AN APPROACH TOWARDS THE DEVELOPMENT OF AN EXPERT SYSTEM FOR PAEDIATRIC PROBLEM DOMAIN

THESIS SUBMITTED FOR THE DEGREE

OF

DOCTOR OF PHILOSOPHY (SCIENCE)

OF

THE UNIVERSITY OF NORTH BENGAL

1999

BY

MADAN GOPAL GOSWAMI

TO MY PARENTS

The thesis contains the results of the work done during the last five years in the Expert Systems Laboratory, Department of Computer Application, North Bengal University. A part of the results has already been published. I would like to record my gratitude to my Supervisors, Dr. R. K. Samanta, Reader of the Department of Computer Application, North Bengal University and Dr. (Mrs) Mridula Chatterjee for their constant care and invaluable guidance. I am indebted to the Council of Scientific and Industrial Research, Human Resource Development Group, Govt. of India for granting me Senior Research Fellowship for the Project entitled "KID : Expert System for Foetal and Child Growth and Development for North Bengal Districts - An Object-Oriented-Approach". I must acknowledge my gratitude to the authorities of i) Department of Computer Application and Computer Centre, North Bengal University, ii) North Bengal Clinic and Research Centre, Siliguri, for giving me the privilege to collect relevant data as well as for their kind gesture in giving me all the facilities I needed during the work. My gratitude to Dr. Santanu kar and Dr. Salil Datta can hardly be overstated. They are who gave me inspiration when I needed it most. Moreover, I would like to record my gratitude to Prof. D. Dutta Majumder, Professor Emeritus of ISI, & Emeritus Scientist of CSIR, Indian Statistical Institute, Calcutta; Er. B. Majumder, Executive Engineer, AGRI-MECH, Jalpaiguri Division, Jalpaiguri; Dr. S. C. Pal, Lecturer, Department of Computer Application, North Bengal University; and Dr. D. Pal, Lecturer, Department of Paediatrics, North Bengal Medical College & Hospital for their various kinds of help. My sincere thanks are due to Dr. D. Rastogi, Consultant Radiologist and Sonologist for his valuable advice and guidance during data collection. I wish to pay my special thanks to Dr. A. K. Saha, Expert Systems Laboratory, Department of Computer Application, North Bengal University for his continuous help all the time during last five years. His suggestions and advice as a Co-author of some part of my work should duly be acknowledged. Lastly, I am specially thankful to my wife Swati whose encouraging role during the hectic days and nights of my work defies exaggeration. No less inspiring was the role of my parents as well as that of the other members of our family during the period and I must thank them all from the core of my heart.

Madan Gopal Goswami

(Madan Gopal Goswami)
15th September 1999

Expert Systems Laboratory
Department of Computer Application
North Bengal University, Darjeeling, 734 430, INDIA.

[redacted]

Child Activities

[redacted]

Typical Classes and **Objects**

[redacted]

KID

[redacted]

List of Publications

INTRODUCTION

1.1. Introduction

A child is the 'future' of a nation. Every nation should take proper care of her each and every child. In this respect, there are some socio-economic problems which obviously varies from country to country. The degree of complexity of the paediatric domain varies with adult vs. child, urban vs. rural, Eastern vs. Western, as well as developing vs. developed countries. The incidence of low birth weight, mortality and morbidity rate etc. widely varies from country to country; and even this account varies from one zone to the other within the same country. **Chapter 2** of this thesis has been devoted to have an account of the global as well as Indian scenario; and this account demands much more attention towards the paediatric domain.

The total paediatric age group is subdivided as: Foetal (Conception - birth), Neonates (0-4 weeks), Infants (4 weeks -1 year), Toddler (1year -3 years), Pre-school (3 years -5 years), School going (5 years -10 years) and Adolescence (10 years -18 years). It should be pointed out that the disease pattern, drug selection, diet, common rearing technologies are different for different paediatric age groups. The following examples should illustrate the situation.

Case 1 : **Not passing urine (Anuria)** for **24-48** hours

Analysis : If the baby is one or two days old, the likely cause is increased environmental temperature with less fluid intake. The significant pathological problems with kidneys are unlikely. For other age groups specially for adults, however, one has to think of the pathological conditions.

Case 2 : **Cardiac diseases**

Analysis : Congenital defects of heart are more common at early age groups (before 5 years); Rheumatic heart diseases are common after 5 years of age; and Coronary heart diseases are common during old age. Moreover, history taking and / or examination procedures in paediatric patients are certainly different from adults.

Case 3 : **Jaundice**

Analysis : If the baby is one or two days old, the possible causes may be ABO or Rh incompatibility, or intra-uterine TORCH infection. The cause of Jaundice at this age group owing to viral hepatitis (Hepatitis A, Non-A-Non-B etc.) are rare but common for other age groups.

Case 4 : **History taking typical to paediatric domain**

History may not come from the patient himself / herself but from accompanying persons.

Case 5 : **"Baby is crying since morning" - complaint**

There is no further information available from the patient and / or from the parents / guardians. Here, the doctor has to find out first where the trouble is with the baby, and then what the cause could be, where the sites of the disease might have been pointed out convincingly by an adult, like, stomach pain, earache or headache. Here the doctor might has to seek for the following informations.

- Family history and social history
  e.g. father-mother relationship, blood dyscrasias;
- Antenatal, intra-natal and post-natal history;
- Milestones;
- Dietary history; and
- Immunization history.

Case 6 : **Examination procedures to paediatric patients**

Usually, the examinations are done using the following four steps in order :

i) Inspection;
ii) Palpation;
iii) Percussion; and
iv) Auscultation.

Inspection is the act of visual examination of the external surface of the body as well as of its movements and posture. Palpation is the process of examining by application of the hands or fingers to the external surface of the body to detect evidence of disease or abnormalities in the various organs. Percussion is the process of use of the finger tips to tap the body lightly but sharply to determine position, size, and consistency of an underlying structure and the presence of fluid or pus in a cavity. Auscultation is the process of listening for sounds within the body, usually to sounds of thoracic or

abdominal viscera, in order to detect some abnormal condition, or to detect foetal heart sounds.

The examination steps should be followed in every case so that it should be methodological. Paediatric patients may be non-cooperative, hostile and resistant to such examinations and therefore, examination steps may have to be altered. For example, very often auscultation is done before palpation and percussion when the baby is quite and co-operative before he turns hostile owing to various examination procedures.

For the paediatric patients, there is a special importance of congenital anomalies affecting normal growth and development. In addition, Anthropometric measurements are also typical to paediatric domain. Even, the preparation of priority list of differential diagnosis might be dictated by the urban vs. rural difference as well as by the Eastern vs. Western difference.

Case 7 : **Diarrhoeal disease**

Analysis : a) Rural : Infective and protozoal diarrhoea are common.
b) Urban : Non-infective and viral causes are highly probable.

Case 8 : **Maculo-Papular Rash**

Analysis : a) Western : Rash due to Adeno virus or allergic cause is more common.
b) Non-Western : Rash due to Measles or Rubella is more common.

Considering the pre-school age group, it may be mentioned that in most countries of the world there is a relative neglect of such children [1]. They are "vulnerable" or special-risk group in any population, deserving special health care. The reasons why the under-fives merit special health care may be stated as follows :

(1) Large numbers : According to the 1991 Indian census, children of the age group 0-6 years constitute 20% of the total population. By virtue of their large number, they are entitled to a large share of health care.

(2) High mortality : In India, 35 to 40 of all deaths occur in children under the age of 5 years. The major causes of death in this age group are diarrhoeal diseases, malnutrition and infection - all preventable. Almost all deaths can be ascribed to failure to make prompt and effective use available means of treatment and prevention.

(3) Morbidity : The first 5 years are full of health hazards - e.g. diarrhoeal diseases,

malnutrition and infectious diseases (e.g. measles, whooping caugh, tetanus, polio). A very great prevalence of intestinal parasites in this age group is well-known.

(4) Growth and development : The pre-school age period is one of rapid growth and development.

(5) Human resource : Ensuring child health is an investment for the future.

(6) Accessibility : While the infant may be easily reached, the young child (1-4 years) is hard to reach. Special "inputs" are needed to bring the child into the orbit of special health care.

The health care management for under-fives is now undergone through 'under-fives' clinic in India [1]. The aims and objectives of the clinic are set out in the symbol as depicted in fig. 1.1.

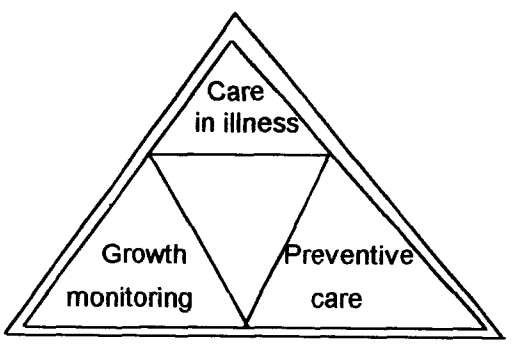


Fig. 1.1 Symbol for under-fives' clinic.

The **illness care** for children will comprise :

(A) Diagnosis and treatment of :

(i) Acute illness;
(ii) Chronic illness including physical, mental, congenital and acquired abnormalities;
(iii) Disorders of growth and development;

(B) X - ray and laboratory services; and

(C) Referrel services.

Preventive care should include :

(i) Immunization;
(ii) Nutritional surveillance;
(iii) Health check-up;
(iv) Oral rehydration;
(v) Family planning; and
(vi) Health teaching.

Growth monitoring is one of the basic activities of the under-fives clinic, i.e. to weigh the child periodically at monthly intervals during the first year, every 2 months during the second year, and every 3 months thereafter up to the age of 5 to 6 years. In this part, developmental activities are also considered (see Appendix A).

Health status of an individual is determined by two ecological ends, the internal environment of a child itself including heredity and the external environment or the surroundings. A delicate balance between these two ends is required for the normal growth and development of a child. The major components of the internal environment are the racial and genetic characteristics. The external environment is governed mostly by food, water, housing, education and sanitation which are controllable factors. Understanding the importance of 'external environment', it is termed 'environmental health' [2] by WHO (World Health Organisation). For the proper development of a child we have to improve the 'environmental health' .

As a representative example, let us have a look on the status of North-Bengal districts of India [+].

1.1.1. North-Bengal Districts of India

The geographical position of the region has been pointed out in fig.1.2.

The total area is 39,864 sq.km. Around 98% of the total geographical area is the rural area. At per the 1991 Indian Census [3], the total population is 1,20,36,292. Around 86% of the total population live in rural areas. 20% of total rural population is of the group 0-6 years. The rural literacy rate (from age group 7 +) is 38%. A significant percentage (46%) of rural population lays below the poverty line (Rs.1200/- per annum). There is one primary health centre per 23 sq. kms. area.

The region is still lagging behind the standard in the environmental health. As a representative example, a study [4] shows that there is an acute problem with standard drinking water supply.

[+] This is based on the publication [CSI Communications, May 1997, 15-21; ibid. June 1997, 21-25] of the author.

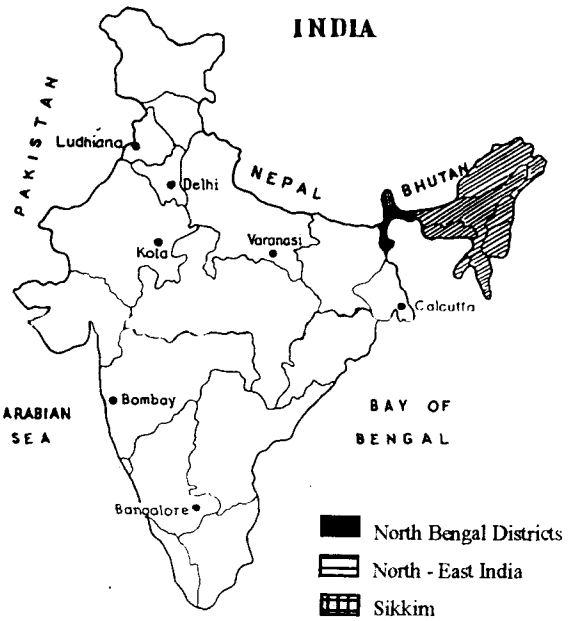


Fig. 1.2. INDIA

In a recent study [5], it has been observed that there is a marked difference of foetal growth in terms of birth weight for North Bengal region of India compared to some Western studies as shown in fig. 1.3. This was conducted on lower as well as higher socio- economic groups which comprise 858 and 1169 live born babies respectively using classical technique of measuring weights after birth. It is evident from the fig. 1.3 that the birth weights of fetuses of mothers of lower socio-economic group (LSEG) are much lower than the mothers of higher socio-economic group(HSEG).

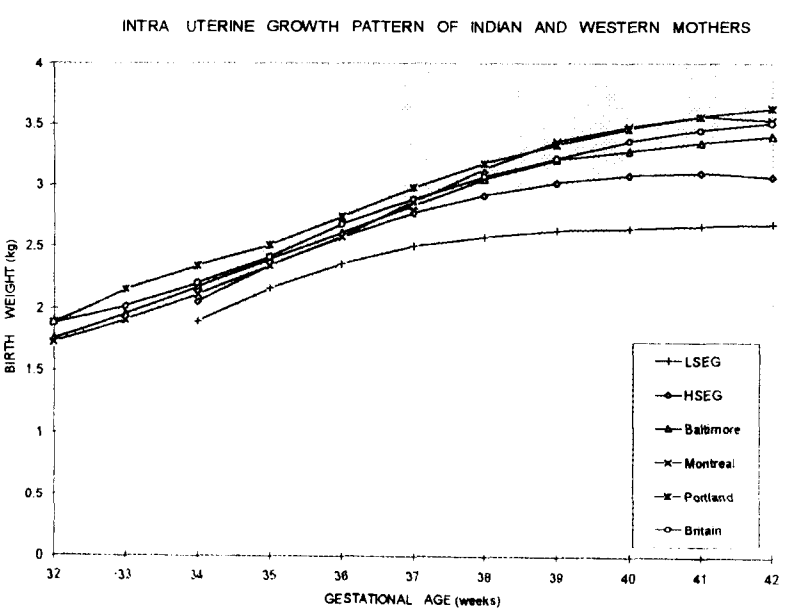


Fig. 1.3. Intra-uterine Growth Pattern of Foetus of Indian and Western Mothers.

Next, one has to find out the status of growth performance of children after birth. The data offered by the National Centre for Health Statistics (NCHS) of USA [6] are considered standard for child growth in many countries including India. The NCHS standard does not differ significantly from the Harvard standard in use earlier, as far as under fives are concerned. There is some difference with respect to the older age groups. In 1991, Agarwal et al. [7] conducted a study amongst affluent Indian children of seven different cities of India. From their studies, Agarwal et al. [7] did not find any significant difference from NCHS standard at least for the Indian regions, they considered. A study has been conducted recently [8] to find out the growth pattern of the rural children (0-5 years) belonging to poor socio-economic condition. The sample size is 1181 for girls and 1470 for boys from lower socio-economic status. Their observations are represented in figs.1.4 and 1.5 for growth in terms of weight. Observations on height, head-circumference and chest circumference are available from ref. 8. It is evident from the figures that the growth performance of children of lower socio-economic group is poor compared to NCHS and other data.

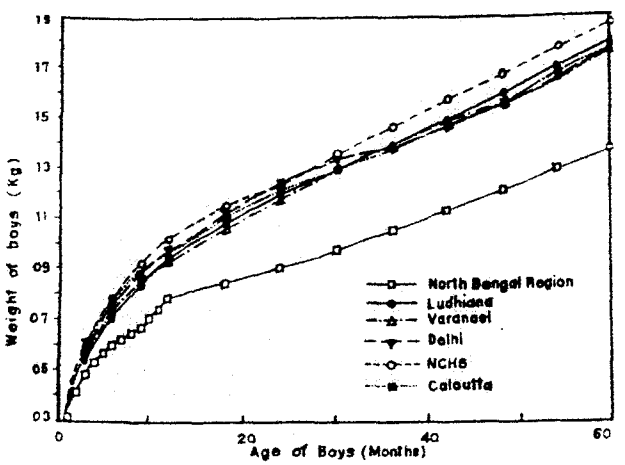


Fig.1.4. Weight of boys of lower socio-economic group and its comparison with reference values.

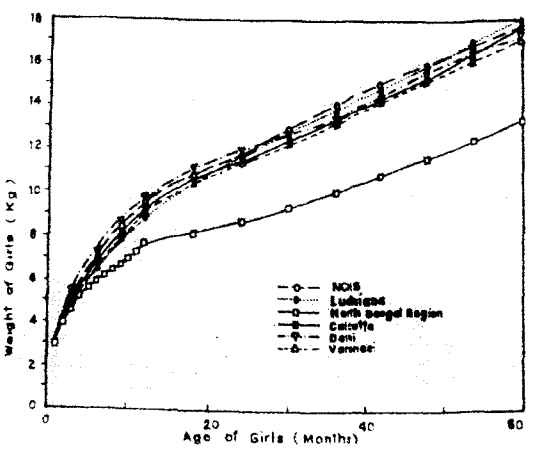


Fig.1.5. Weight of girls of lower socio-economic group and its comparison with reference values.

It is evident from the above facts and figures that this region is lagging behind the standard in the paediatric growth and development. This type of picture or even more discouraging pictures one may have from the facts and figures described in **Chapter 2**.

It is demanding to consider two matters concurrently : (1) improving the environmental health, and (2) improved health care management. Regarding the first matter different developing countries including India are taking actions by providing drinking water at per WHO specifications as well as supplying supplementary food to them. This work mainly concentrates on the second issue i.e., improved health care management. For the proper paediatric care the role of a paediatrician is pivotal. From the table 2.14

(Chapter 2) we get a picture of the availability of paediatricians working in rural regions of India what is not at all satisfactory. They are really a scarce commodity at per the requirements in general and specially for rural region of a developing country like India. To cover the complexity of the domain as well as mitigating the scarcity of domain experts one may consider an automated consultation system in an expert system framework as an alternative. Starting from MYCIN [9], a good number of attempts have been made to develop such expert systems in medicine with their relative success and failure. We may cite here example like BABY [10], a system for the paediatric domain, but, however, to the best of our knowledge, not so much of the issues relating to paediatric health care management have been addressed in this context. In this work an attempt has been made towards the development of a consultation system for the domain in an expert systems framework to partially meet up the requirements of the paediatric domain.

1.2. The Rise of Medical Informatics[+]

If the physiology literally means the 'logic of life', and pathology is the 'logic of disease', then medical informatics is the logic of health care [11]. It is the study of the way, we think about patients, and the way the treatments are undertaken. It is study of how the medical knowledge is created, analysed and are shared among professionals and applied to the medical domain. Ultimately, it is the study of how we can establish proper health care management.

Although the term 'medical informatics' appeared in the literature in around 1973, but the study is as old as when a doctor examined a patient, first recorded the findings and later used those information to his/her next patient. Medical informatics is as much about computers as cardiology is about stethoscopes. The field of medical informatics is concerned with computers, communications, doctors, patients, nurses and medical knowledge. The study of informatics in the coming century will probably be as fundamental to the practice of medicine as the study of anatomy has been this century.

The interests, role and applications of information sciences in medicine are growing exponentially this decade, the Internet decade, compared to the last decade of Eighties. Not only the patient records are being stored and retrieved but it covers everything from the manner in which evidence is used in clinical decisions, how one decides which piece of knowledge is relevant to a particular situation, and ultimately to how medical knowledge itself is created. Another dimension has been added with the introduction of world wide web (www) and Internet services. It has resulted some new techniques, concepts and ideas (e. g. telemedicine).

[+] This is based on the publication [Int. AMSE Conf. (CMSC-99), Jaipur, India: 1-3 Dec. 1999 (to appear)] of the author.

Summarily, medical informatics use more or less all methods of information technology such as acquisition, processing, control, interpretation, transformation, transfer and presentation of data for medical purpose. Networking of large health care groups, linking of hospitals and research centres, transfer of diagnostic and therapeutic information, video conferencing, application of hospital information system, expert systems for diagnosis, image analysis and pattern recognition for pathological investigations, telediagnosis and telemedicine, virtual reality training for surgical applications and modelling of brain for psychophysiological and individual animations will be useful and integral part of biomedical technology in the 21st century.

Let us now have a look on the the issues (i) Telemedicine, (ii) Internet and world-wide-web, and (iii) Artificial Intelligence in medicine (AIM) contributing to the advancement of medical informatics.

