, sub-sub-systems, etc.

Step 7: Carry out modular design and analysis of ***iMLE*** products while purchasing off the shelf from the global market.

The visualization of the step-by-step procedure for the complete design and analysis of ***iMLE*** system is shown in Figure 10.

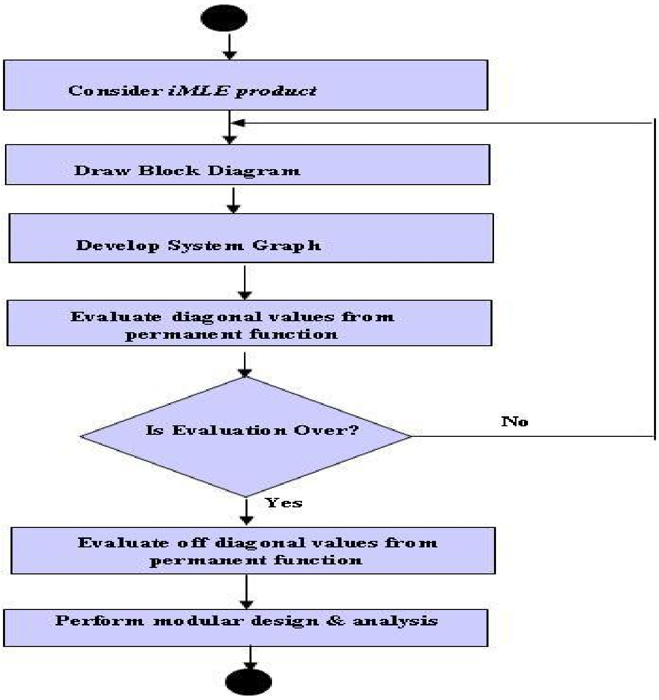

Figure 10. Visualization Model

The values (or functions) of interactions $\mathbf{e_{ij}}$ (i ,j = 1,2,…,N) between different subsystems $S_1, S_2, \ldots, S_N$ can be written as a multinomial or a matrix, depending upon the type of interaction/reaction between the two subsystems. The sub-subsystems can again be treated as systems, as every sub-subsystem is a system in itself. Following the above procedure, these subsystems can be broken down into sub-subsystems and different graphs, matrices, and permanent representations can be obtained. Depending upon the depth of analysis required, the process could be taken to the constituent level and further ahead. In certain cases, it may be possible to evaluate $\mathbf{e_{ij}}$ **'s** experimentally or using available mathematical models. With the help of this data, complete multinomial for the ***iMLE*** system can be evaluated. Using/available standard modules of ***iMLE*** architectural sub-systems (e.g. dyads and loops of different subsystems) in global market, designers can develop alternative designs of ***iMLE*** products and carry out analysis and improvement of existing ***iMLE*** products. Work is in progress to carry out performance analysis of any ***iMLE*** system architecture from different perspectives using the structural model presented in this article.

12. Conclusions

The following concluding remark highlights the contributions of the present study.

1. The proposed ***iMLE*** system architecture is developed using system methodology and graph theoretic model. They represent its structural information, including its systems, their subsystems and their interconnections.
2. The systems methodology consists of the ***iMLE*** system digraph, the ***iMLE*** system matrix, and the ***iMLE*** system permanent function. These permit us to derive and exploit a number of results, which are useful to analysts, designers and developers of the system for quality products.
3. The ***iMLE*** digraph is the mathematical representation of the structural characteristics and their interdependence, useful for visual modeling and analysis. The ***iMLE*** system matrix converts digraph into another mathematical form. This matrix representation is a powerful tool for storage and retrieval of subsystems in computer database and also for computer processing. The ***iMLE*** system permanent function is a mathematical model characterizing the structure of the ***iMLE*** product and also helps one to determine the ***iMLE*** system index.
4. The permanent function of the ***iMLE*** system architecture at a particular level of hierarchy represents all possible combination of its subsystems. The terms of permanent function not only represent different subsets of ***iMLE*** system architecture but also guide the analysts, designer, developer, manager, decision maker and purchaser to generate large number of alternative design solution before selecting an optimum system.
5. The present work emphasizes the numerical methodology of ***iMLE*** that can also optimize the design and the development parameters.
6. The proposed systems model is a very a powerful tool from the commercial point of view in this highly competitive world. As the industry, university and organization has complete knowledge of every sub-system and all process parameters and their interactions through this systems model, they have a number of choices to shape its designing and developing strategy based on market dynamics.

7. As it is an integrated systems approach, all the subsystems up to the component level are modeled and evaluated to be used as inputs for diagonal elements at next higher level and so on. It can be inferred that to get the structural performance level (i.e. permanent index) of the overall system, the structural performance level of each subsystem at the lower level need to be calculated and substituted as diagonal elements of the variable permanent adjacency matrix at higher level.
8. The proposed structural methodology is comprehensive enough to deal with different structural and performance issues of **iMLE** system architecture at different levels of its life cycle.
9. A generalized methodology is also proposed to model a system consisting of **N** sub-systems and their interactions.
10. Current undergoing research deals with correlation of structural models with the desired performance parameters and associated characteristics. The outcome will be reported in future publications.