- **Telemedicine**

The essence of telemedicine is the exchange of information at a distance, whether that information is voice, an image, elements of a medical record, or commands to a surgical robot. It seems reasonable to think of telemedicine as the communication of information to facilitate clinical care.

At its inception telemedicine was essentially about providing communication links between medical experts in remote locations. The health care system, however, is clearly inefficient because of its poor communication infrastructure and telemedicine is now seen as a critical way of reducing that cost. One estimate suggests that the health system in the United States could save $30 billion a year with improved telecommunications [12]. Consequently, telemedicine has now become an important subject for research and development.

Currently, the press is flooded with articles about the information superhighway, the internet and the rapid growth in the use of mobile telephones. Telemedicine is often presented in the guise of sophisticated new communications technology for specialist activities such as teleradiology and telepathology.

Telephone systems can make significant improvements to the delivery of care. For example, follow up of patients is often possible on the telephone. Rapid communication of hospital discharge information using existing electronic data transfer mechanisms is beneficial for general practitioners. Perhaps more interestingly, inexpensive voice messaging systems can deliver simple but powerful services over existing telephone networks. A voice mail system can reduce both tardiness and complete forgetting. As more patients get access to electronic mail, this will offer further avenues for innovative health services. Already in some populations, access to electronic mail is high. All these

points suggest that the potential for the clinical application of communication technologies is indeed great, but equally that there is much still to learn. In particular, the relation between telemedicine and informatics needs to be explored in greater detail. Informatics focuses on the use of information, and telemedicine on its communication.

- **Internet and World-Wide-Web**

The internet is essentially a network of networks. It is an open and unregulated community of people who communicate freely across an international electronic computer network. The number of medical sites joining the internet increases monthly, as does the number of information resources available on it.

At present, the internet is in a phase of massive expansion, with growth rates of well above 10% per month. This is largely because, where before it's value for most people was marginal, it now offers communication and information services thet surpass those possible by plain telephony or television. As more of these information services become available, even more people are being persuaded to join.

For a long time, users communicated on the internet using electronic mail. This essentially allowed users to exchange text messages across the globe, usually in a matter of minutes. However, it was probably the advent of the World Wide Web that triggered the recent growth of the internet. Essentially a set of software standards, Web programs allow users to navigate rapidly across the global internet. Here they can view a bewildering variety of information, from the bizarre and inaccurate, to the most up-to-date information available from scientific bodies, newspapers and academic journals.

There is now good evidence that such services are valuable and in constant use. The OncoLink information resource, for example, provides oncologists with up to date trial and treatment information, as well as acting as an educational resource for cancer patients and their families. OncoLink was reportedly accessed 36,000 times in March 1994. The figure for November 1995 was 7,51,261 accesses.

Perhaps just as important as the ability to view information, the Web provides methods for anyone to publish information, and make it immediately available across the globe. This is achieved through a simple set of document standards that allow users to create electronic documents using text, image and video. The quality of these documents is now so high that the Web is used, for example, by several medical educational institutions. But, however, there are some problems with quality and other related issues what have been discussed in chapter 4.

- **Artificial Intelligence in Medicine (AIM)**

Although telemedicine, Internet and World-Wide-Web help (i) acquire knowledge, (ii) advice through communication channels etc., but, however, it is required to have in-house expertise for routine paediatric care management. In this respect, AI and expert system technology can contribute to the paediatric domain.

In the year 1984, Clancey and Shortliffe [13] provided the following definition of AIM :

> 'Medical artificial intelligence is primarily concerned with the construction of AI programs that perform diagnosis and make therapy recommendations. Unlike medical applications based on other programming methods, such as purely statistical and probabilistic methods, medical AI programs are based on symbolic models of disease entities and their relationship to patient factors and clinical manifestations'.

But, however, today the above definition would be considered narrow in vision and scope. At per the state of the art AIM has already contributed much more than defined by Clancey and Shortliffe.

AIM systems are by and large intended to support health care workers in the normal course of their duties, assisting with tasks that rely on the manipulation of data and knowledge. An AI system could be running within an electronic medical record system, for example, and alert a clinician when it detects a contraindication to a planned treatment. It could also alert the clinician when it detected patterns in clinical data that suggested significant changes in a patient's condition.

Expert or knowledge-based systems are the commonest type of AIM system in routine clinical use. They contain medical knowledge, usually about a very specifically defined task, and are able to reason with data from individual patients to come up with reasoned conclusions.

There are many different types of clinical task to which expert systems can be applied :

- **Diagnostic assistance**

When a patient's case is complex, rare or the person making the diagnosis is simply inexperienced, an expert system can help come up with likely diagnoses based on patient data.

- **Therapy critiquing and planning**

Systems can either look for inconsistencies, errors and omissions in an existing treatment plan, or can be used to formulate a treatment based upon a patient's specific condition and accepted treatment guidelines.

- **Generating alerts and reminders**

In so-called real-time situations, an expert system attached to a monitor can warn of changes in a patient's condition. In less acute circumstances, it might scan laboratory test results or drug orders and send reminders or warnings through an e-mail system.

- **Agents for information retrieval**

Software 'agents' can be sent to search for and retrieve information, for example on the Internet, that is considered relevant to a particular problem. The agent contains knowledge about its user's preferences and needs, and may also need to have medical knowledge to be able to assess the importance and utility of what it finds.

- **Image recognition and interpretation**

Many medical images can now be automatically interpreted, from simple X-rays through to more complex images like angiograms, CT and MRI scans. This is of value in mass-screenings, for example, when the system can flag potentially abnormal images for detailed human attention.

Although, some of us had a question in mind whether the AI systems should come in clinical use or not. Now, today, there is good evidence that expert systems are working well (table 3.3 in chapter 3) as routine clinical use. Efforts are needed, however, to increase the number of such systems.

1.3. Usage and benefits

Before the development of an expert system starts, the issues related to its usage and benefits should be thought for its successful use. Various aspects of the usage issues are particularly important such as (i) benefits of usage of the system to different potential users; (ii) who will use it and how; (iii) what are / will be the expected reactions of various people in using such a system; (iv) what problems users might face and how to overcome or minimise such problems; and (v) how the question-answer sequence should be tailored.

1.3.1. Levels of usage of a technical artifact

Gillies [14,15] suggests that there are three levels of human computer interaction : physical, task and organisational. The physical level concerns the user interface. The task level, the business tasks that the primary user undertakes while at the user interface. The organisational level concerns the organisational purpose of the tasks. This suggests that benefits which stem from such interaction can also be separated into three types, corresponding to three levels. Thus, for instance, speedier access to functionality would be a benefit of an improved user interface; speedier obtaining of a budget estimate would be a benefit at the task level; and speedier completion of the project proposal, of which a budget estimate is one part, would be a benefit at the organisational level.

Hart [16] discerns three levels at which we can discuss and describe the usage of a technological artifact; these are similar, but not identical, to those of Gillies. He gives the example of using a telephone. At the lowest level, we talk in terms of movements of hand, arm and finger which describe the picking up of the phone and dialing. These are components of the entities engaged in the process and are charactersied by being unable to stand alone. Consideration of the size of people's fingers and of buttons is validly carried out at this level. At the next level we talk in terms of phoning a certain person. We talk in terms of actions which are carried out by or on the entities as a whole - the person, the telephone. But, at this level the actions, while whole, have little meaning in themselves. One does not normally phone a person just for the sake of using the phone. This brings us to the top level, which gives meaning to the action of using the phone. At this level we talk about what we are trying to achieve by using the phone - the maintenance of a relationship, for example.

Let us now understand the difference between the three levels. At the first level we are concerned with technological features, with components that do not stand alone. It is at this level at which we are concerned with the design of such features or components. At the middle level we are concerned with whole entities rather than components and the actions or tasks they carry out. We are usually concerned with the intentional subject. At the top level we are concerned with the role of the entities rather than their actions or tasks. Tasks are reasonably well defined, even though there may be variants among the components of which they are made. They have little meaning in themselves and are carried out in fulfilment of roles, which give them their meaning. Hart characterised the difference between tasks and roles by stating that tasks could be entered in a diary while roles (he gave the example 'Love thy neighbour') could not. Hart maintains that the three levels are irreducible to each other.

The similarity between Hart's and Gillies' levels should be obvious, especially at the task level, but the differences are important. Hart's lowest level concerns any component

of the whole entity and, in the case of software, this means not just the user interface but also other features and facilities. While Gillies' highest level is differentiated from the task level by moving from the single user to the organisation, Hart's is differentiated by the addition of meaning or purpose. While it is often the case that the organisation does supply the purpose of tasks so that there is indeed a strong correlation between the two top levels, Hart's top level is not restricted to consideration of organisations. Hart's taxonomy seems more exhaustive and contains less ambiguity.

1.3.2. **Usage of expert systems**

One can now understand the use of an expert system using Hart's taxonomy. At the lowest level, we can talk about what we do with parts of the expert system, such as answering questions, using the explanation facility or using the help facility. It is at this level that discussion of the features of an expert system becomes important - what type of explanation it gives, which types of reasoning it employs, etc.

At the middle level we are concerned with the expert system as a whole, rather than its components and features. We are concerned with the individual users and which tasks the expert system supports. It is at this level that classification of expert tasks by Stefik [17] is largely relevant : prediction, diagnosis, planning etc. While consideration of features is largely technology centered, or involves only the primary user's actions at the user interface, consideration of tasks involves the primary user's actions within their immediate organisation.

At the top level, we are concerned with the role the users play in making use of the expert system and the purposes for which they carry out the tasks that are supported by it. In Basden [18] a list of roles was proposed which expert systems might fulfil : consultancy, checklist, program, communication, knowledge refinement, training and demonstration, in an attempt to say something about the application of expert systems. But, one can observe that this list was ill formed and is confusing, since it focuses on the roles the software plays rather than the roles of the human actors. Some of the list are true roles, if translated to the perspective of the human user. For example, by 'communication' it was meant for the clarification and spreading of expertise; and by 'consultancy' it was meant for enabling a person to solve a problem by the provision of expert advice. Both these are examples of roles. But by 'program' it was meant that the expert system shell could be used simply as a language in which to write programs and this is not the role : writing a program is a task.

It is a fact that an automated knowledge-based consultation system should be useful for improved health care. It might not be feasible to appoint one paediatrician for each rural health centre whereas the proposed system can be operated by a general medical practitioner. This happens to be the primary benefit of such a system

expected to mitigate such expertise, obviously to a restricted sense. Once, this PC-based consultation system is installed at remote health centres, the system is expected to assist the general physicians who are not expert in paediatrics. This system is also expected to assist the general physicians, not expert in paediatrics, working at sub-divisional / district hospitals or engaged in private practice. As a worst case, it may not be possible to install such a PC-based system at different rural health centres. In such a situation, this consultation system may be carried with a mobile medical unit having a personal computer (PC) along with the required storage cells or a generator unit for power supply. This mobile medical unit may, at least, be controlled by a general physician. Worthwhile to note here that medical students work with the expert guidance at various fields during their undergraduate / training courses. When appointed at a rural health centre, this system should certainly work as a companion of theirs. Even, the system may be used by a paediatrician looking for a second opinion. The overloading problem of an expert is expected to be relieved to some extent by the use of this automated system. It may also be used by the medical students as a 'training kit'. In all such situations, one may observe the increase of the level of accuracy and confidence during an interaction with a child under treatment. We also certainly expect a better patient management and treatment planning. It is now easy to observe the tangible benefits of the society as well as of rural people. In most of the situations, the rural people need not come to distant-located hospitals, which should save time, money and harassment.

From the above discussions, it is evident that the common users of the system will be the medical professionals who may not have any exposure to AI and expert systems technology. Even, they may not have any basic training on computers and computations. However, such types of introductory ideas on computers and computations are now being introduced in school levels in different institutions. Possibly, the by-product of the above situations is the lack of motivation and / or a kind of inertia to maintain the traditional systems. The situation should certainly be improved by the co-ordinated efforts from some expert groups arranging seminars, special talks for such users. This curriculum should contain the basics of computers and computations; what is AI and expert systems technology - the usefulness and limitations of this modern technology; and lastly, an overview of the present system. Hopefully, one-week intensive course would be adequate. In the initial phase of such use, an easy developer-user interaction should certainly be encouraging to minimise such problems.

Let us now have a look on the reactions of various people in using such a system. Various people include : government policy, doctors, patients, and parents/ guardians. It should be noted here that different funding agencies of Govt. of India encourages research projects for the development of such automated systems using AI and expert systems technology under human resource development schemes [19]. Doctors are

also taking interests in developing and using such automated systems within the confines of the present status of AI and expert systems technology in collaboration with computer professionals. Paediatric patients may not have any adverse impact on this. Rather, they may be attracted by the colourful screen of the computer what may assist a doctor to manage a child during examinations. Usually, parents / guardians are motivated by the advice from doctors. If the doctors are really motivated in using the system, there should really be no such adverse motivation build-up. Currently, there are good evidences that doctors and bio-medical researchers are actively participating with computer professionals in such projects.

The question-answer sequence should be tailored : (i) to the needs of the region, (ii) to the users' level, and (iii) to the nature of the paediatric problem domain. If the users were para-medical staffs, obviously, question-answer sequence should not have any knowledge from the intimate areas of paediatrics. On the contrary, as the users are medical professionls having basic degree of medical science or undergoing such type of course, the sequence should be as per the paediatric norms. The question-answer sequence should certainly be governed by the age-group property : Neonates, Infants, Pre-school, School-going, and Adolescence. Few examples are given below.

Case A : **Neonates**

During this period, the importance of antenatal, intranatal and immediate postnatal history are very significant along with the family history and the history of maternal diseases. The congenital deformities are also common at this age group. These issues are to be taken carefully during question-answer sequence.

Case B : **Infants**

Milestones and immunization history along with the birth history are important at this age group. Anthropometric measurements are also very significant during this period.

Case C : **Adolescence**

Psychological and behavioural disorders should be taken care of during the question-answer sequence.

1.3.3. **Benefits**

If there are three levels at which usage of an expert system can be discussed then it is not surprising if there are three levels of benefits, each pertaining to a different usage level. Basden proposed three levels at which benefits accrue [20] :

a) feature benefits:

b) task benefits;
c) role benefits.

Feature benefits are those advantages that arise from technological features of functionality and user interface. Ease of manipulation can arise, for instance, from graphical user interfaces. An example found in DTI [21] is 'critical items are highlighted'. Task benefits are those which arise from using the expert system to support a task. Examples in DTI [21] include 'fewer changes to schedule' and 'improved visibility of requirements'. Role benefits arise from the effect the expert system has on the roles the user fulfils by carrying out the supported tasks, such as 'improved supplier relations', which in DTI [21] is seen to arise from the two task benefits above.

While it is useful to be able to classify benefits into three sets, there is indication of some form of causal, or at least enabling, link between them. Feature benefits can lead to task level benefits, which can in turn lead to role level benefits as shown in fig.1.6.

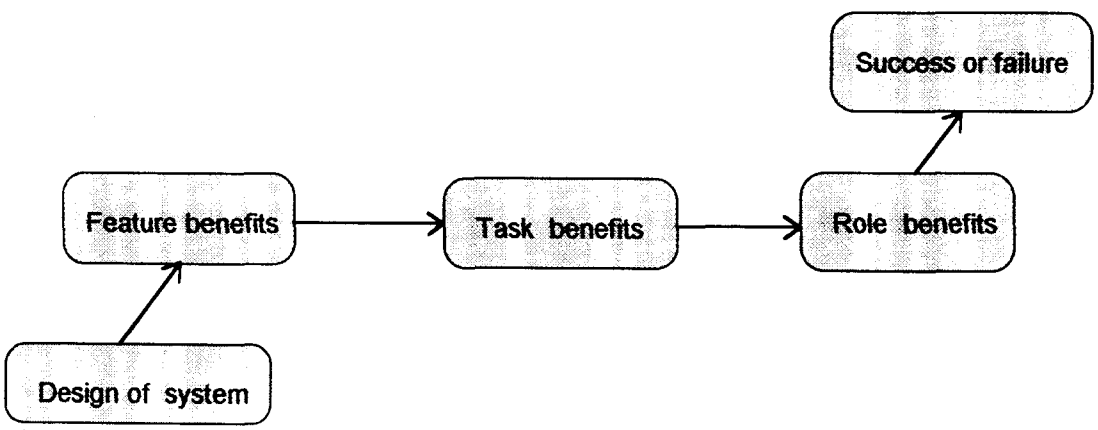


Fig.1.6 Three levels of benefit

It is role benefits, rather than feature and task benefits, that are a determiner of the success or failure of a system since, as discussed above, tasks and features find their meaning only in the context of a role.

It is now easy to view the expected benefits of our proposed system at per the tripartite division : feature benefits, task benefits, and role benefits as shown in fig.1.7.

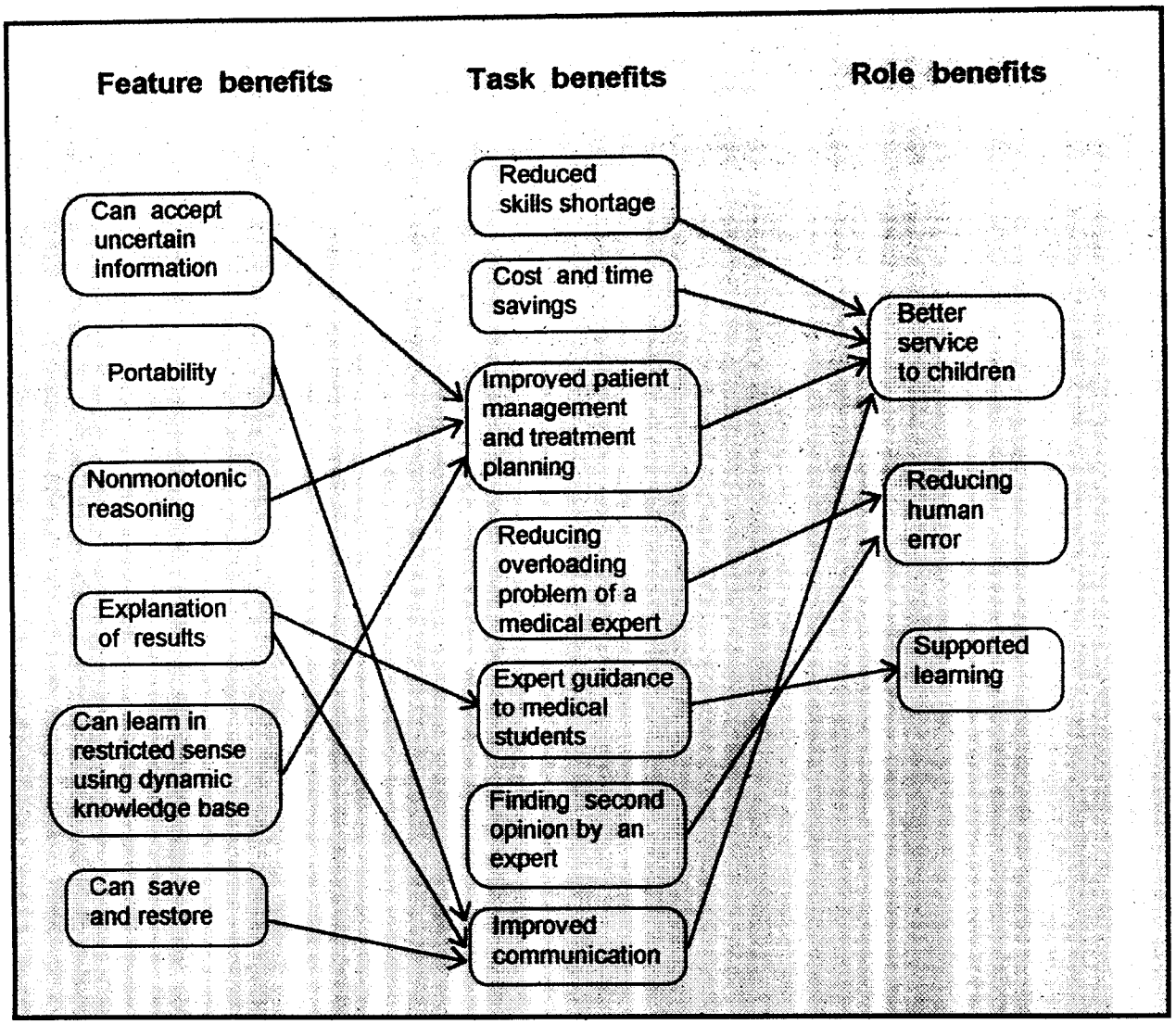


Fig.1.7. The levels of expected benefits in use of our proposed system

From the above fig. 1.7 it is seen that the links are many-to-many rather than one-to-one. Thus, a feature benefit can enhance a number of tasks and a given task may require several feature benefits. Moreover, though there is some causality of the links, while others are merely enabling. Further, there are other factors which influence the operation of each link, such as working practices, organizational norms and attitudes of users or others around them.

1.4. **Aim of the work**

The aim of the present work is to develop a knowledge-based consultation system for paediatric health care management. This we intend to do in steps. We initially have tried to explore the status of the domain which helped us in understanding the needs of the domain. To meet up the needs of the domain, atleast partially, an attempt has been

made to develop an expert or knowledge based consultation system. Emphasis has been on different potential issues connected to an expert system development. Potential issues such as (i) knowledge acquisition and representation, (ii) uncertainty management, (iii) expert system tool selection, (iv) user interface, (v) performance evaluation and (vi) usage and benefits should be mentioned here.

The process of building an expert system is inherently experimental [22]. The applicability of different potential issues pointed out above has been studied in different steps of the development following the prototyping approach of system development. Moreover, expert system development is the first and foremost software engineering [23] and therefore, there has been an attempt in the study to consider some software engineering issues during the development (e. g. phase refinement vs. prototyping).

1.5. **Summary of the work**

The scheme of presentation is as follows :

A. Chapter 2 of this thesis contains a brief survey on different aspects such as low birth weight, mortality and morbidity of the paediatric domain. There has been an attempt to include global as well as Indian scenario.

B. Chapter 3 contains two important issues : (i) why it is an expert system domain; and (ii) what requirements the domain lays on an expert system. This chapter initially discusses, in brief, what is artificial intelligence and expert systems, categories and application areas of expert systems, components of a typical expert system, general desirable features of an expert system, different stages of an expert system development, rule based vs. model based system etc. and lastly, the above mentioned two important issues (i) and (ii) have been presented.

C. In chapter 4, we have presented two issues (i) knowledge acquisition and (ii) knowledge representation. Under the first issue, levels of knowledge and knowledge categories have been discussed. Next, we have explored different sources of knowledge acquisition - their merits and demerits. Here we have also pointed out the prime sources of knowledge acquisition used in the current research. Under the second issue, we have discussed different schemes for knowledge representation - their merits and demerits. We have also discussed the relative suitability of the methods in context to the present problem domain.

D. Chapter 5 contains the results of our sonographic studies on foetal growth performance of North Bengal districts of India. As knowledge acquisition source, this study was conducted. A comparison has also been made among the results of this study, one other Indian study and the results of different Western studies.

E. Chapter 6 deals with the issue of selection of an appropriate expert system tool. A detailed study has been provided here among three types of tools : AI-languages, tool kits and shells.

F. Chapter 7 deals with the issue of uncertainty management. The sources and nature of inexactness have been identified and discussed with examples. A suitability analysis of different methods of handling inexactness which seem(s) to be most sympathetic to the problem domain at our hand has been provided.

G. Chapter 8 presents the issue of prototyping development; a software engineering issue. Phase refinement vs. prototyping has been presented. Lastly, our first prototype system is presented.

H. In chapter 9, we have presented the ideas on a research direction incorporating an outline of some fuzzy concepts in paediatrics in order to design a powerful expert system which needs to take into account some fuzzy concepts along with other types of inexactness of knowledge. Here we have presented (i) prototype 2.0 based on some linguistic articulations and (ii) a fuzzy knowledge-based neonatal resuscitation management (prototype 3.0) with performance evaluation.

I. Chapter 10 presents an integrated prototype system (KID) for the domain (prototype 4.0). Performance evaluation of the system has been presented. Object-Oriented analysis, design and implementation have been incorporated. The graphical user interface (GUI) issue has also been discussed.

J. Chapter 11 has been devoted to a case study on neonatal resuscitation management incorporating a comparison of case based reasoning with rule-based reasoning (prototype 5.0). Performance evaluation has also been presented.

References

1. K. Park. Park's Text book of preventive and social medicine. 15th ed. Banarsidas Bhanot, Jabalpur; India, 1997, 365-366.

2. K. Park. Park's Text book of preventive and social medicine. 15th ed. Banarsidas Bhanot, Jabalpur; India, 1997, 468-541.

3. Census of India. Govt. of India; 1991.

4. M. Chatterjee, A. K. Saha and R. K. Samanta. Factors affecting the growth performance of rural children of North-Bengal Terai region-2, bacteriological analysis of drinking water and a comparison with diarrhoeal disease. Modelling, Measurement & Control; C, vol.53, no.2, 1995, AMSE Press, France, 7-16.

5. M. Chatterjee, K. C. Mitra, A. K. Saha and R. K. Samanta. Foetal growth performance - a comparison between Indian and Western studies; Modelling, Measurement and Control; C, AMSE Press, France, vol.47, no.2, 1995, 23-44.

6. National Centre for Health Statistics, USA.

7. K. N. Agarwal D. K. Agarwal, D. G. Benakappa, S. M. Gupta, P. C. Khanduja, S. P. Khatua, K. Ramachandran, P. M. Udani and C. Gopalan. Growth performance of affluent Indian children (under fives). Nutrition Foundation of India; 1991,1-70.

8. M.Chatterjee, A. K. Saha and R. K. Samanta. Growth performance of rural children of Northern Region of West Bengal. Modelling, Measurement and Control; D, vol.10, no.4, 1994, AMSE Press, France,1-52.

9. E. Shortliffe, S. G. Axline, B. G. Buchanan, T. C. Merigan and S. N. Cohen. An artificial intelligence program to advise physicians regarding antimicrobial therapy. Computers and Bio-medical Research; vol.6,1973, 544-560

10. L. E. Rodewald. Baby : an expert system for patient monitoring in a newborn intensive care unit. M. S. thesis; Computer Science Dept., University of Illinois, Champaign-Urbana, 1984.

11. E. Coiera. Guide to Medical Informatics. The Internet and Telemedicine; 1997.

12. E. Coiera. Medical Informatics. BMJ 310; 1995, 1381-1386.

13. W. J. Clancey and E. H. Shortliffe (eds.). Readings in Medical Artificial Intelligence - The First Decade. Addison-Wesley; Reading MA, 1984.

14. A. C. Gillies. The integration of expert systems into main-stream software. Chapman and Hall; 1991.

15. A. C. Gillies. Software quality : Theory and management. Chapman and Hall; 1992.

16. H. Hart. Understanding our World : An Integral Ontology. University press of America; 1984.

17. M. Stefik, J. Aikins, R. Balzer, J. Benoit, L. Birnbaum, F. Hayes-Roth and E. Sacerdoti. The organisation of expert systems : A tutorial. Artificial Intelligence; vol. 18, no.2, 1982, 135-174.

18. A. Basden. On the application of expert systems. International Journal of Man-machine Studies; vol.19,1983, 461-477.

19. Directory of extramural research and development projects. Approved for funding by selected Central Govt. Agencies / Departments d uring 1994-95. Government of India, Ministry of Science & Technology, Department of Science & Technology, New Delhi, April 1996.

20. A. Basden. Three levels of benefits in expert systems. Expert systems; vol.11, no.2, May 1994, 99-107.

21. DTI. Manufacturing Intelligence : A decision makers' briefing. Applying knowledge based systems technology in the manufacturing & process industries, Department of Trade and Industry, London, 1991.

22. B. G. Buchanan, D. Barstow, R. Bechtal, J. Bennett, W. Clancey, C. Kulikowski, T. Mitchell, and D. A.Waterman. Constructing an expert system. In F. Hayes-Roth, D. A. Waterman and D. B. Lenat, (eds.). Building expert systems. Addison-Wesley; Mass, 1983, 127-167.

23. J. Rothenberg. Expert system tool evaluation. Topics in expert system design, ed., G. Guida and C.Tasso, Elsevier Science Publishers. B. V.; (North-Holland), 1989, 205-229.

THE PAEDIATRIC DOMAIN

2.1. Introduction

In any community, children constitute a priority group. In India, children under 15 years of age is about 34.8% of total population as per the Govt. of India report 1995 [1]. By virtue of their numbers, children are the major consumer of health services.

They not only constitute a large population group, but also are vulnerable or special risk group. The risk is related with growth, development, disease pattern and survival. From the commonly accepted indices, it is evident that mortality rates in the paediatric age group are higher than adult population especially in developing countries. Thus by improving the health of children, we contribute to the health of the general population. These cosiderations have led to the formulation of special health services for children all over the world.

In response to the acute needs of children throughout the world, the United Nations World Summit,1990, for children was convened. The objective of the summit was to establish national programmes of actions for achieving basic health and social goals. These goals include control of major childhood diseases, halving of child malnutrition, one-third reduction in under-fives mortality rate, halving of maternal mortality rate, provision of safe water to all communities, universal availability of family planning, and basic education for all children. The summit also urged upon the ratification of the convention on the rights of the children.

2.2. Statistical Importance

As stated above, statistically children constitute a priority group. This is further confirmed by the tables 2.1, 2.2, and 2.3 showing the age wise distribution.

Table 2.1. Distribution of population by age group (in 000's) in India [2].

Sl.No.	India / States	All Ages	0 - 4	5 - 9	10 -14
1	2	3	4	5	6
	INDIA	846303	108296	112047	99436
1.	Andhra Pradesh	66508	7448	8951	7776
2.	Assam	22414	3206	3247	2746
3.	Bihar	86374	12452	12955	10571
4.	Gujarat	41310	4887	5091	4809
5.	Haryana	16464	2321	2231	2021
6.	Karnataka	44977	5277	5657	5338
7.	Kerala	29099	2759	2851	3053
8.	Madhyapradesh	66181	9345	9064	7663
9.	Maharashtra	78937	9806	9683	8820
10.	Orissa	31660	3881	4122	3618
11.	Punjab	20282	2500	2340	2312
12.	Rajasthan	44006	6394	6394	5465
13.	Tamil Nadu	55859	5397	5964	6002
14.	Uttar Pradesh	139112	19708	20110	16991

Table 2.2. Age-wise percentage distribution of West Bengal, India[3].

Age groups	Urban	Rural	Total
0 - 4	2.24	9.25	11.49
5 - 9	2.84	10.43	13.27
10-14	3.02	8.3	11.85

Table 2.3. Distribution of children by age in the United States [4].

Age (year)	Number (Resident Population)		
< 1		4011	
1		3969	
2		3806	
3		3718	
4		3717	
	<5		19222
5		3702	
6		3681	
7		3575	
8		3512	
9		3767	
	5-9		18237
10		3703	
11		3662	
12		3484	
13		3414	
14		3409	
	10-14		17671
15		3293	
16		3362	
17		3360	
18		3383	
19		3808	
	15-19		17205
20		4080	
21		3969	
0-21	80384		

2.3. **Children in developing countries**

The total dependence of child on adults for survival and development puts them in an extremely vulnerable situation. In most societies, the parents or the close relatives look after the children. In many countries, some economically advanced and others with a socialistic pattern of government, the State plays a major role in child care ensuring that his basic health and education requirements and physical needs are met with.

Many developing countries, by and large, face some problems in this regard [5]. India, in view of its vast population, leads in numbers [6]. There are about 20 million children in organized labour work force, 30 million disabled children (2 million blind children) and more than 25 million street children. The number of those suffering from malnutrition and chronic diseases such as tuberculosis is extremely high. We have serious problems of destitute children, child prostitution and various other forms of child abuse and neglect and discrimination against the girl child. Adequate care and full opportunities for development (stimulatory environment, proper education, health care) are available largely to the small affluent section of the society.

With the alarming increase in population the factors like poverty, illiteracy and ignorance, poor state of village environment (where the majority of children live), and increasing urban slums are responsible for leading to a poor quality of life for the majority of our children. Child survival has been improved, as indicated by a decline in various indices of mortality, but / however, the necessary inputs towards various other needs of children have been grossly insufficient.

Besides, curative and preventive health care, comprehensive child care comprises various aspects of child development including a stimulatory environment, education and prevention of child abuse and exploitation. The later encompass a spectrum of cruelty to children from mild forms of neglect to employment in hazardous industries. Problems like child labour, sexual abuse and child prostitution are also very serious. These problems are multifactorial. Society as a whole has to think in these matters.

2.4. Age related health problems in India

The morbidity and mortality patterns vary from one age group to another. For example, congenital anomalies are common during early months of life, haemolytic diseases, hypoxic-ischemic-encephalopathy, aspiration syndromes, RDS are unique problems of neonates. Sudden infant death syndromes are common during infancy. Rotaviral diarrhoea are seen below two years of age. Infections and infestations are common during almost whole of the childhood. Personality disorders are the problems of adolescence. Age related common health problems are narrated in table 2.4.

2.5. Burden of the diseases in India

More than 90% of World's children are born each year in the developing world. Nearly 35,000 of them die each day most from common and preventable problems. Health and illness for these children are the result of a complex dynamics of environmental, social, and economic factors. India is also facing the similar problem.

Table 2.4. Age related health problems chart

Age group	Disease
Perinatal period	i) Still birth ii) Congenital disorder
Neonate	LBW, Respiratory distress syndrome, Complications of pregnancy, Haemolytic disease, Hypoxic ischemic encephalopathy, Congenital anomalies, Malformations, Aspiration syndromes, Metabolic disturbance of neonates.
Infant	Congenital anomalies, Sudden infant death syndromes, RTI, Diarrhoeal diseases, Infections and infestations, Accidents and adverse effects, Malnutrition.
Toddler and Pre-school	RTI, Diarrhoeal diseases, Infections and infestations, Accidents, Malnutrition, Behavioural problems : habit disorders.
School going	Parasitic Infestations, Deficiency diseases, RTI, Infectious diseases, Malnutrition, Diarrhoea, Injury and accident, Behavioural disorder : a) Personality disorders, b) Psychosomatic disorders, Diseases of the skin, eyes, ears, teeth.

- **Acute Diarrhoeal Diseases**

Diarrhoea is a major health problem in developing countries. An estimated 1.3 billion episodes of diarrhoea occur each year and 3.2 million children under the age of 5 years die of diarrhoeal diseases. Again 80 percent of these deaths affect children under the age of two years. The incidence of diarrhoeal attack was calculated may be 6-12 episodes per child per year in most developing countries. The total diarrhoea morbidity for a given child may be as high as one third of its first two years of life. Overall children are ill with diarrhoea for 10-20 percent of their first 3 years of life [7]. In India, according to SRS estimates during the year 1992, the child mortality was about 26.6 per 1000 children under 5 years age, out of which 20% death were due to diarrhoeal diseases [8].

- **Respiratory Diseases**

Acute respiratory infection (ARI), particularly Pneumonia is one of the leading causes of health problem and accounted for an estimated 3.9 million death among children under the age of 5 years in the developing world during the year 1996 [9]. On an average, a child in both developed and developing countries has 5 to 8 attacks of ARI annually. The incidence and severity of Pneumonia are not spread evenly around the world. The incidence of Pneumonia is almost constant in children in developed countries, varying 3-4 percent a year. By contrast, the incidence in developing countries ranges between 10-20%.

In India, ARI is one of the major causes of death. Hospital records from states with high infant mortality rate show that upto 13% of inpatient deaths in Paediatric wards are due to ARI. In the community, this may be much higher as children may die at home [10].

- **Nutritional Diseases**

The specific nutritional problems in India are :

a) **Protein-Energy malnutrition** (PEM)
Protein-energy malnutrition has been identified as a major health and nutrition problem in India. It occur particularly in weaklings and children in the first year of life. It is characterized by low birth weight if the mother is malnourished, poor growth in children and high level of mortality in children between 12-24 months, and is estimated to be an underlying cause of 30% of deaths among children under age five [11].

b) **Nutritional anaemia**
India has probably the highest prevalence of nutritional anaemia in woman and children. About one-half of the non-pregnant women and children are estimated to suffer from Anaemia.

c) **Xeropthalmia**

About 0.04% of total blindness in India is attributed to nutritional defficiency of Vitamin A. Keratomalacia has been the major case of nutritional blindness in children usually between 1-3 years of age.

- **Infections and Infestations**

Infections and Infestations are quite prevalent in India, out of which water-borne diseases (e. g. acute diarrhoea, dysentery and enteric fever, viral hepatitis) lead all. Studies show that half of the world's tuberculosis patients are in India accounting from 14 million cases of which approximately 3.5 million are infectious cases [11]. Tetanus and Diphtheria are not yet under complete control. Among the parasitic diseases, Malaria and Kala-azar have staged a come back. Intestinal parasites such as ascariasis, hookworms, giardiasis and amoebiasis are widely prevalent.

2.6. Indicators and denominators in India

- **Mortality Rate in Infancy Childhood**

Mortality rates are good indicators to measure the level of health and health care in different countries. They also help in assessing the overall socio-economic development of a country and correlate well with certain economic variables such as GNP. Medical and social progress have substancially reduced mortality rate in childhood.

- **Perinatal Mortality Rate**

Perinatal mortality is one of the most sensitive indices of maternal and child health. The perinatal mortality rate is an indication of the extent of pregnancy wastage as well as the quality and quantity of health care available to the mother and the new born. Various factors are known to be associated with perinatal morbidity and mortality including socio-economic status, parity of the mother, quality and quantity of prenatal, intranatal and neonatal care. Provision of special care to mothers at risk is mandatory to bring about a reduction in preventable death rate.

About two-thirds of all perinatal deaths occur among infants having less than 2500 gm. birth weight. The causes involve one or more complications in the mother during pregnancy or labour, in the placenta or in the foetus or neonate. The main causes of death are intrauterine and birth asphyxia, low birth weight, birth trauma and intrauterine or neonatal infections.

Perinatal mortality rates in developed countries like USA, UK and developing country like India are shown in table 2.5. [12,13].

Table 2.5. Perinatal mortality rates.

INDIA			USA	UK
Rural	Urban	Combined		
49.1	32.7	46.0	10.0	10.0

- **Neonatal and post-neonatal mortality rate**

Neonatal mortality is a measure of the intensity with which **endogenous factors** (e.g., low birth weight, birth injuries) affect infant life. The neonatal mortality rate is directly related to the birth weight and gestational age; the lighter or more immature is the baby, the higher is the death rate. The higher concentration of infant deaths in the **early** neonatal period with **endogenous factors** causes, suggests the need to improve the antenatal and post-natal services to expectant mothers. However, neonatal mortality is the most difficult part, of infant mortality, to alter, because of the endogenous factors which are not sensitive to improvement in environmental conditions. Neonatal mortality is greater in boys throughout the world, because newborn boys are biologically more fragile than girls.

Deaths occuring from 28 days of life to under one year are called **post neonatal death**. The causes of neonatal and post neonatal mortality are given under table 2.6 [11]. Table 2.7 shows the status of perinatal and neonatal mortality of South-East Asia region.

Table 2.6. Causes of neonatal and post neonatal mortality

Neonatal mortality (0-4 weeks)	Post-neonatal mortality (1-12 months)
1. Low birth weight	1. Diarrhoeal diseases
2. Birth injury and difficult labour	2. Acute respiratory infections
3. Congenital anomalies	3. Other communicable diseases
4. Haemolytic diseases of newborn	4. Malnutrition.
5. Conditions of placenta and cord	5. Congenital anomalies
6. Diarrhoeal diseases	6. Accidents
7. Acute respiratory infections	
8. Tetanus	

Table 2.7. Perinatal and Neonatal mortality of the South-East Asia Region, 1995 [14].

Country	Number of Perinatal deaths (in thousands)	Perinatal mortality rate per 1000 births	Number of neonatal deaths total (in thousands)	Neonatal mortality rate (per 1000 live births)
India	1736	65	1328	50
Bangladesh	368	85	264	65
Bhutan	.....	100	.....	75
DPR Korea	.....	20	8	15
Indonesia	214	45	161	35
Maldives	< 1	45	< 1	30
Myanmar	86	55	53	35
Nepal	67	75	49	55
Sri Lanka	9	25	7	20
Thailand	26	20	23	20

- **Infant mortality rate (IMR)**

The infant mortality rate is one of the good overall indicators of health status. Infant mortality is given a separate treatment by demographers because (a) infant mortality is the largest single age-category of mortality, (b) deaths at this age are due to a peculiar sets of diseases and conditions to which the adult population is less exposed or less vulnerable, (c) infant mortality is affected rather quickly and directly by specific health programmes and hence may change rapidly than the general death rate.