References

1. **Bee-Gent and Plangent 2003,** Available online at March 2003 in: http://www.toshiba.co.jp/rdc/plangent/index.htm.
2. Bosch, J. **Design and Use of Software Architectures,** Addison-Wesley, 2000
3. Bucur, O., Boissier, O., Beaune, P. **A Context-Based Agent Architecture for Learning: How to Make Contextualized Decisions,** in "Proceedings of First International Workshop on Managing Context Information in Mobile and Pervasive Environments", 2005
4. Cai, X., Lyu, M. R., Wong K. F., Ko, R., **Component Based Software Engineering: Technologies, Development Frameworks and Quality Assurance Schemes,** in the "Proceedings of the Asia-Pacific Software Conference", 2000
5. Capuano, N., Marsella, M., Salerno, S. **ABITS: An Agent Based Intelligent Tutoring System for Distance Learning,** in "Proceeding of the International Workshop an Adaptive and Intelligent Web-Based Education Systems", 2000
6. Chen, S., Nahrstedt, K. **Distributed Quality-of-Service Routing in Ad Hoc Networks,** IEEE Journal on Selected Areas in Communications, 2000, pp. 1594-1603
7. Chen, Y. S., Lai, K.C. ***MESH:* Multi-Eye Spiral-Hopping Protocol in a Wireless Ad-Hoc Network,** IEICE Transactions on Communications, E84-B(8), 2000, pp. 2237-2248
8. Chiang, C. C., Gerla, M., Zhang, L. **Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks,** Cluster Computing, 1998, pp. 187-196.
9. Clements, P., Bachmann, F., Bass, L., Garland, D., Ivers, J., Little, R., Nord, R., Stanford, J. **Documenting Software Architectures: Views and Beyond,** Addison-Wesley, 2002
10. Deo, N. **Graph Theory with Applications to Engineering and Computer Science,** Prentice-Hall of India Private Limited, New Delhi,2004
11. Fabiano, A. D., Carlos, R. L., Marcia, A. F. **A Multiagent Architecture for Distance Education Systems,** in "Proceedings of the 3rd IEEE International Conference on Advanced Learning Technologies", 2003
12. Gandhi, O. P., Agrawal, V. P., Shishodia, K. S. **Reliability Analysis and Evaluation of Systems,** Reliability Engineering and System Safety, 32, 1991, pp. 283-305
13. Goel, A. L., Okumoto K., **Time Dependent – Error Detection Rate Models for Software Reliability and Other Performance Measures,** IEEE Transactions on Reliability, 1979
14. Gray, R. **Agent TCL: A flexble and Secuer Mobile AgentSystem,** PhD Thesis, Dept. Of Comp. Science, Dartmouth College, 1997