There are wide variations between countries or regions in the rates of infant mortality. The world average of IMR for 1995 has been estimated to about 60 per 1000 live birth [15]. However, infant mortality rate varies from 6.9 per 1000 live births in the developed countries to 106.2 per 1000 live births in the least developed countries.The average in the developing countries was 66.6 per 1000 live births.

India is still among one of the countries having high infant mortality rate (74 per 1000 live births in the year 1996). Infant mortality rate has declined slowly from 204 during 1911-15 to 129 per 1000 live births in 1970 and remained at around 127 for many years, then declined a bit once again to114 in 1980 and to 110 in 1982, coming down to 74 in the year 1996. Despite this significant decline, the rates are high as compared to developed countries (4-9 per 1000 live births) as shown in tables 2.8. Trends in infant mortality in South-East Asia region are shown in table 2.9.

Table 2.8. Infant mortality rate in selected countries [16-18]

Country	1900	1950	1985	1996
India	232	127	95	74
Bangladesh	...	159	133	100
Sri Lanka	...	77	38	15
Sweeden	96	22	6	5
Switzerland	139	32	8	6
France	149	53	8	7
New Zealand	75	23	11	9
USA	162	33	11	7
UK	145	33	9	7
Japan	151	60	6	4

Table 2.9. Trends in Infant mortality in South-East Asia region [14].

Country	Estimated infant mortality rate (per 1000 live births)						
	1965-1970	1970-1975	1975-1980	1980-1985	1985-1990	1990-1995	1995-2000
Bangladesh	140	140	137	128	110	91	78
Bhutan	189	178	165	150	131	117	104
DPR Korea	63	50	38	32	28	24	22
India	145	132	129	106	93	78	72
Indonesia	124	114	105	90	75	58	48
Maldives	136	121	106	94	82	60	49
Myanmar	136	122	114	106	103	90	78
Nepal	175	160	142	125	109	96	82
Sri Lanka	61	56	44	35	24	18	15
Thailand	84	65	56	44	39	32	30

- **Under-five mortality rate**

Under-five mortality refers to deaths occuring after birth but before five years of age. It is often used as an approximation for the probability of dying between the exact ages of 0 and 5 years. Another commonly used indicator is the child mortality rate which approximates the probability of dying between the exact ages of 1 and 5 years.

A large part of the reduction in under- five mortality is accounted for by reductions in infant mortality. The successful implementation of immunization programmes in all countries resulted in a dramatic decline in infant and child deaths due to the six vaccine - preventable diseases. The implementation of diarrhoeal diseases and acute respiratory infection control programmes, improved sanitation, access to clean water,

and better accessibility to health services have been other factors leading to the decline in under-five mortality.

The global average for under-five mortality in 1996 was 81 per 1000 live births. In the developed countries, the rate was 8 per 1000 live births. In the developing countries, it was 89 per 1000 live births; in the least developed countries it was 153 per 1000 live births - 18 times higher than the rate of an industrialized nation. Child mortality has declined significantly during the past 25 years, although the pace of decline has not been uniform among the children of different age groups in different countries. Globally, child mortality declined from 134 per 1000 live births in 1970 to about 81 in 1996, i.e., about 40% during the period 1970 - 1996. Table 2.10 shows child mortality rate of some selected countries.

Table 2.10. Under - five mortality rate in some selected countries during 1960, 1980 and 1996 (per 1000 live birth) [16, 19].

Country	1960	1980	1996
India	236	177	99
Srilanka	130	52	19
Thailand	146	61	43
China	209	65	43
Bangladesh	247	211	144
UK	27	14	8
USA	30	15	9
Japan	40	11	6
Singapore	40	13	9

However, most of the reduction in under-five mortality is due to decline in infant mortality. This reduction was early due to a 35% drop in vaccine preventable deaths diseases as well as 13% fewer deaths from acute repiratory infection and 10% fewer deaths from diarrhoea [19]. The estimated mortality among children under-five by cause of death in the developing countries during 1985, 1990 and 1993 are shown in table 2.11.

2.7. **Low birth weight (LBW)**

Low birth weight is the major significant indicator of the chance of survival of a baby and its growth and development. It is thus an important guide to the level of care needed for individual baby. A preterm baby whose gestational period is less than 37 weeks, is physiologically immature and is at a high risk of dying during the neonatal period. Low birth weight babies are at higher risk of death and severe morbidity than full term and full sized infants; not only during the neonatal period but also during infancy and childhood.

Table 2.11. Estimated mortality among children under-five years by cause of death in the developing countries(in 1000)[20].

Cause of death	1985	1990	1993
Acute respiratory infections	4730	4200	4110
Neonatal and perinatal causes	3640	3750	3715
Diarrhoea alone	2955	2870	2740
Neonatal tetanus	790	275	290
Tuberculosis	295	295	280
Measles	495	275	290
Malaria	740	995	940
Pertusis	590	375	360
Malnutrition	-	250	190
HIV related	-	75	100
Accidents	195	195	170

It is estimated that low birth weight infants have a three or four times greater risk of dying from diarrhoeal diseases and acute respiratory infections. Owing to the frequent attacks of infections, by the age of five years they are more likely to be stunted.There is also evidence that there might be a link between low birth weight and cardiovascular disease, hypertension and diabetes in adult life.

The weight of the infant at birth in relation to its gestational age portrays, to some extent, the effects of intrauterine environment and maternal factors such as nutrition, toxaemia, chronic illness, etc. The birth weight of the baby is the most crucial determinant of its chances of survival and freedom from morbidity.

By International agreement low birth weight has been defined as a birth weight of less than 2.5 kg (up to and including 2499 gm); the measurement being taken preferably within the first hour of life, before significant postnatal weight loss has occured [21]. Apart from birth weight, babies can also be classified into 3 groups according to gestational age, using the words '**pre-term**', '**term**' and '**post-term**' as follows :

Pre-term : Babies born before the end of 37 weeks of gestation (less than 259 days);
Term : Babies born from 37 completed weeks to less than 42 completed weeks (259 to 293 days) of gestation.
Post-term : Babies born at 42 completed weeks or any time thereafter (294 days and over) of gestation.

A LBW infant then, is any infant with a birth weight of less than 2.5 kg regardless of gestational age.

2.7.1. **Incidence of low birth weight(LBW)**

World-wide, some 25 million of the 142 million infants born in 1990 had low birth weight. Nineteen million of the LBW infants were born in developing countries. Globally, this means, about 1 in every 6 infants has a LBW, but the incidence is not evenly spread around the globe. It ranges from about 4 per cent in the most developed countries to almost 50 per cent in some of the least developed countries as shown in the table 2.12. For the world as a whole, the average for the year 1990 has been estimated at around 17 per cent [20]. The trends of low birth weight values of South-East Asia region [22] are shown in table 2.13 and graphical representation have been shown in fig. 2.1.

Table 2.12. Incidence of LBW babies.

Countries	Percentage of live births
India	33
Sri Lanka	25
Thailand	13
China	9
Bangladesh	50
USA	7
Netherlands	4
Sweden	5
UK	7
Switzerland	5

Table 2.13. Trends in Low birth weight of South-East Asia Region [22].

Country	Percentage of infants with low birth weight			
	1980-1988	1990	1990-1994	1994-1996
India	30	33	33	30.0
Bangladesh	47	50	50	50.0
Bhutan	...	...	...	16.0
DPR Korea	...	...	...	1.0
Indonesia	14	14	14	11.0
Maldives	...	...	...	20.0
Myanmar	16	16	16	...
Nepal	...	...	...	23.2
Sri Lanka	25	25	25	18.0
Thailand	12	13	13	7.25

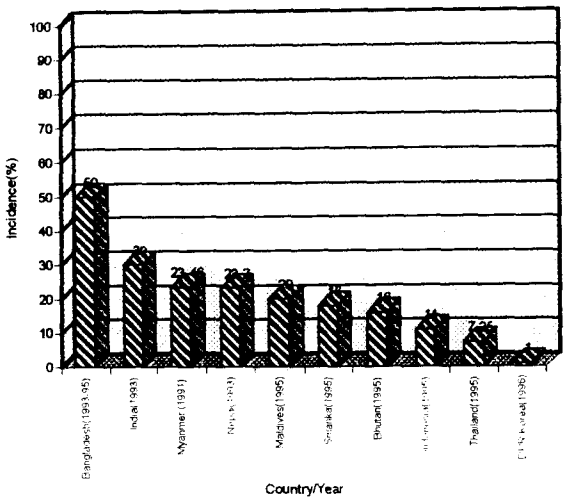


Fig. 2.1. Trends in low birth weight.

The gravity of the problem has now been universally recognized and at the 34th World Health Assembly, the Member States of WHO adopted, as part of the global strategy for health for all by the year 2000, the proportion of infants born with a LBW as one of the global indicators with which to monitor progress. The target is reduction in the incidence of LBW to less than 10 per cent.

Infants who weigh less than 2.5 kg at birth represent about 33 per cent of all live births in India [22]. More than half of these are born at term. The Government of India wishes to control this problem and decrease the incidence to 10 per cent by the year 2000 along with the strategies that had been developed to achieve "Health for All" by the year 2000. As birth weight is conditioned by the health and nutritional status of the mother, the percentage of infants born with a low birth weight closely reflects the health status of the communities in which they are born. The babies of healthy well nourished mothers weigh about 3.5 kg. If mothers are malnourished, it is 3 kg or less. In most parts of India, the mean birth weight is between 2.7 kg and 2.9 kg [23].

While the concept of LBW has the advantage of introducing uniformity, it carries the disadvantage that it does not take into consideration the genetic and environmental factors which also determine birth weight. Low birth weight, as defined by WHO, has less practical significance in countries like India, as most LBW infants are mature by gestation. For example, the mean birth weight of a mature Indian infant is about 500 gm less than that of the American infant. The criterion laid down by WHO where 2.5 kg is the dividing line between LBW baby and mature baby cannot be applied for Indian infants. It was assessed by Indian scientists by maturity, respiratory distress and feeding

problems that 2 kg or less should be taken as the criterion of LBW babies [24]. When this criterion was applied, the incidence of LBW babies having a birth weight of 2 kg was found to 5.5 per cent as against 25-30 per cent when the criterion was 2.5 kg and less.

2.7.2. **Importance of reducing low birth weight rates**

LBW is one of the most serious challenges in maternal and child health in both developed and developing countries. Its public health significance may be ascribed to numerous factors - its high incidence; its association with mental retardation and a high risk of perinatal and infant mortality and morbidity (half of all perinatal and one-third of all infant deaths are due to LBW); human wastage and suffering; the very high cost of special care and intensive care units and its association with socio-economic under development [25].

LBW is the single most important factor determining the survival chances of the child. Many of them die during their first year. The infant mortality rate is about 20 times greater for all LBW babies than for other babies. The lower the birth weight, the lower is the survival chance. Many of them become victims of protein-energy malnutrition and infection. LBW is thus an important guide to the level of care needed by individual babies. LBW also reflects inadequate nutrition and illhealth of the mother. There is a strong and significant positive correlation between maternal nutritional status and the length of pregnancy and birth weight. A high percentage of LBW therefore points to deficient health status of pregnant women, inadequate prenatal care and the need for improved care of the newborn.

2.8. **Socio-cultural and economical problems in India** [26]

The socio-cultural practices and the economic deprivation affect child's health more than adults. Few examples are given below :

- The common practice amongst the Indian families specially in rural areas is to feed the male earning members and head of the family maximum and the best, followed by male childrens, female childrens and lastly mothers. In maximum cases these mothers suffer from scarcity of food both in quantity and quality. This is even aggravated during poor economic conditions. However, the recommendation of ICMR nutritional needs for pregnant woman 300 extra calories during pregnancy. When it is not fulfilled the mothers suffer from anaemia and / or nutritional deficiencies. The resultant effects on foetus and neonates are as follows :

    a) Low birth weight, both preterm and / or small for gestational age(SGA);
    b) Poor brain growth;
    c) Poor storage of iron in neonates leading to anaemia during infancy.

- Certain beliefs and practices like offering honey, water, sugar water just after birth to neonates and not allowing to feed them colostrum leads to supression of breast milk in mothers and deprivation of good effects of colostrum on neonates. Colostrum contains high concentration of protein and other nutrients the body needs. It is also rich in anti-infective factors which protect the baby against respiratory infections and diarrhoeal diseases.

- The unhygienic ways of feeding neonates leads to Gastro-Intestinal infections. The cutting-off of umbilicus with unsterile blades, bamboo nail and application of cowdung may lead to Tetanus, Umbilical sepsis or septicaemia. The wide prevalent practice of hot fomentation with soots of oil lamps may lead to umbilical sepsis. Keeping the baby and mother in an unhygienic room may cause neonatal septicaemia. The practice of oil massage during early neonatal period in unhealthy way causes to abrations and skin infections.

- The wrong advices by the unwanted female advicers regarding formula feed, buffallo milk, goats milk, cows milk lead to supression of breast milk, and increase in the diarrhoeal diseases, malnutritions amongst infants.

- The mother's milk is sufficient for proper growth of infants upto 4 to 6 months of the age after which supplementary feeds should be started. But, due to poor practices or cultural belief babies receive nutritionally poor food in unhygienic way. This further leads to malnutrition and diarrhoeal disease.

- Other false belief i. e. eating rice in early age may lead to abdominal distention. Pulses and wheat cereals are not digested by the babies, and some fruits are too hot and some are too cold for the babies may lead to nutritional diseases.

- The cultural belief and practice may produce obstacles to the immunization programmes. Due to lack of immunization children suffer from communicable diseases like, Diphtheria,Tetanus, Measles, Whooping Cough etc.

- Poor maternal literacy rate leads to unregulated family size and is the cause of nutritional gap, trauma, accident, less medical care, deficiency etc.

2.9. **Status of Rural Health Centres in India**

With the above discussion it is evident that percentage of perinatal, neonatal, infant under-five,1-4 year age, mortality rates in Indian rural region is high. The reasons behind high mortality rate have been discussed in brief. Large number of paediatric populations are living in rural region. To mitigate these problems proper treatment planning is required. For proper treatment planning the basic requirement is paediatric experts. Actual picture of paediatric expert distribution in rural region is given in table

2.14 which is not at all satisfactory. To mitigate the scarcity of expertise, automated consultation systems within expert systems framework may be useful.

Table 2.14. Paediatricians working in rural areas of India [27].

Sl. no.	State/UT	PAEDIATRICIANS		
		S	P	V
1.	Andhra Pradesh	38	20	18
2.	Arunachal Pradesh	5	5	Nil
3.	Assam	INR	INR	INR
4.	Bihar	INR	INR	INR
5.	Goa	1	Nil	1
6.	Gujrat	29	8	21
7.	Haryana	60	3	57
8.	Himachal Pradesh	6	6	-
9.	J & K	4	1	3
10.	Karnataka	33	27	6
11.	Kerala	INR	INR	INR
12.	Madhya Pradesh	134	58	76
13.	Maharastra	28	25	3
14.	Manipur	2	2	Nil
15.	Meghalaya	INR	INR	INR
16.	Mizoram	1	Nil	1
17.	Nagaland	3	Nil	3
18.	Orissa	81	77	4
19.	Punjab	58	47	11
20.	Rajasthan	65	26	39
21.	Sikkim	9	1	8
22.	Tamil Nadu	INR	INR	INR
23.	Tripura	INR	INR	INR
24.	Uttar Pradesh	142	123	19
25.	West Bengal	80	11	69
26.	A & N Islands	Nil	Nil	Nil
27.	Chandigarh	1	1	Nil
28.	D & N Haveli	Nil	Nil	Nil
29.	Daman & Diu	Nil	Nil	Nil
30.	Delhi	1	Nil	1
31.	Lakshadweep	Nil	Nil	Nil
32.	Pondicherry	Nil	Nil	Nil

INR = Information not received
S = Number sanctioned,
P = Number in position
V = Vacant posts

References

1. Govt. of India. CSSM review: A Newsletter on the Child Survival and Safe Motherhood Programme; no.34, Oct. 1995.

2. Year Book. Family Welfare Programme in India. Department of Family Welfare, Ministry of Health & Family Welfare; Govt. of India, New Delhi, 1996-1997, 34.

3. Health on the March, State Bureau of Health Intelligence, Directorate of Health Services; Calcutta, West Bengal 1996-97, 17.

4. United States Bureau of the Census. 1991.

5. The State of World's Children. United Nations Children's Fund, Wallingford, Oxford University Press; 1995.

6. Statistics on Children in India. National Institute of Publication Co-operation and Child Development; New Delhi, 1995.

7. L. Mata. World Health; April 1986.

8. Govt. of India. Health information, DGHS; New Delhi, 1993.

9. WHO. The world health report, conquering suffering, enriching humanity; 1997.

10. Govt. of India. National Child Survival and Safe Motherhood Programme, Programme interventions MCH Division, Ministry of Health and family welfare, New delhi, 1994.

11. K. Park. Park's Text Book of Preventive and Social Medicine. Banarsidas Bhanot; Jabalpur, 15th edition, 1997, 23, 364-376.

12. WHO. The World Health Report. Report of the Director General;1996.

13. Govt. of India. CSSM review: A newsletter on the child survival and safe motherhood programme; no. 17, May 1994, New Delhi.

14. Health Situation in the South-East Asia Region, 1994-97, WHO, Regional Office for South-East, New Delhi, 54,1999.

15. WHO. The World Health Report. Report of the Director General; 1996.

16. WHO. The World Health Report 1997, Conquering suffering, enriching humanity, Report of the Director-General; 1997.

17. World Bank. World Development Report, Oxford University Press;1987.

18. C. M. Cipolla. The Economic History of World Population. A Pelican Original; 1974.

19. UNICEF. The State of Worlds Children, 1995.

20. WHO. Bridging the gaps, The World Health Report 1995, Report of the Director General, 1995.

21. WHO. International classification of Diseases, Tenth Revision, vol.2, 1993.

22. UNICEF, The State of the World's Children 1992, 1996 and 1998 and Country reports on the third evaluation of the implementation of HFA strategy, 1997.

23. Shanti Ghosh. The Feeding and Care of Infants and Young Children, UNICEF, New Delhi, 1976.

24. ICMR. ICMR Bull.; Oct, vol. 13, no. 10, 1983.

25. J. K. Harfouche. Bull WHO; vol.57, no.3, 1979, 387-403.

26. A. Parthasarathy (Ed. in chief). IAP Textbook of Peadiatrics. Jaypee; New Delhi, 1999, 150.

27. Bulletin on Rural Health Statistics in India, Rural Health Division, Directorate General of Health Services, Ministry of Health & Family Welfare, Govt. India, Nirman Bhavan, New Delhi, March 1995, 34.

ARTIFICIAL INTELLIGENCE, EXPERT SYSTEMS AND THE DOMAIN[+]

3.1. Introduction

Several computer systems have been built over the past few decades that can perform tasks which are comparable to many human mental activities, such as writing computer programmes, doing some mathematics or engaging in common sense reasoning or understanding natural languages or even driving automobile. There are also computer systems that can diagnose diseases, plan the synthesis of organic chemical compounds, solve differential equations in symbolic form, understand limited amounts of human speech and natural language text, analyse electronic circuits or write small computer programmes to meet some formal specifications - we shall say that such systems possess some degree of artificial intelligence (AI). AI systems are developed, undergo experimentation, and are improved. A second motivation for AI research is a large scientific goal of constructing an information processing theory of intelligence.

In its more than three decades history one area of AI that can claim a large measure of responsibility for the current AI awareness in the world is expert systems technology which are computer softwares that embody human expertise. Human expert in any field are a scarce commodity in our society, for example, a medical practitioner in some remote village may be competent but will be in great difficulty, when confronted by a patient with unfamiliar symptoms. If the specialist is not accessible the patient may have to settle for inadequate treatment. The scarcity of expertise exists in almost all fields, such as repairing automobiles, drilling for oil, managing a stock portfolio, or analysing chemicals. In all these and in many other cases there are times when access to the knowledge, experience, and judgement of an expert is an invaluable asset. One solution to the dilemma is the expert systems technology, which can help with new approaches to organisation, productivity, expertise, knowledge, competence, and smart automatic equipments that can act as intelligent assistance to human experts as well as assisting people who otherwise might not have access to expertise. It is different from data base programme that retrieves facts that are stored while an expert system uses reasoning to draw conclusion from stored facts.

The process of building an expert system is inherently experimental. In order to have successful development of an expert system in a domain, different potential issues have

[+] This is based on the publication [CSI Communications, May 1997, 15 - 21; ibid June 1997, 21 - 25] of the author.

to be fixed up which demands a thorough analysis. This chapter is meant for fixing up two potential issues, namely (i) why it is an expert system domain, and (ii) what requirements the domain lays on an expert system. These two potential issues should unfold some important matters relating to design and implementation.

In the next section, some potential issues such as categories and application areas of expert systems, trends of applications of expert systems, typical architecture of an expert system, desirable features of an expert system, different stages of an expert system development, types of expert systems have been discussed. In section 3.3, an attempt has been made to explain why the present problem domain may be considered suitable for an expert system domain. In section 3.4, we have tried to find out what requirements the domain lays on an expert system. A discussion has been provided at the end.

3.2. Expert systems technology

3.2.1. Categories and application areas of expert systems

We have mentioned that expert systems may be applied to any situation that normally requires human expertise. One can divide typical expert system applications into thirteen functional categories [1,2] shown in table 3.1. In table 3.2, we indicate application areas for which some expert system has been developed [2,3].

In Aerospace, we may cite here REX as an example of expert system shell. REX [4] an object-oriented, asynchronous real-time expert system shell to meet the challenges of the dynamic aerospace environment.

In Agriculture, different expert systems have been reported such as PLANT/CD, PLANT/DS, POMME. PLANT/CD [5], for example, predicts the damage to corn due to the black cutworm. The system uses a combination of rules and a set of black cutworm simulation programs to produce the predictions. Knowledge is represented as rules accessed by a backward chaining control mechanism.

In Business, we may cite here SUTA as an example of expert system. SUTA [6], an expert system called Soviet Union Trade Advisor (SUTA) was developed by Deloitte and Touche, a large management consulting (and CPA) company. The major objective of the system is to provide advice on trade opportunities and licensing requirements for medium to high-technology products.

Table 3.1. Generic categories of expert system applications

Category	Problem addressed and application types
Control	Governing overall system behavior for - air traffic control and battle management.
Debugging	Prescribing remedies for malfunctions for computer software.
Design	Configuring objects under constraints for circuit layout and CAD.
Diagnosis	Inferring system malfunctions from observables for medical and electronic fields.
Instruction	Diagnosing, debugging and repairing student behaviour
Interpretation	Inferring situation descriptions from sensor data for speech and image analysis and surveillance.
Planning	Designing actions - automatic programming and military planning.
Prediction	Inferring likely consequences of given situations for weather forecasting and crop estimation.
Prescription	Recommending solutions to system malfunctions.
Monitoring	Comparing observations to expected outcomes - for power plant and fiscal management.
Repair	Executing plans to administer prescribed remedies for automobiles / computers.
Selection	Identifying the best choice from a list of possibilities.
Simulation	Modeling the interaction between system components.

In Chemistry, different expert systems have been reported such as DENDRAL, CONGEN, CRYSALIS, C-13, GA1, META-DENDRAL, MOLGEN, OCSS, SECS, SEQ, SPEX, SYNCHEM, SYNCHEM2, TQMSTUNE. DENDRAL [7], for example, infers the molecular structure of unknown compounds from mass spectral and nuclear magnetic response data. Knowledge in DENDRAL is represented as procedural code for the molecular structure generator and as rules for the data-driven component and evaluator.

In Communications, we may cite here COMPASS as an example of expert system. COMPASS [8] (Central Office Maintenance Printout Analysis and Suggestion System) analyzes maintenance printouts of telephone company control switching equipment and suggests maintenance actions to be performed.

Table 3. 2. Application areas for expert systems.

Aerospace	Law
Agriculture	Manufacturing
Business	Mathematics
Chemistry	Medicine
Communications	Meteorology
Computer System	Military Science
Education	Mining
Electronics	Physics
Engineering	Process Control
Environment	Power Systems
Geology	Science
Image processing	Space Technology
Information Management	Transportation

In Computer Systems, different expert systems have been reported such as DART, CRIB, IDT, ISA, MIXER, R1-SOAR, TIMM / TUNER, XCON, XSEL, YES/MVS. DART [9], for example, assists in diagnosing faults in computer hardware systems using information about the design of the device being diagnosed.

In Education, we may cite here MIKE as an example of expert system. MIKE [10], developed by the Mandell Institute, is the automated admissions representative of Brandeis University. Built to expand the pool of high-quality applicants for the school, the system is designed to be used by high school seniors who are narrowing their choice of colleges. MIKE explains all the academic and extracurricular programs in which the prospective applicant has an interest.

In Electronics, different expert systems have been reported such as SYN, ACE, BDS, CADHELP, COMPASS, CRITTER, DAA, DFT, EL, EURISKO, FG502-TASP, FOREST, IN-ATE, MESSAGE TRACE ANALYZER, NDS, PALLADIO, PEACE, REDESIGN, SADD, SOPHIE, TALIB, TRANSISTOR SIZING SYSTEM. SYN [11], for example, assists engineers in synthesizing electrical circuits.

In Engineering, different expert systems have been reported such as SACON, CONPHYDE, DELTA, NPPC, REACTOR, SPERIL-I, SPERIL-II, STEAMER. SACON [12], for example, helps engineers determine analysis strategies for particular structural analysis problems. SACON is a backward chaining, rule-based system impliemented in EMYCIN.

In Environment, we may cite here DustPro, an expert system for controlling environment in mines. This has already been discussed in the mining application.

In Geology, different expert systems have been reported such as PROSPECTOR, DIPMETER ADVISOR, DRILLING ADVISOR, ELAS, HYDRO, LITHO, MUD. PROSPECTOR [13], for example, acts as a consultant to aid exploration geologists in their search for ore deposits. Given field data about a geological region, it estimates the likelihood of finding particular types of mineral deposits there. PROSPECTOR uses a combination of rule-based and semantic net formalism to encode its knowledge and bases its inferences on the use of certainty factors and the propagation of probabilities associated with the data.

In Image processing, we may cite here 3DPO [14] as an example of expert system. Image understanding is the process of establishing an interpretation of a scene based on one or more images of that particular scene.

In Information Management, different expert systems have been reported such as GCA, CARGUIDE, CODES, EDAAS, FOLIO, IR-NLI, PROJCON, RABBIT, RESEDA. GCA [15], for example, helps graduate students plan their computer science curriculum. The system gathers information about a student's academic history and interests and then acts as a faculty adviser by suggesting a schedule of courses for the student.

In Law, different expert systems have been reported such as LRS, AUDITOR, DSCAS, JUDITH, LDS, LEGAL ANALYSIS SYSTEM, LRS, SAL, SARA, TAXADVISOR, TAXMAN. LRS [16], for example, helps lawyers retrieve information about court decisions and legislation in the domain of negotiable instruments law, an area of commercial law that deals with checks and promissory notes.

In Manufacturing, different expert systems have been reported such as ISIS, IMACS, PTRANS. ISIS [17], for example, constructs factory job shop schedules. The system selects a sequence of operations needed to complete an order, determines start and end times, and assigns resources to each operation. ISIS uses a frame-based knowledge representation scheme together with rules for resolving conflicting constraints.

In Mathematics, different expert systems have been reported such as MACSYMA, MATHLAB 68. ADVISOR. MACSYMA [18], for example, performs symbolic manipulation of algebraic expressions and handles problems involving limit calculations, symbolic integration, solution of equations, canonical simplification, and pattern matching. The

system uses mathematical expertise organized as individual knowledge sources and chosen for a particular problem by sophisticated pattern-matching routines. MACSYMA achieves very high quality and efficient performance on the mathematical problems within its scope.

In Medicine, different expert systems have been reported such as BABY, AI/COAG, AI/MM, AI/RHEUM, ABEL, ANGY, ANNA, ARAMIS, ATTENDING, BLUE BOX, CASNET/GLAUCOMA, CENTAUR, CLOT, DIAGNOSER, DIALYSIS THERAPY ADVISOR, DIGITALIS ADVISOR, DRUG INTERACTION CRITIC, EEG ANALYSIS SYSTEM, EMERGE, EXAMINER, GALEN, GUIDON, HDDSS, HEADMED, HEART IMAGE INTERPRETER, HEME, HT-ATTENDING, INTERNIST-I/CADUCEUS, IRIS, MDX, MECS-AI, MEDICO, MED1, MI, MODIS, MYCIN, NEOMYCIN, NEUREX, NEUROLOGIST-I, OCULAR HERPES MODEL, ONCOCIN, PATHFINDER, PARTEC, PEC, PIP, PUFF, RADEX, RX, SPE, SYSTEM D, THYROID MODEL, VM, WHEEZE. BABY [19], for example, aids clinicians by monitoring patients in a newborn intensive care unit (NICU). BABY contains neonatology medical expertise for interpreting the clinical and demographic data. BABY is a forward chaining, rule-based system that uses rules embedded in a PROSPECTOR-like network. The system handles certainty by using a Bayesian probabilistic method similar to that used in PROSPECTOR. MYCIN [20] assists physicians in the selection of appropriate antimicrobial therapy for hospital patients with bacteremia, meningitis and cystitis infections. The system recommends drug treatment (type and dosage) according to procedures followed by physicians experienced in infectious disease therapy. MYCIN is a rule-based system employing a backward chaining control scheme. It includes mechanisms for performing certainty calculations and providing explanations of the system's reasoning process. At per the generic categories of expert system applications, some selected expert systems in medicine are shown in fig.3.1. Table 3.3 shows the systems that are being used in routine clinical use.

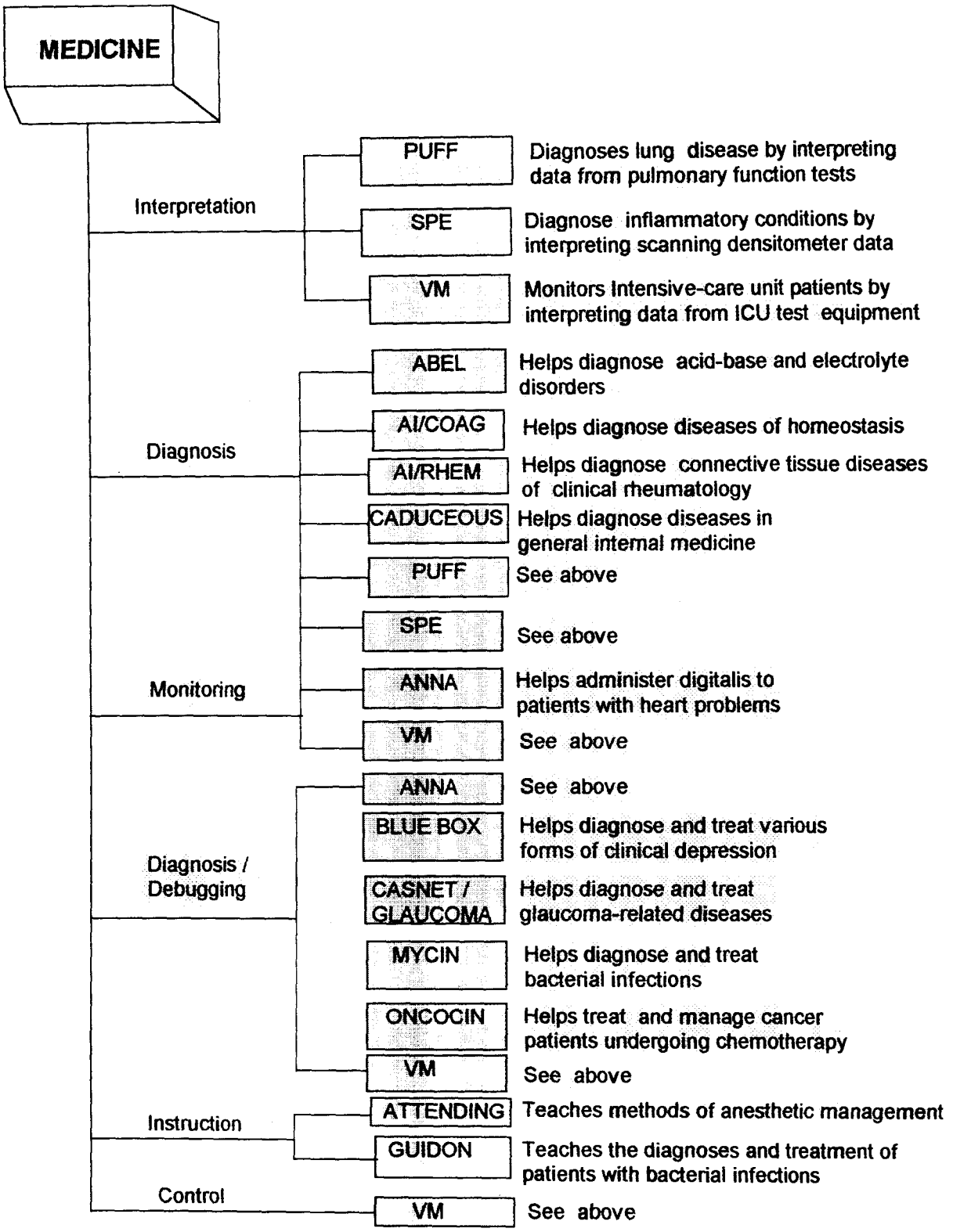


Fig. 3.1 Selected expert systems in medicine [3].

Table 3.3 Artificial Intelligent systems in routine clinical use(**http/www.coiera.com/ailist/list-idx.htm**)

Acute Care Systems

Name	Status	Type	Brief Description	Commissioned
ACRON	Decommissioned	Coronary care admission	Hybrid rule-based & Bayesian system for advising on management of chest pain patients in the emergency room.	1987
NeoGanesh	Routine use	Ventilator manager	Knowledge-based system for the management of mechanical ventilation in (ICUs).	
POEMS		Post-operative care	Decision support system for Post-operative care.	1992
SETH	Routine use	Clinical toxicology	Expert Systems in clinical toxicology.	April, 1992
VentEX	Under evaluation	Ventilator manager	VentEx is a knowledge-based decision-support and monitoring system applied in ventilator therapy.	
VIE-PNN	Under evaluation	Neonatal parentral nutrition	Expert system for composition of parenteral nutrition of neonates in (ICUs).	1993

Decision Support Systems

Name	Status	Type	Brief Description	Commissioned
DXplain	Routine use	Clinical decision support	A diagnostic decision support system in general medicine.	
Epileptologists' assistant	Decomissioned	Nurse progress note assistant	A cost effective expert system used by nurses to produce preliminary progress notes for physicians in epilepsy follow up clinic.	1989
Jeremiah	Routine use	Orthodontic treatment planner	A rule based / fuzzy logic system to provide dentists with orthodontic treatment plans for cases suitable for treatment by	1992

			general dental practitioners with a knowledge of removable orthodontic techniques.	
Help	Routine use	Knowledge-based HIS	Help is a complete knowledge based hospital information system.	1980
Iliad	Routine use	Clinical decision support		
MDDB	Routine use	Diagnosis of dysmorphic syndromes	Diagnosis of dysmorphic syndromes.	
Orthoplanner	Routine use	Orthodontic treatment planner	A knowledge based system to provide dentists with orthodontic treatment plans for cases where fixed orthodontic appliance techniques must be employed.	September 1994
RaPiD	Routine use	Designs removable partial dentures	Knowledge-based system for designing removable partial dentures.	1994
TxDENT	Routine use	Screening dental patients	An expert dental diagnostic screening and tracking system.	1997

Educational Systems

Name	Status	Type	Brief Description	Commissioned
Cancer, Me ??		Patient cancer advice		1989

Laboratory Systems

Name	Status	Type	Brief Description	Commissioned
Becton Dickinson Systems	Routine use	Haematology, Microbiology	1. QBC (TM) haematology analyser. 2. The Sceptor (TM) MIC interpretr.	
Coulter (R) FACULTY(TM)	Routine use	Haematology	Coulter FACULTY knowledge-Based system software functions as a	April 26, 1996

			consultant, assists workflow, and act as an educational tool in laboratory haematology.	
Dose Checker	Routine use	Drug Dose Checker	To assist the staff pharmacists at Barnes and Jewish Hospitals with monitoring drug orders for a set of drugs which must be carefully dosed for patients with possible renal impairment.	September 1994
Germ Alert	Routine use	Infection Control	To assist the infection control departments of Barnes and Jewish Hospitals with their infection control activities.	February 1993
Germwatcher	Routine use	Infection Control	To assist the infection control departments of Barnes and Jewish Hospitals with their infection control activities. These activities include surveillance of microbiology cultures data.	February 1993
Hepaxpert I, II	Routine use	Hepatitis serology	Automatic Interpretation of fetus for Hepatitis A & B.	1989
Interpretation of acid-base disorders	Routine use	Acid-base	Expert system for interpretation of acid-base disorders.	1989
Liporap	Routine use	Dyslipoproteinaemia phenotyping	Automatic Phenotyping of dyslipoproteinemia.	1987
Microbiology / Pharmacy Expert System	Routine use	Drug sensitivity	dBase based ES utilizing downloads of laboratory and Pharmacy data to detect patients whose antibiotic therapy is not consistant with pathogens detected by culture.	September 1991
Pro M. D.	Routine use	CSF interpretation		
PEIRS	Routine use	Pathology reports	Interpretative reporting of chemical pathology reports.	May 1991
PUFF	Routine use	Pulmonary function tests	The PUFF system diagnoses the results of pulmonary function tests.	1977
Sahm Alert	Routine use	Drug sensitivity	To assist the Microbiology laboratory at Barnes and Jewish	October 1995

			Hospitals with identifying organisms that have unusual patterns of antibiotic resistance.	

Quality Assurance and Administration

Name	Status	Type	Brief Description	Commissioned
ADE Monitor	Prototype	Adverse drug events	To assist the staff pharmacists at Barnes and Jewish Hospitals with monitoring patient clinical data for potential adverse drug events(ADEs).	June 1995
Apache III	Routine use	Clinical scoring system	Acute Physiology and Chronic Health Evaluation.	
Clinical Event Monitor	Routine use	Clinical alerts	Based on clinical events and a centralized patient database, the clinical event monitor generates alerts, interpretations, screening messages, etc. for health care providers throughout the medical centre.	March 1992
Colorado Medical Utilization Review System		Prescription quality review	An expert system which performs quality review of drug prescribing for Medical patients.	1990
Geriatric Discharge Planning System		Patient discharge planning		
Managed Second Surgical Opinion System		Managed care		1989
Reportable Diseases	Routine use	Infection control	To assist the infection control departments of Barnes and Jewish Hospitals with their infection control activities. These activities include surveillance of microbiology cultures data.	February 1995

Medical Imaging

Name	Status	Type	Brief Description	Commissioned
Perfex	Under evaluation	Cardiac SPECT interpretation	Expert system for automatic interpretation of Cardiac SPECT data.	
Phoenix	Decomissioned	Radiology consultant		
Thallium Diagnostic Workstation		Thallium myocardial	TWD learns to diagnose thallium myocardial scintigraphy from a training set of examples.	

In Meteorology, we may cite here WILLARD as an example of expert system. WILLARD [21] helps meteorologists forecast the likelihood of severe thunderstorms occurring in the central United States. The user may specify a particular geographical area for WILLARD to consider. The system characterizes the certainty of severe thunderstorm occurrence as "none", "approaching", "slight", "moderate", or "high", and each is given a numerical probability range. WILLARD's expertise is represented as rules generated automatically from examples of expert forecasting.

In Military Science, different expert systems have been reported such as MES, ACES, ADEPT, AIRID, AIRPLAN, AMUID, ANALYST, ASTA, ATR, BATTLE, DART, EPES, EXPERT NAVIGATOR, HANNIBAL, KNOBS, OCEAN SURVEILLANCE, RTC, RUBRIC, SCENARIO-AGENT, SIAP, SPAM, SWIRL, TATR and TWIRL. MES [22], for example, helps aircraft technicians diagnose aircraft problems. MES is a forward chaining, rule -based system.