15. Jelinski, Z. Moranda, P. B. **Software Reliability Research, Statistical Computer Performance Evaluation,** edited by Freigerger, W., Academic Press, 2001
16. Jennings, N. *et al.* **Automomous Agents for Business Process Management,** International Journal of Applied Artificial Intelligence, 2000
17. Johnson, D. B., Maltz, D. A. **Dynamic Source Routing in Ad Hoc Wireless Networks,** Mobile Computing, edited by Mielinski, T. and Korth, H. Chapter5, 1996, pp. 81-153.
18. Jurkat, W. B., Ryser, H. J. **Matrix Factorization and Permanents,** J. Algebra, 16, 1996, pp. 1-26.
19. Kozaczynski, W., Booch, G. **Component Based Software Engineering,** IEEE Software, 1998
20. Kruchten, P. **The 4+1 View Model of Architecture,** IEEE Software, 1995
21. Lehner, F., Nosekabel, H. **The Role of Mobile Devices In E-Learning -First Experiences With A Wireless E-Learning Environment,** WMTE 2002, pp. 103-106.
22. Lin, C. R., Liu, J. S. **QoS Routing in Ad Hoc Wireless Networks,** IEEE Journal on Selected Areas in Communications, 17(8), August 1999, pp. 1426-1438
23. Littlewood, B. A., Vernall, J. L. **A Bayesian Reliability Growth Model for Computer Software,** Applied Statistics, 1973
24. Littlewood, B., Strigini, L., **Software Reliability and Dependability: A RoadMap,** in Finkelstein, A. (editor) "The Future of Software Engineering", ACM Press, 2000
25. Luo, K. Q., Huang, Y. L. **Intelligent Decision Support for Waste Minimization in Electroplating Plants,** Surface and Coating Technology, 2005, pp. 2223-2227
26. Lyu, M. R. **Handbook of Software Reliability Engineering,** McGraw-Hill, 1996
27. Lyu, R. M., Rangarajan, S., Aad, P. A. **Optimal Allocation of Test resources for Software Reliability Growth Modeling in Software Development,** IEEE Transaction of Software Engineering, 2002
28. Maes, P., Guttman, R. H. **Cooperative vs. Competetive Multi-Agent Negotiations in Retail Electronic Commerce,** "Proceedigns of the Second International Workshop on Mobile Agents (CIA'98)", Paris, 1998
29. Guttman, R. H., Moukas, A. G., Maes, P. **Agent-mediated Electronic Commerce: A Survey,** Knowledge Engineering Review, 1998
30. May, J. H. R., Lunn, A. D. **A Model of Code Sharing for Estimating Software Failure on Demand Probabilities,** IEEE Transaction of Software Engineering, 1995
31. May, J. H. R., Lunn, A. D., **New Statistice for Demand Based Software Testing,** Information Processing Letters, 1995
32. Medvidovic, N., Taylor, R. **Separating Fact from Fiction in Software Architecture,** in "Proceeding of the 3rd International Software Architecture Workshop", 1998
33. Mohan, M., Gandhi, O. P., Agrawal, V. P. **Systems Modeling of a Coal-Based Steam Power Plant,** "Proceedings of Institution of Mechanical Engineers. UK. Part A: Power and Energy", 217, 2003, pp. 259-277
34. Musa, J.D., Okumoto, K. **Logarithmic Poisson Execution Time Model for Software Reliability Measurement,** in "Proceedings of Compsac", 1984
35. Papaionnou, T., Edwards, J. **Using Mobile Agents To Improve the Alignment Between Manufacturing and its IT support Systems**, Journal of Robotics and Autonomous Systems, Vol. 27, 1998, pp. 44-57.
36. Pesty, S., Webber, C. **The Baghera Multiagent Learning Environment: An Educational Community of Artificial and Intelligent agents. Upgrade,** Journal of CEPIS (Coucil of European Professional Informatics Societies), 4, 2004, pp. 40-44
37. Randell, B., **Software Dependability: A Personal View,** in "Proceedings of 25th International Symposium of Fault Tolerant Computing", IEEE Computer Society Press, 1995
38. Sacks, I. J., Ashmore, B. C., Alesso, H. P. **Systems Interaction Results from the Digraph Matrix Analysis of Watts bar Nuclear Power Plant High Pressure Safety Injection System,** Lawrence Livermore National Laboratory, Livermore, USA. Report No. UCRI-53467, Vol 1 and 2, 1993, pp. 16-33.

39. Saradhi, M. **System Modelling and Description,** Software Engineering Notes, ACM SIGSOFT, 1992
40. Seghal, R., Gandhi, O. P., Angra, S. **Reliability Evaluation and Selection of Rolling Element Bearings,** Reliability Engineering and System Safety, 68, 2000, pp. 39-52
41. Shaw, M., Garlan, D. **Software Architecture-Perspectives on an Emergine Disipline,** Prentice Hall, 1996
42. Tang, T. Y., Wu, A. **The Implementation of a Multiagent Intelligent Tutoring System for the Learning of Computer Programming,** in "Proceedings of 16th IFIP World Computer Congress International Conference on Educational Uses of Communication and Information Technology, ICEUT", 2000
43. Upadhyay, N. **Analysis and Design of Algorithms,** S. K. Kataria and Sons, New Delhi, 2004
44. Upadhyay, N. **M-Learning- A New Paradigm in Education,** International Journal of Instructional Technology and Distance Learning, 3(2), 2006, pp. 31-34.
45. Upadhyay, N. **Role of AI in Effective E-Learning,** in "Proceedings of National Conference on Cyber Security, Data Mining and ICT for Society", CCSDIS06, 2006, p. 24
46. Woit, D. **Specify Component Interactions for Modular Reliability Estimation,** in "Proceedings of First International Software Quality Week", 1997
47. * * * **Windows CE for Mobile Devices White Paper,** Microsoft Corp., 2001

---

[1] Nitin Upadhyay is an acknowledged teacher and prolific writer. He is currently working as Faculty in the Department of Computer Science and Information Systems, BITS, PILANI-GOA Campus. He has created a definite niche for himself in the field of Computer Science by contributing eight books. His research and creative zeal has enabled him to contribute research papers for journals and conferences. His major research areas are mobile learning and mode of education, mobile computing, system engineering, object oriented system, graph theory, software architecture and software engineering.

[2] Dr. V. P. Agrawal is working as a visiting professor of Mechanical Engineering at the Birla Institute of Technology and Science, Pilani, Goa campus, Goa, India after his retirement from IIT Delhi, India. He had been at IIT Delhi for the last 28 years and scaled from lecturer to professor in the Mechanical Engineering Department. He completed his graduation in 1966, post graduation in 1969 and PhD in 1983 from Jiwaji University Gwalior, University of Roorkee, and IIT Delhi respectively. He has published around 120 papers in International Journals and Conferences. He has guided successfully a large number of BTech, MTech projects and supervised a number of PhD theses. He has worked extensively in the areas of mechanisms, systems approach, and machine design.