In Mining, we may cite here DustPro as an example of expert system. DustPro [23] is a small rule-based system developed by the U.S. Bureau of Mines. It includes about 200 rules and was developed with a Level5 shell on a micro-computer. The system is now in operation in more than 200 mines. DustPro advises in three areas : control of methane gas emission, ventilation in continuous operations, and dust control for the mine's machines.

In Physics, different expert systems have been reported such as MECHO and GAMMA. GAMMA [24], for example, helps nuclear physicists identify the composition of unknown substances by interpreting gamma-ray activation spectra produced when the substance is bombarded with neutrons.

In Process Control, different expert systems have been reported such as FALCON and PDS. FALCON [25], for example, identifies probable causes of process disturbances in a chemical process plant by interpreting data consisting of numerical values from gauges and the status of alarms and switches. Knowledge is represented in two ways - as a set of rules controlled by forward chaining and as a causal model in network form.

In Power systems, we may cite here ENERGY MANAGEMENT as an instance of expert system. ENERGY MANAGEMENT [26] provides the utility needed a real-time ES integrated with a signaling system as well as with existing databases in hydroelectric plant. System capabilities include (1) detecting a disturbance when it happens, (2) locating the element of the fault, (3) tracking the location of the disturbance, and (4) recommending repairs (and justifying them).

In Science, different expert systems have been reported such as DENDRAL, MOLGEN. DENDRAL [7] for example, provides a rule-based program for identifying chemical compounds from laboratory data and performs this task better than chemists because it tirelessly considers all possible candidates - even those that a human expert might initially rule out as unlikely. Developed by Stanford University in the late 60's, DENDRAL is now widely used by industrial and academic researchers. Another Stanford program (MOLGEN) plans experiments for determining the coding sequences of DNA molecules. Its knowledge base encodes the DNA synthesis and analysis procedures of some of the world's leading genetic engineers.

In Space Technology, different expert systems have been reported such as ECESIS, FAITH, KNEECAP, LES, NAVEX, RBMS and RPMS. ECESIS [27], for example, provides autonomous control of an environmental control / life support sub-system (EC/LSS) for use aboard a manned space station. The system decides how to shift the modes of the various EC/LSS sub-systems during the transition from shadow to sun. It also monitors the EC/LSS, triggering actions in response to various events. Although ECESIS is intended to operate autonomously, it has a simple explanation capability to facilitate system demonstration. ECESIS has a bybrid architecture involving both rule-based and semantic net formalisms, and it uses the Bayesian scoring model developed for PROSPECTOR to handle uncertainty.

In Transportation, we may cite here CARGEX as an example of expert system. CARGEX [28] is an expert system constructed to assist in making loading decisions. The system contains approximately 300 rules that are described in about 6,000 lines of code. The basic goal of the application of CARGEX is increased productivity of the consolidation system.

Fig. 3.2 shows the major application areas that naturally developed - the breadth of the applications is remarkable. The fig. 3.2 also shows the number of developed systems for each area [2].

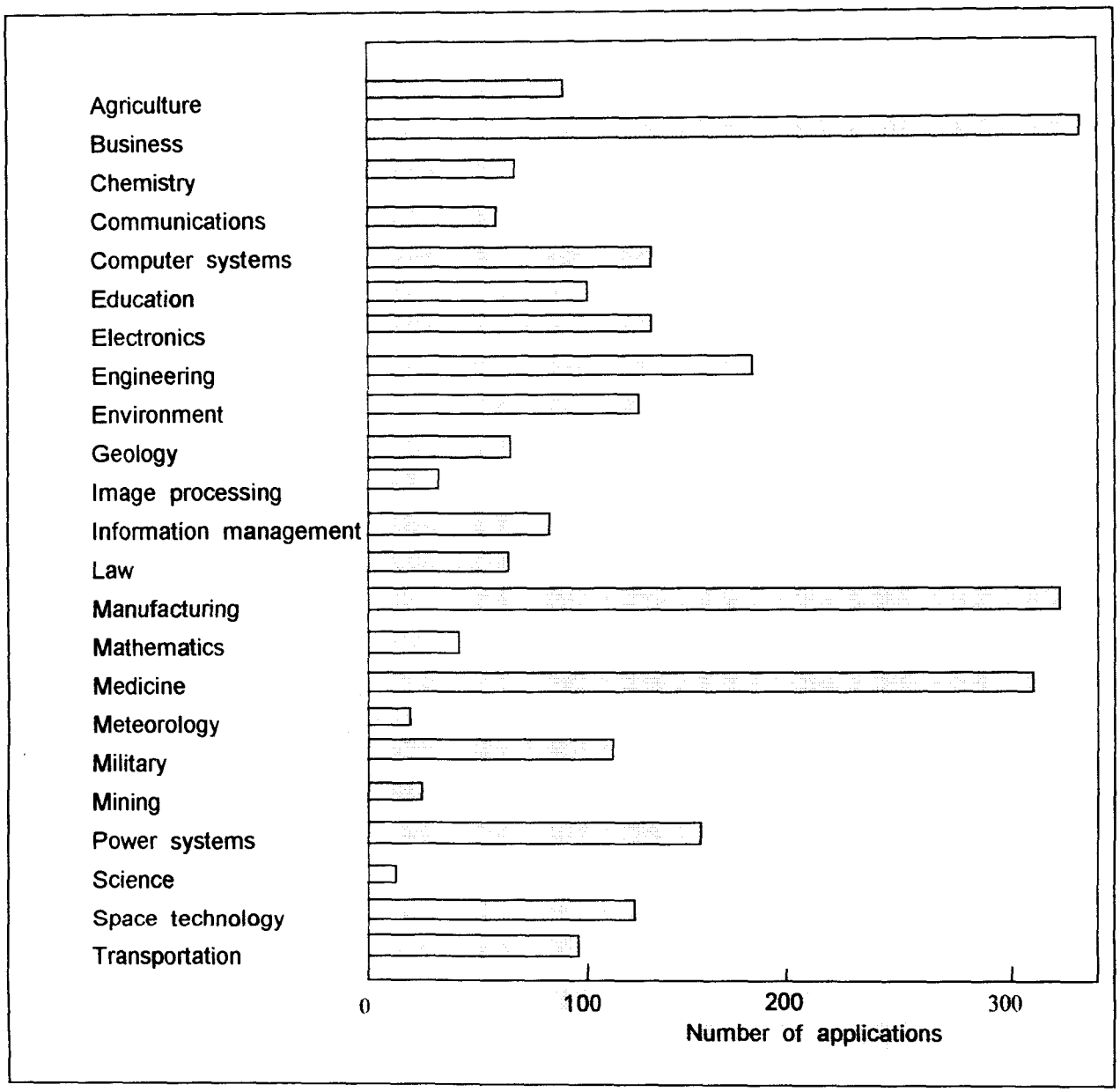


Fig. 3.2 The number of developed expert systems in various application areas.

Fig. 3.3 shows the percentage of applications for each problem type in table 3.1. Many applications employ more than one activity. For example, a diagnostic system might first interpret the available data, and later prescribe a remedy for the recognized fault.

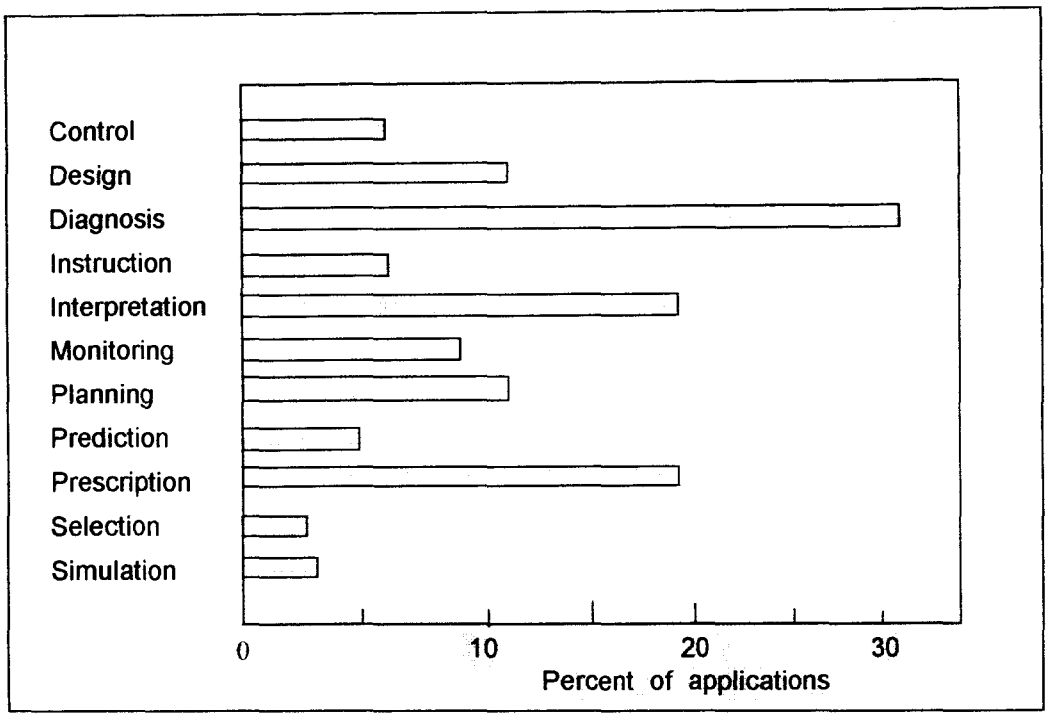


Fig. 3.3 The percentage of expert system applications by category.

The predominant role of expert systems has been the diagnosis as evident from fig. 3.3. One reason for the result is that this is the role most experts play. Fields such as medicine, engineering, and manufacturing have many individuals who help diagnose problems. Another reason for the large percentage of diagnostic systems is their relative ease of development. Most diagnostic problems have a finite list of possible solutions and a limited amount of information needed to reach a solution. These bounds provide an environment that is conducive to effective system design.

The drop-off from the large number of diagnostic applications to that of some other problem types is dramatic. Two reasons help explain this result. First, tasks such as design and planning are difficult to implement in an expert system framework because their steps vary greatly between application areas and it is often hard to precisely define these steps. Second, tasks such as instruction, control, and simulation, although they are excellent areas for expert system applications, are relatively new ventures [2].

3.2.2 **Trends of using expert systems**

Expert systems have primarily been used in business, manufacturing, and medicine as evident from fig. 3.2. Fig. 3.4 shows the number of developed expert systems per year in these areas. However, one may find in fig. 3.5 the dramatic increase in the number of developed expert systems during recent years [2].

Medical expert system applications dominated the scene in the early eighties. This is primarily due to the diagnostic nature of these applications and the relative ease of developing such systems. However, as we moved toward the mid-eighties, more difficult problems were approached. It was also time to develop systems that benefited the commercial sectors. Unfortunately, initial attempts frequently met with limited success. Three primary reasons help explain this result.

First, early applications of expert systems in industry often over-challenged the technology, leading to poor results. Many designers tried to build systems to solve problems that were beyond even the best experts. The thinking was "Well, we can't solve this problem, so let's try throwing AI at it". Second, other designers often took on a project whose scope was so broad that completing it in a reasonable time frame was impossible. Third, some designers developed remarkably intelligent systems; but failed to meet up the client's need to integrate the system into existing hardware and software. As a result, the powerful finished products were left on the shelves to collect dust.

With the few successes being produced during this period, coupled with earlier glowing promises of the technology, critics crept out of the bushes and quickly pounced on the situation. Journal and conference papers, newsletters, and the national media were swift to point to the shortcomings. For example, Forbes [29] asked, "What happened to those 'expert' systems that were supposed to transform the world of business forever"? Expert system designers began to realize that finding a place for the technology can be as tedious as matching the glass slipper to Cinderella's foot [2].

The turning point came in the mid-eighties when designers began to focus on very narrow, well-defined, and sometimes even mundane tasks. They also took the time to look at where the technology would be embedded.

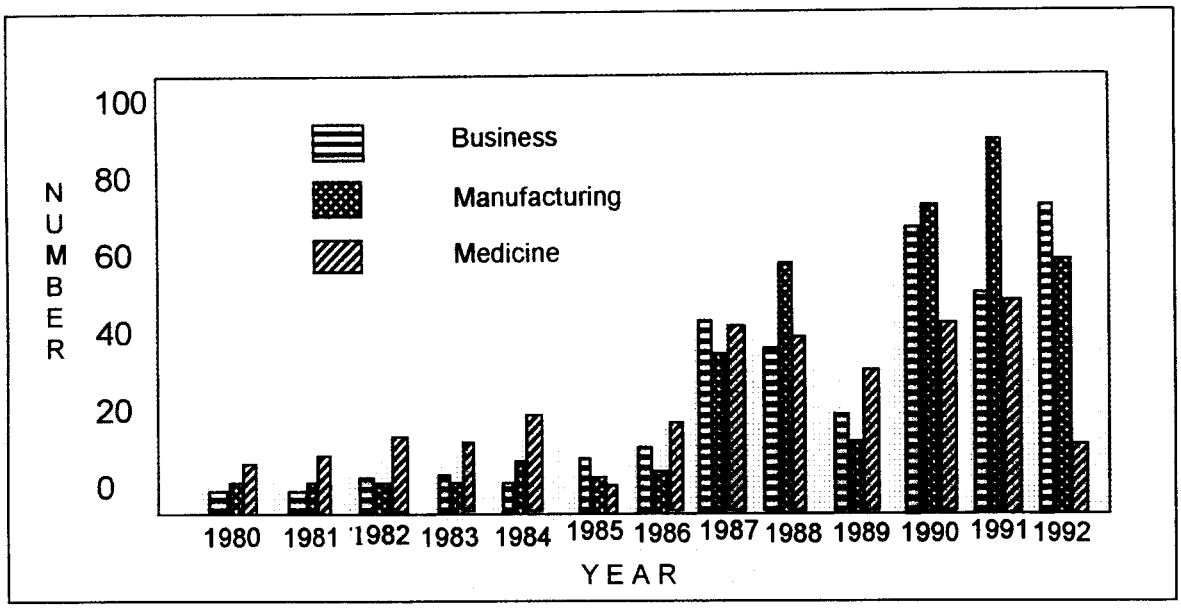


Fig. 3.4 The number of expert systems developed per year for business, manufacturing and medicine.

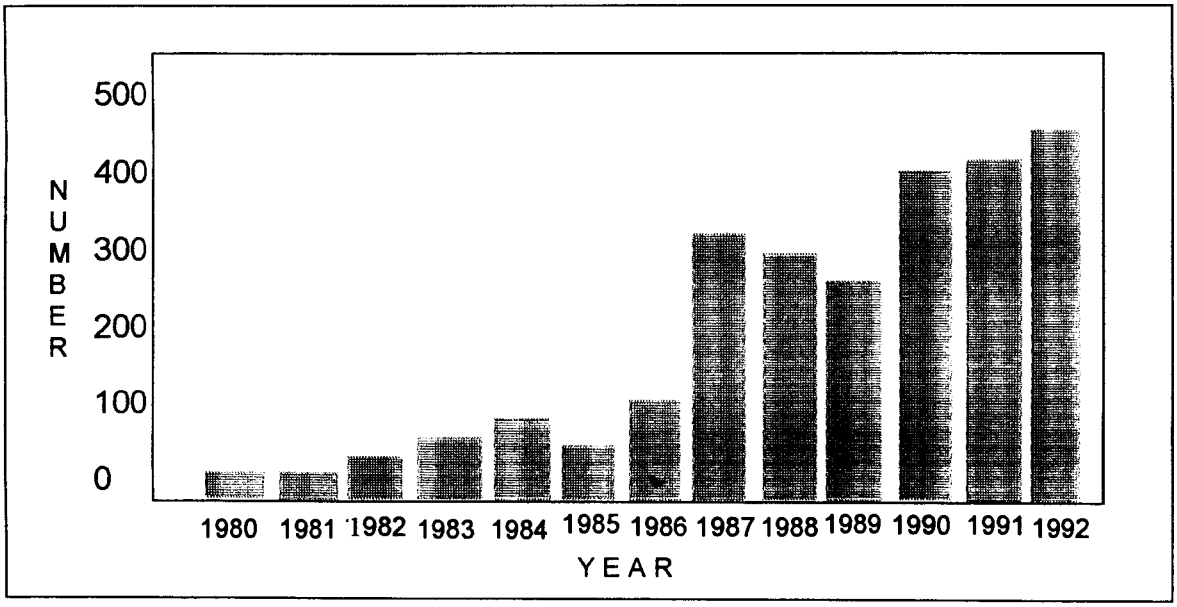


Fig. 3.5 The number of expert systems developed from 1980 - 1992.

One can observe a dramatic swing toward commercially viable systems, and applications for business and manufacturing began to pick up steam. Medical applications continued to grow. but not at a similar rate.

Durkin [2] divided the application areas into two categories : commercial and scientific. The first category includes systems that produce economically beneficial products for organizations in business, manufacturing, power systems, and transportation. The second category includes systems that produce primarily scientific results for chemistry, geology, image processing, and space technology. Fig.3.6 shows the ratio of systems developed in the commercial category to those developed in the scientific category per year.

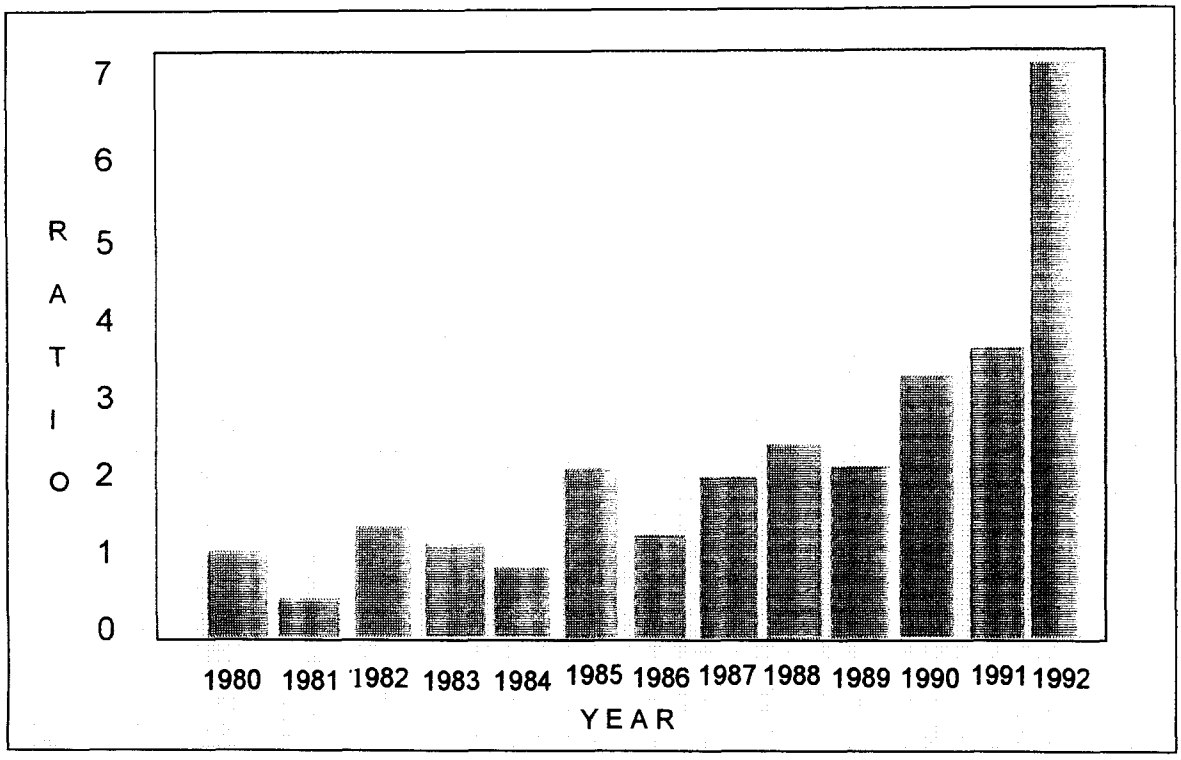


Fig. 3.6 The ratio per year of developed commercial expert systems to developed scientific ES.

In the early eighties this ratio remained around one-to-one; that is, as many commercial as scientific applications. The mid-to-later eighties showed a two-to-one ratio, while the early nineties showed a dramatic increase toward commercial applications.

During the seventies AI researchers were centered on producing intelligent general-purpose reasoning machines. The fascination of achieving this academic challenge drove their efforts. By the eighties, when the fuel for the advancement of the technology came from sectors that demanded a return on their investment, researchers began to realize that this is not a religious experience but an economic one. The trend in fig.3.6 - from laboratory to industrial applications - is one measure we can use to judge the technology's value. But, however, we feel that there is the need of exploring more domains suitable for expert systems applications on the one hand and then developing practical systems on the other.

3.2.3. **Components of an expert system**

Although at present there is no such thing as a standard expert system, but, however, most expert systems have a knowledge base and inference engine and a user interface. AI environments for expert system development are shown in block diagrams (fig.3.7 and fig.3.8) which are more or less self explanatory. The component of the expert system that contains collection of the domain knowledge for the system is called its knowledge base. This element of the system is so critical to the way most expert systems are constructed that they are also popularly known as knowledge based system. The **knowledge base** of an expert system contains both declarative (facts about objects, events and situations) and procedural (information about courses of action) knowledge depending on the form of knowledge representation chosen that two types of knowledge may be separate or integrated. There are several knowledge representation schemes such as Logic, Semantic Networks, Frames, Rules etc. which will be discussed in **chapter 4**.

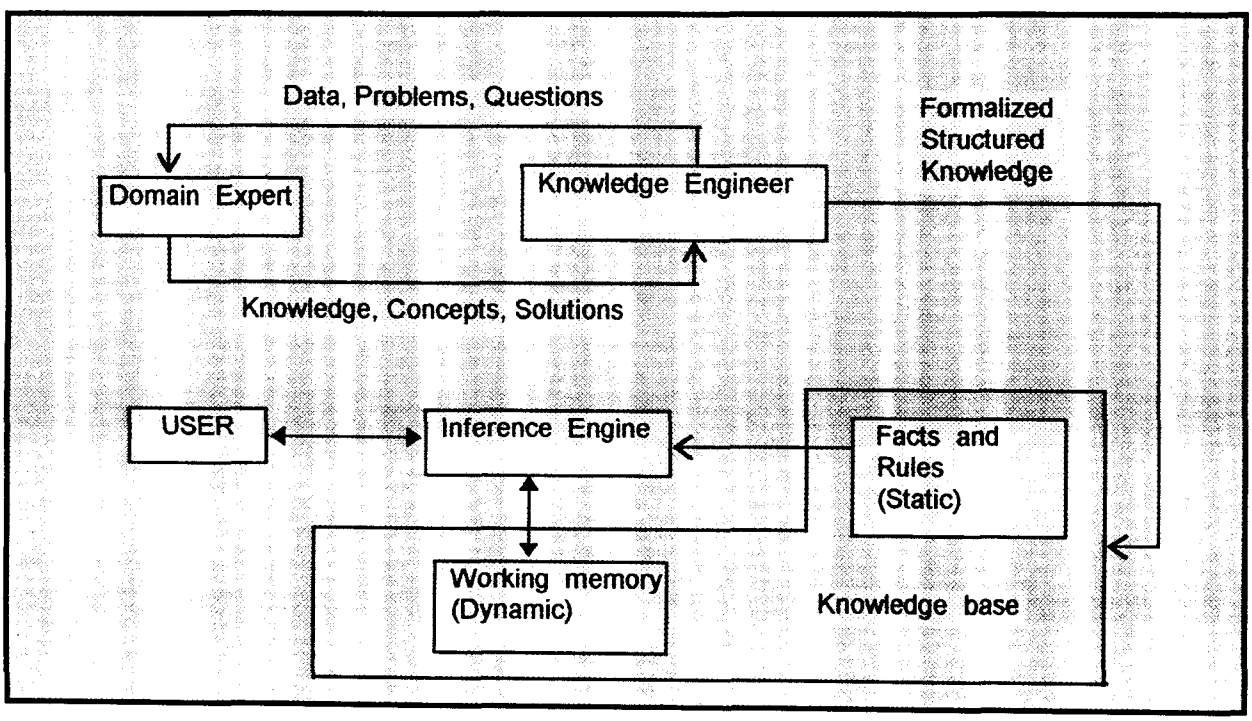


Fig. 3.7. An architecture of a typical expert system.

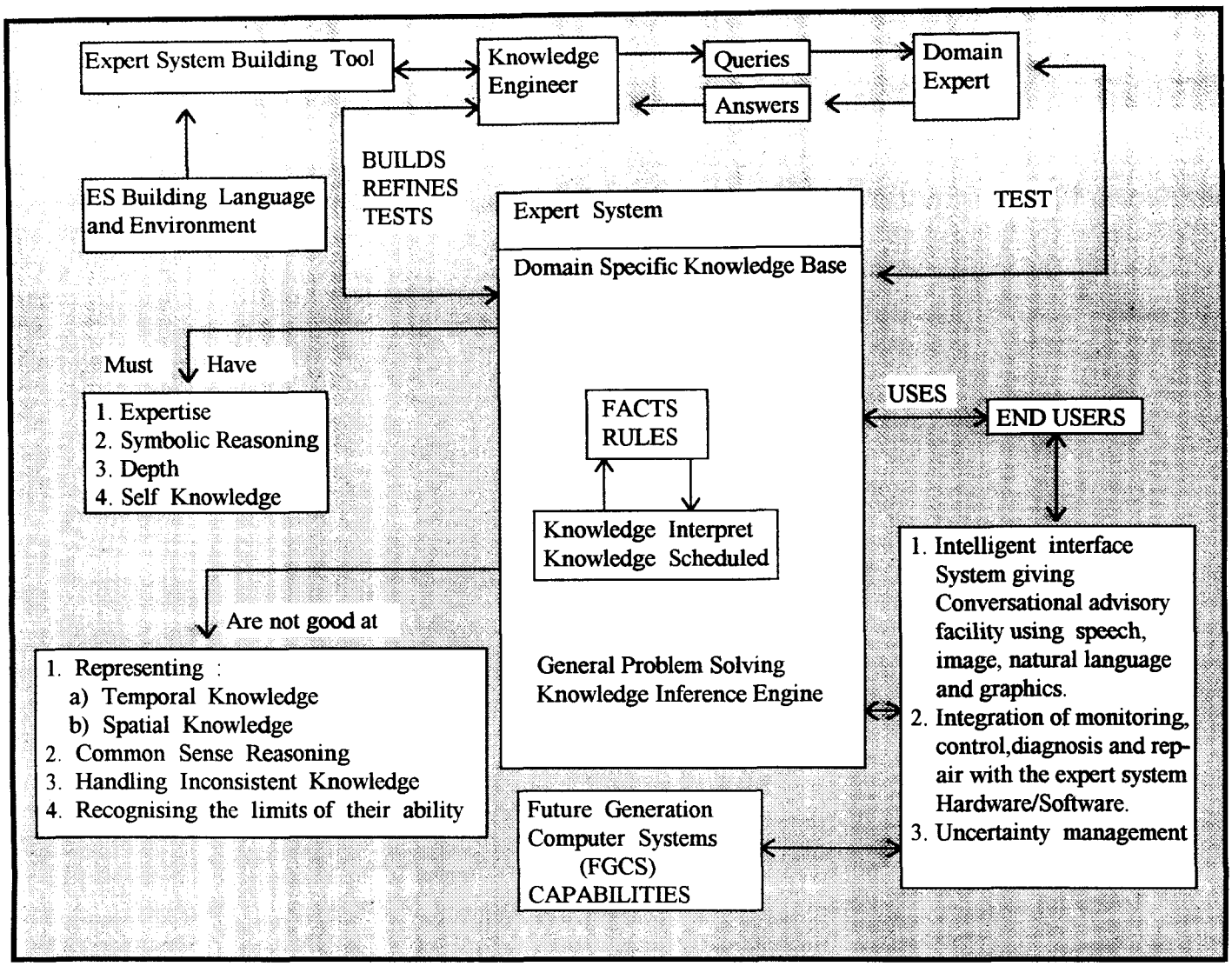


Fig. 3.8. A block diagram indicating ES-development environment [30].

Simply having lot of knowledge does not make one an expert. The system must know how and when to apply the appropriate knowledge. So having a knowledge base itself does not make an expert system intelligent. The component that is responsible to direct the implementation of the knowledge is known variously as the control structure, the rule interpreter, or the inference engine. The **inference engine** defines which heuristic search techniques are used to determine how the rules in the knowledge base are to be applied to the problem. As a matter of fact the inference engine runs an expert system determining which rules are to be invoked accessing the appropriate rules in the knowledge base executing the rules and determining when an acceptable solution has been found. The knowledge in an expert system is not interwined with the control structure. As a result of which an inference engine that works well in one expert system may work just as well with a different knowledge base. For example, the inference engine of one of the most famous medical expert system MYCIN is available separately

as EMYCIN (essential MYCIN). EMYCIN can be used with a different knowledge base to create a new knowledge system eleminating need to develop a new inference engine.

Next important component is the **user interface** that enables user to communicate with an expert system. The communication performed by a user interface is bi-directional. At the simplest level the user must be able to describe his problem to the expert system and the system must be able to respond with its recommendations. The user may also ask the system to explain its reasoning or the user may ask the system for additional information about the problem. The system may also ask the user for additional information about the problem. In fig. 3.8 different features that are desirable for the end-user interface is described. As a matter of fact the capabilities of using speech, natural language, pictures and graphics are the most important features of the fifth generation systems also.

3.2.4. **Typical features of an expert system**

Inspite of the fact that each expert system is unique in some sense, certain features are desirable for any expert system. Some authors suggested 6 criteria as pre-requisites for acceptance of an expert system by its intended users. These are :

1. The program should be useful which means the expert system should be developed to meet a specified need for which it is recognised that assistance is needed;

2. The programme should be usable which means it should be designed so that a less experienced computer user can use it;

3. The programme should be educational and appropriate which means a non-expert can use the system and increase his own expertise by using the system;

4. The programme should be able to explain its advice, which means the reasoning process of the system should be transparent so that the user is able to decide whether to accept the system's recommendations;

5. The programme should be able to learn new knowledge, which means the system should ask questions to the user to gain additional information and incorporate it to the system, if necessary;

6. The programme's knowledge should be easily modifiable so that the knowledge base of an expert system can be revised easily to correct errors or to add new information.

3.2.5. Major stages of an expert system development

Expert system development can be viewed as five highly interdependent and overlapping phases : identification, conceptualization, formalization, implementation and testing [3]. Fig. 3.9 illustrates the stages of an expert system development.

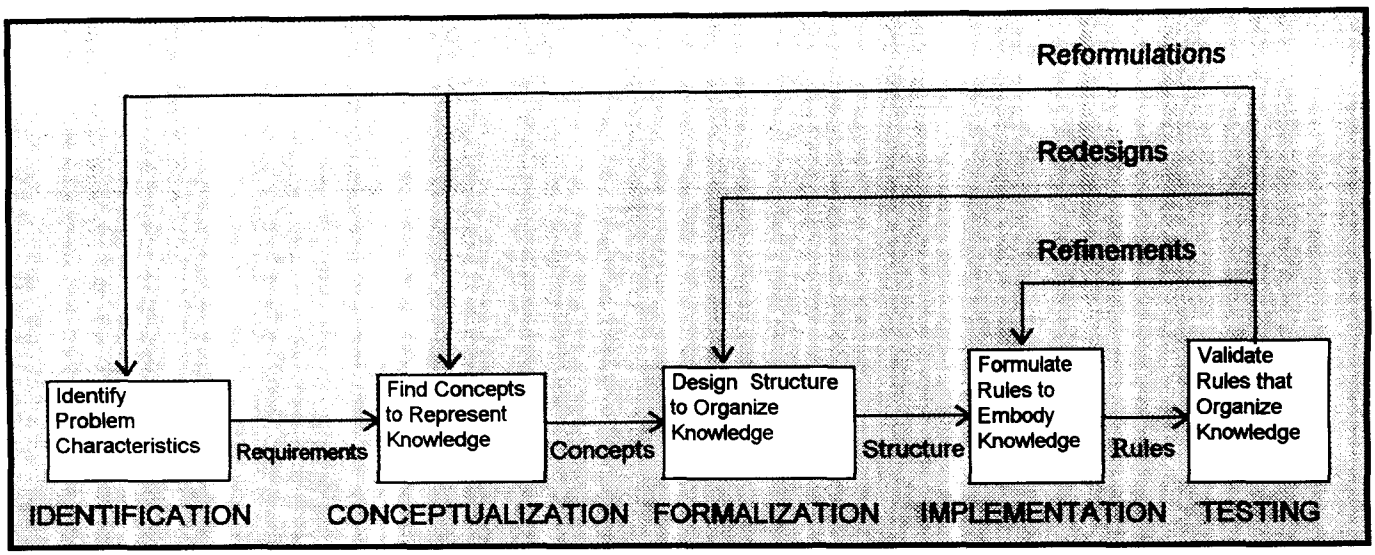


Fig. 3. 9 Stages of expert system development.

- **Identification**

The knowledge engineer and expert determine the important features of the problem. This includes identifying the problem itself (e.g. type and scope), the participants in the development process (e.g. additional experts), the required resources (e.g. time and computing facilities), and the goals or objectives of building the expert system (e.g. improve performance or distribute scarce expertise). Of these activities, identifying the problem and its scope gives developers the most trouble. Often the problem first considered is too large or complex and must be scaled down to a manageable size. The knowledge engineer may obtain a quick measure of this complexity by focusing on a small but interesting sub-problem and implementing routines to solve it.

- **Conceptualization**

The knowledge engineer and expert decide what concepts, relations and control mechanisms are needed to describe problem solving in the domain. Subtasks, strategies and constraints related to the problem-solving activity are also explored. At this time the issue of granularity is usually addressed. This just means considering at what level of detail the knowledge should be represented. The knowledge engineer will normally pick the most abstract level of detail (coarsest grain) that still provides adequate discrimination between key concepts. A word of warning - the developers must avoid

trying to produce a complete problem analysis before begining program implementation. They will learn much from the first implementation that will shape and direct the conceptualization process.

- **Formalization**

Formalization involves expressing the key concepts and relations in some formal way, usually within a framework suggested by an expert system building language. Thus the knowledge engineer should have some ideas about appropriate tools for the problem by the time formalization begins. For example, if the problem seems amenable to a rule based approach, the knowledge engineer might select ROSIE as the system building language and gather expertise in the form of IF-THEN rules. If a frame-based approach seems more appropriate, the knowledge engineer might instead select SRL and work with the expert to express domain knowledge as a large network.

- **Implementation**

The knowledge engineer turns the formalized knowledge into a working computer program. Constructing a program requires content, form and integration. The content comes from the domain knowledge made explicit during formalization, that is, the data structures, inference rules and control strategies necessary for problem solving. The form is specified by the language chosen for system development. Integration involves combining and reorganizing various pieces of knowledge to eliminate global mismatches between data structures and rule or control specifications. Implementation should proceed rapidly because one of the reasons for implementing the initial prototype is to check the effectiveness of the design decisions made during the earlier phases of development. This means that there is a high probability that the initial code will be revised or discarded during development.

- **Testing**

Testing involves evaluating the performance and utility of the prototype program and revising it as necessary. The domain expert typically evaluates the prototype and helps the knowledge engineer to revise it. As soon as the prototype runs on a few examples, it should be tested on many problems to evaluate its performance and utility. This evaluation may uncover problems with the representational scheme, such as missing concepts and relations, knowledge represented at the wrong level of detail, or unwieldy control mechanisms. Such problems may force the developers to recycle through the various development phases, reformulating the concepts, refining the inference rules and revising the control flow.

3.2.6. Classifications of expert systems

3.2.6.1. Based on reasoning

- **Rule-based reasoning**

A type of knowledge representation in which the knowledge about a domain is expressed in rules that define relationships between facts. Rules provide a formal way of representing recommendations, directives or strategies. They are often appropriate when the domain knowledge results from empirical associations developed through years of experience solving problems in an area.

- **Case-based reasoning**

A knowledge base for case-based reasoning [31] is a set of relevant examples rather than general rules. These cases are applied to new problems by an analogical reasoning process. This is another response to the complexities encountered in trying to handcraft a knowledge base of general rules that will cover all situations. Proponents of case-based reasoning argue that this is closer to human reasoning. Case-based approaches have played an important role in expert programs in law and medicine. A comparison of rule-based reasoning vs. case-based reasoning has been shown in table 3.4. In the recent years case-based approaches are gaining momentum in different domains.

Table 3.4. Comparison of Case-based and Rule-based reasoning [32]

Criterion	Rule-based reasoning	Case-based reasoning
Knowledge unit	Rule	Case
Granularity	Fine	Coarse
Knowledge acquisition units	Rules, hierarchies	Cases, hierarchies
Explanation mechanism	Backtrace of rule firings	Precedent cases
Characteristic output	Answer, plus confidence measure	Answer, plus precedent cases
Knowledge transfer across problems	High, if backtracking Low, if deterministic	Low
Speed as a function of knowledge base size	Exponential, if backtracking; Linear,if deterministic	Logarithmic, if index tree balanced
Domain requirements	Domain vocabulary Good set of inference rules Either few rules or Rules apply sequentially Domain mostly obeys rules	Domain vocabulary Database of example cases Stability - a modified good solution is probably still good Many exception to rules
Advantages	Flexible use of knowledge Potentially optimal answers	Rapid response Rapid knowledge acquisition Explanation by examples
Dis-advantages	Computationally expensive Long development time Black-box answers	Suboptimal solutions Redundant knowledge base

- **Frame-based reasoning**

Reasoning with frames is much more complicated than reasoning with rules. The slot provides a mechanism for a kind of reasoning called expectation-driven processing. Empty slots (i.e. uncofirmed expectations) can be filled, subject to certain conditioning, with data that confirm the expectations. Thus, frame-based reasoning looks for confirmation of expectations and often just involves filling in slot values.

Perhaps the simplest way to specify slot values is by default. The default value is attached loosely to the slot so as to be easily displayed by a value that meets the assignment condition. In the absence of information, however, the default value remains attached and expressed.

The reasoning process that takes place with frames is essentially the seeking of confirmation of various expectations. This amounts to filling in the slots and verifying

that they match the current situation. With frames, it is easy to make inferences about new objects, events, or situations because the frames provide a base of knowledge drawn from previous experience.

The reasoning in frames can be executed in different ways. Two most common ways are using rules and employing hierarchial reasoning.

In frame-based system, it is always easy to see the order and relationship of the elements. Frame-based systems assume that the hierarchical relationship of the objects is relatively static. If the order is dynamic, using a frame-based system becomes difficult.

Frame-based systems are most applicable to biological classification systems, and similar types of systems, in which a static hierarchical classification is a part of the knowledge.

- **Model-based reasoning**

Model-based reasoning is based on knowledge of the structure and behavior of the devices the system is designed to understand. Model-based systems are especially useful in diagnosing equipment problems. The systems include a model of the device to be diagnosed that is then used to identify the cause(s) of the equipment's failure. Because they draw conclusions directly from knowledge of a device's structure and behavior, model-based expert systems are said to reason from "first principles".

Unlike rule-based expert systems which are based on human expertise, the model-based ones are based on knowledge of the structure and behaviour of the devices they are designed to understand.

3.2.6.2. **Based on other technical issues**

Mentioning two major problems in building expert systems : (i) constructing and debugging knowledge base, and (ii) management of uncertainties might be relevent here. In the recent years, different ideas, concepts, methodologies have been introduced in circumventing the above and allied problems in building knowledge-based expert systems and / or in improving the performance in decision making systems. The resulting basic modules of various expert systems [33] are shown in fig. 3.10.

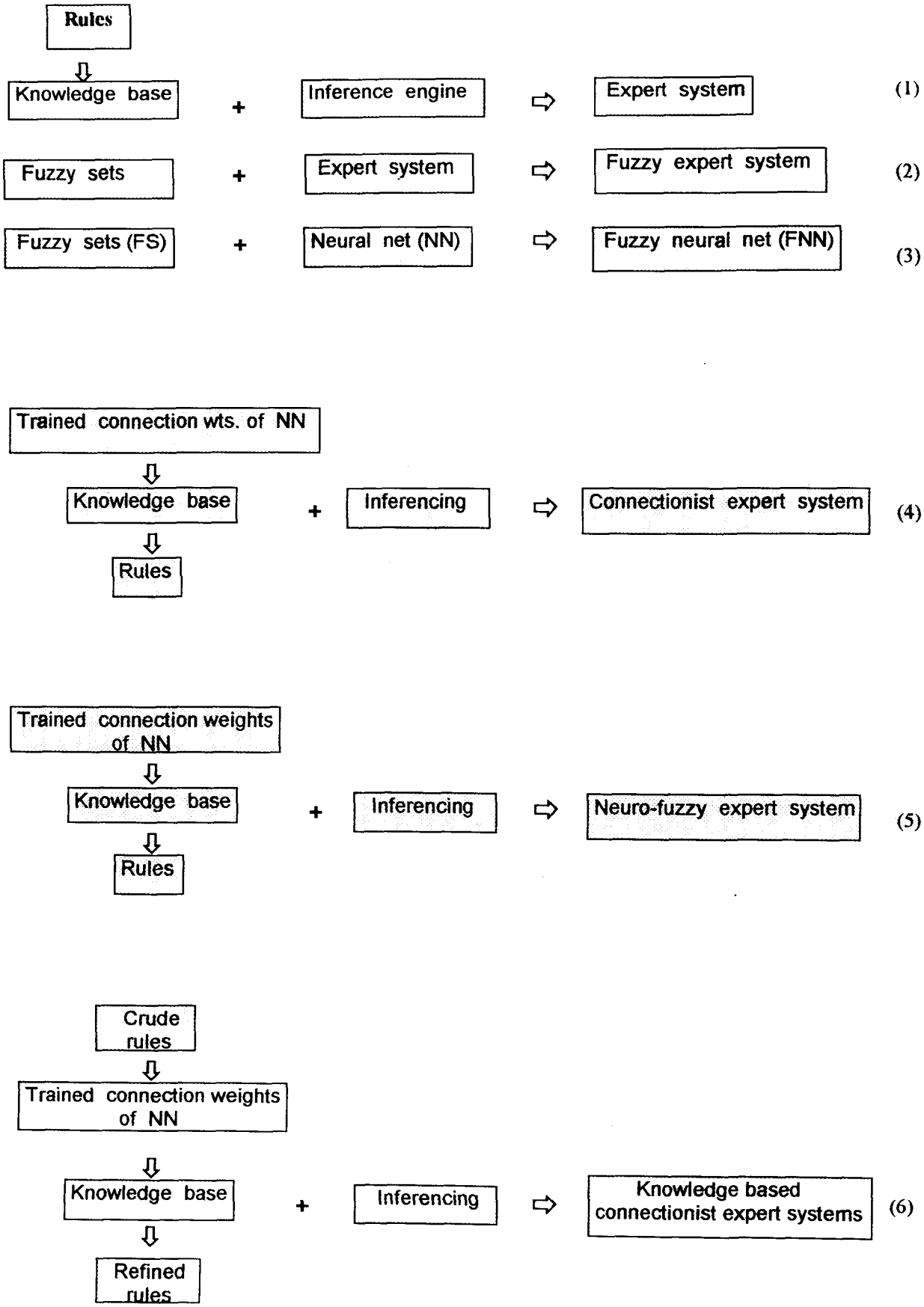


Fig. 3.10. Block diagram of the basic modules of various expert systems.

Artificial neural networks [34-37] can be formally defined as massively parallel interconnections of processing elements that interact with objects of the real world in a manner similar to biological systems. All information is stored distributed among the various connection weights. The networks can be trained by examples and sometimes they generalize well for unknown test cases.

Fuzzy logic is based on the theory of fuzzy sets and, unlike classical logic, it aims at modeling the imprecise (or inexact) modes of reasoning and thought processes (with lingistic variables) that play an essential role in the remarkable human ability to make rational decisions in an environment of uncertainty and imprecision. This ability depends, in turn, on our ability to infer an approximate answers to a question based on a store of knowledge that is inexact, incomplete, or not totally reliable.

We see that fuzzy set theoretic models [38,39] try to mimic human reasoning and the capability of handling uncertainty, whereas the neural network models attempt to emulate the architecture and information representation schemes of the human brain. Integration of the merits of fuzzy set theory and neural network theory therefore promises to provide more intelligent systems to handle real life recognition / decision making problems. For the last five to seven years, there have been several attempts [40-43] by researchers over the world in making a fusion of the merits of these theories under the heading 'neuro-fuzzy computing' for improving the performance in decision making systems.

As the knowledge base of an expert system is a repository of human knowledge and since some of these may be imprecise in nature, often, this may result in a collection of rules and facts which for the most part are neither totally certain nor totally consistent. The expert system is also likely to be required to infer from premises that are imprecise, incomplete or not totally reliable. The uncertainty of information in the knowledge base of the question-answering system thus induces some uncertainty in the validity of its conclusions [44]. Hence a basic problem in the design of expert systems is the analysis of the transmitted uncertainty from the premises to the conclusion and the association of a certainty factor [45]. Fuzzy expert systems [45,46], incorporating the concept of fuzzy sets at various stages, help to a reasonable extent in the management of uncertainty in such situations.

Neural networks are also used in designing expert systems. Such models are called connectionist expert systems [47], and they use the set of connection weights of a trained neural net for encoding the knowledge base for the problem under consideration.

The block diagram of the basic modules of an expert system, fuzzy expert system, fuzzy neural net, connectionist expert system, neuro-fuzzy expert system and knowledge-based connectionist expert system have been provided in fig. 3.10. As stated above, a

fuzzy neural net constitutes the knowledge base of a neuro-fuzzy expert system. While the rules are collected by knowledge engineers for designing the knowledge base of a traditional expert system or fuzzy expert system, the connectionist models use the trained link weights of the neural net / fuzzy neural net to automatically generate the rules, either for later use in a traditional version or for providing justfication in the case of an inferred decision. This automates and also speeds up the knowledge acquisition process. The use of fuzzy neural nets helps in the handling of uncertainty at various levels (e.g. input, output, learning and neuronal) and generates fuzzy rules capable of more realistically representing real-life situations. The knowledge-based connectionist expert systems, on the other hand, initially encode crude domain knowledge among the connection weights of the neural net, thereby speeding up the training phase and generating better performance. Refined rules are later extracted from the less redundant trained network.

A comparative analysis of the basic features of these models with those of the traditional and connectionist (non-fuzzy) versions is provided in table 3.5.

Table 3.5. Comparative study of various expert systems.

	Expert system	Connectionist expert system	Neuro-fuzzy expert system	Knowledge-based connectionist/ Neuro-Fuzzy expert system
Knowledge base	Knowledge acquisition and representation in the form of rules, frames, semantic nets or belief networks	Connection weights of trained neural net that were initialised with small random values	Connection weights of trained fuzzy neural net that were initialised with small random values	Connection weight of trained nonfuzzy/ fuzzy neural net that were initialised with crude domain knowledge in rule form with binary link weights apriori class information and distribution of pattern points
Knowledge refinement	Addition of new knowledge (say, as new rules)	Empirical addition of hidden nodes/links	Empirical addition of hidden nodes/links	Network optimization using growing and pruning of nodes /links, based on training data and additional knowledge
Iferencing	Matching facts with the existing knowledge base	Presentation of crisp input, forward pass and generation of crisp output	Presentation of fuzzy input, forward pass and generation of fuzzy output	Presentation of input , forward pass and generation of output
Rule generation		Crip rules obtained during backward pass using changes in levels of input and output units, magnitude of connection weights	Fuzzy rules obtained during backward pass using node activations and link weights	Rules obtained during backward pass; negative rules also possible

3.3. Why it is an expert system domain

There are two distinct parts under this aspect : (i) why does the domain demand ES-technology ? and (ii) why does the ES-technology suit the domain ? Let us consider the first issue, the foundation of which may be traced in Chapter 1 and chapter 2. Constant monitoring of growth and development performance of children is highly required. As an ideal case, atleast one paediatrician should be placed at each rural health centre. But, for a developing country like India, paediatrician is really a scarce commodity. To mitigate such needed expertise, an automated knowledge-based consultation system may be helpful. Let us now examine how the ES-technology suits the problem domain. The key attributes of a domain, to be a good candidate for expert system domain, are neither all absolute nor limited to the following [48-50] :

- The Knowledge associated with the domain must be bounded;
- Non-algorithmic approach is more useful than conventional algorithmic approach;
- Human experts or literature should be available or some prior case studies should be conducted to gather knowledge where human experts knowledge is neither adequate nor any literature is available;
- There should be some advantage to using computers with a significant payoff;
- The complete logic is not known in advance;
- Primarily it requires symbolic reasoning;
- There may be use of heuristics by the expert(s). Problems require multicriteria decision making (MCDM) [51] or use of incomplete or uncertain information;
- The domain is fairly stable or at least slowly changing;
- No alternative solution to the problem is being pursued or is expected to be pursued. The present solution under investigation for the domain problem will be used for quite sometime.

The knowledge associated with the domain is large and varied. The boundaries of a geographical region are somewhat defined but defining such boundaries of knowledge in paediatrics may not readily be possible. Reaching a conclusion with unbounded knowledge may not be possible. Therefore, experts somehow confine their knowledge while handling any problem. We confine ourselves within such expertise knowledge during the knowledge acquisition process. It may be stated as a fact that the stimulation reactions or disease patterns on human body do not obey any particular algorithm or a particular set of algorithms. So, it is better to use here a non-algorithmic approach. Multiple experts of the domain are available here. We have no doubt to state that a significant pay off from the completed system will be achieved. This pay off may be in terms of tangible benefits or may be in terms of social values. Knowledge of this problem domain is vast and varied and hence complete and sound knowledge may not be available in advance. This requires the use of expert systems technology where one may expect the ease of updating and maintainability of

knowledge base. Primarily, the domain requires to deal with some symptoms (some symbols) like "axial muscle tone" or "spontaneous gestures" or "rhythms" (**Appendix A**). So, symbolic reasoning is the primary component here. It is such a domain where the use of heuristics by the experts gained in a number of years of practice will be useful. Multiple criteria decision making and incomplete or uncertain information processing are also the characteristics of the domain. The characteristics of the domain under consideration are of fairly stable nature. It is unlikely that the characteristics of this region will change abruptly in near future. It is also unlikely that the characteristic diagnostic parameters of child as well as the expected values will change drastically. From the socio-economic point of view, no better solution, seems feasible. It may not be possible to appoint even one human expert on child care per health centre of a developing country. The needs of the domain may be fulfilled, at least partially, by the present system under investigation. It is expected that the full system once achieved will partially; be used as long-term basis, an important pay off.

3.4. What requirements the domain lays on an expert system

With the above justifications of using expert systems technology for the domain, let us now investigate the requirements the domain lays on an expert system.

- **Portability**

To have its increased usage an expert system is expected to be portable. This essentially means that the system can be run on different types of target machines which can be procured at low cost and can be transported easily to different remote health centres. Moreover, the recurring expenditure should be low in terms of power consumption, maintenance etc. During the system development, one has to select a software development tool to satisfy the said purpose. For example, one may suitably select PROLOG / LISP or an ES-shell or a tool-kit based on PC running under MS-DOS/Windows. Summarily, a low cost and easily manageable by the end users PC-based system is being proposed here. This portability feature should certainly encourage the usage issue discussed in **chapter 1**. It should be easier then for doctors who already have hardwares with them to procure this system. This may require a small upgradation rather than procuring specialised LISP-based machines or AI workstations.

- **Modifiability**

The domain knowledge in knowledge base may have to be enhanced owing to different reasons. Three specific reasons may here be noted. First, when complete and sound knowledge may not be available in advance, a fact for the present domain, existing system should easily and quickly incorporate the required changes, specially

bearing in mind the state-of-the-art knowledge of the domain. Secondly, the complete and sound knowledge may not be possible to acquire in the initial stage of the knowledge acquisition process. At the later stage of the development, further enhancement would be required. Third, for its survival, a system should be of open type. This essentially means that the system should cope with the changing environment, obviously small, suggesting the modifiability feature to incorporate in the system. A closed system should eventually die. In a system, the modifiability has to be taken care of at two levels : i) at the design level, and ii) at the implementation level.

- **Dealing with inexact information**

In real world, we have the experience that sometimes either we have no knowledge about an object or we have some incomplete, fuzzy or uncertain knowledge about the object. But, one has to reason in this situation and has to reach a decision. For a paediatric domain this is more critical. An expert system should be capable of handling these inexact situations. The importance of the topic demands an elaborate discussion which has been provided in a separate chapter (Chapter 7).

- **Non-monotonic reasoning**

The information supplied by the parents / guardians of an investigative child is subjective sometimes. To deal with this subjective reply nonmonotonic reasoning (NMR) will be useful. NMR proceeds with its reasoning as if the assumptions are true with their definite certainty values. With its reasoning it reaches a conclusion. If one finds the conclusion to be absurd, it is demanding to change an assumption and / or to change the uncertainty values.

Question : Does your baby take 4-5 meals / day?
Answer : Doctor. She doesn't want to take meals.

With this reply (CF = 0.9) the doctor proceeds to other examinations. After overall examinations, the doctor finds that the baby is disease-free and the growth of the baby is normal. This finding, obviously, contradicts the reply from the mother whose CF may be 0.3 or 0.2. The associated facts and rules may have to be changed with this changing CF. This is how the nonmonotonic reasoning works to offer a safeguard to the subjective reply during a consultation session. NMR is also relevant in connection with modifiability. One can also have the view that uncertain reasoning itself has a nonmonotonic aspect.

- **Transparency**

For a firm diagnosis as well as for further course of action(s), a doctor may not be satisfied with the decision only offered by an expert system. He / she may demand the total reasoning path traversed by the system. It may, sometimes, also be demanded by the parents / guardians of the child under investigation, may be, for their mental satisfaction. Generally, 'HOW' and 'WHAT IF' types of transparency are expected. So, an explanation tracing procedure should be there, as a module, with the system. The most of the users of the system would be the general medical professionals and medical students who are not experts in paediatrics. This particular feature should assist them to view the chain of reasoning leading to a conclusion. This chain of reasoning should certainly assist a doctor for further analysis and treatment planning. This chain of reasoning should also be useful to nonmonotonic reasoning issue.

- **Learning facility with a dynamic knowledge base**

It may be useful to remember the results or facts of at least one previous consultation session for better comparison, especially for the paediatric field. It is true that the deficiency in growth and development should be estimated in comparison with a set standard i.e. milestones what are stored in static part of knowledge base. But, however, it should also be useful to estimate the increment / decrement of growth parameters in comparison with the previous consultation session. This should give us an idea about the parameters which need more attention. It may also be useful to avoid any repetitive questionnaire during interrogation with the child and / or with the parents / guardians. This is essentially a learning facility with the system. This facility should be useful with this type of application on the argument that there are a number of cases where paediatric patients are resistant to such interrogation. They may become hostile with any repetitive process. It is necessary to complete the process as quickly as possible. This facility may be achieved with a dynamic knowledge base. We call this dynamic portion of knowledge base as short-term knowledge base (STKB). This STKB may also help to achieve 'improved backtracking' compared to 'blind or chronological backtracking'. This STKB, we observe, may also play an active role on nonmonotonic reasoning.

- **Structured and modular data structure**

Let us now identify some key requirements of the domain in connection with its knowledge representation where structuredness and modularity are demanded for :

Managing a large and varied knowledge base

The domain knowledge size is significantly large and varied. In this situation, the knowledge can become unmanageable. To make it manageable, it will be worthwhile to use structured and modular data structure for knowledge representation.

Avoiding redundancy and thereby removing inconsistency

Any component of knowledge is expected not to be duplicated in a knowledge base either in the design phase or in the implementation phase. This redundant information requires more space and also leads to inconsistency problem during upgradation of knowledge. Using a structured and modular data structure one can avoid this redundancy problem.

High level of abstraction

An abstraction is a way of representing a group of related things by a single thing which expresses their similarities and suppresses their differences. For the present domain, the level of abstraction is expected to be high for the ease of proper diagnosis from a large and varied knowledge base. A high level of abstraction may be achieved using an equally highly structured and modular data structure for knowledge representation.

In **chapter 4**, a detail discussion has been provided on the knowledge representations schemes along with their relative merits and demerits.

3.5. **Discussions**

After a brief introduction to AI and expert systems technology, categories and application areas of expert systems with some examples have been provided. Some pages have been devoted for the discussion on the trends of applications of expert systems. Components of a typical expert system, typical features of an expert system, major stages of expert system development have been provided in brief. Then we have devoted some pages on the discussion on the types of expert systems with a note on the recent trends of the technology.

One can observe that recently case-based and / or model-based reasoning are preferred by some researchers in some domains (e.g., medical domain). For the generation of more intelligent decision making systems some researchers propose fuzzy systems, some propose neuro-fuzzy models, some propose knowledge-base networks model and some propose connectionist model. They have their relative merits and demerits. A comparative study of the various methodologies has been provided in tabular form.

In our present study we have explored the development of a rule based fuzzy object-oriented knowledge based system for the domain as well as applying case-based reasoning to the domain.

References

1. Frederick Hayes-Roth, Donald A. Waterman and Douglas B. Lenat eds. Building Expert Systems, Reading, MA., Addison-Wesley; 1983, 13-16.

2. John Durkin. Expert Systems : A view of the field. IEEE Expert; April 1996, 56-64.

3. Donald A. Waterman. A Guide to Expert Systems. Addison-Wesley Publishing Company; 1985.

4. B. E. Prasad, T. S. Perraju, G. Uma and P. Umarani. An expert system shell for Aerospace Applications. IEEE Expert; August 1994, 56-64.

5. A. G. Boulanger. The expert system Plant/cd : A case study in applying the general purpose inference system ADVISE to predicting black cutworm damage in corn. M S. thesis; Computer Science Deptt., University of Illinois at Champaign-Urbana, July 1983.

6. E. Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; Int. ed.,1992, 725-727.

7. B. G. Buchanan and E. A. Feigenbaum. Dendral and Meta-dendral : their applications dimension. Artificial Intelligence; vol.11,1978.

8. S. I. Goyal et al. COMPASS : An Expert System for Telephone switch maintenance. Expert Systems; July, 1985.

9. J. S. Bennett and C. R. Hollander. DART : An expert system for computer fault diagnosis. Proceedings IJCAI-81; 1981, 843-845.

10. E. Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; Int. ed.,1992, 732.

11. J. Dekleer and G. J. Sussman. Propagation of constraints applied to circuit synthesis. Circuit Theory and Applications; vol.8, 1980, 127-144.

12. J. Bennett, L. Creary, R. Engelmore and R. Melosh. A Knowledge-based consultant for structural analysis. Computer Science Dept., Stanford University, Stanford, California, September, 1978.

13. J. Gaschnig. Prospector : an expert system for mineral exploration. In Machine Intelligence. Infotech State of the Art Report 9; no.3,1981.

14. P. Suetens and A. Oosterlinck. Using expert systems for image understanding. Associated with the National Fund for Scientific Research (NFWO); Belgium, 1987, 61-73.

15. M. G. Valtorta, B. T.Smith and D. W. Loveland. The graduate course advisor : A multi-phase rule-based expert system. Proc. of the IEEE workshop on principles of knowledge-based systems, IEEE Computer Society, IEEE Computer Society Press; 1109 Spring Street, Silver Spring, Md., 1984.

16. Carole D. Hafner. Representation of knowledge in a legal information retrieval system. In R. Oddy, S. Robertson, C. Van Rijsbergen, and P. Williams (eds.) Information Retrieval Research, London : Butterworths and Co.; 1981.

17. M. S. Fox, B. Allen and G. Strohm. Job-shop scheduling : an investigation in constraint-directed reasoning. Proceedings AAAI-82; 1982, 155-158.

18. W. A. Martin and R. J. Fateman. The MACSYMA system. Proc. of the Second Symposium on symbolic and algebraic manipulation; March 1971, 59-75.

19. L. E. Rodewald. Baby : an expert system for patient monitoring in a newborn intensive care unit. M. S. thesis; Computer Science Dept., University of Illinois, Champaign-Urbana, 1984.

20. E. Shortliffe, S. G. Axline, B. G. Buchanan, T. C. Merigan and S. N. Cohen. An artificial intelligence program to advise physicians regarding antimicrobial therapy. Computers and Bio-medical Research; vol.6,1973, 544-560.

21. D. Michie, S.Muggleton,C. Riese and S. Zubrick. Rulemaster : a second-generation knowledge-engineering facility. Proc. of the first Conf. on artificial intelligence applications, IEEE Computer Society; December 1984.

22. G. R. Ferguson. Aircraft maintenance expert systems. Master's thesis: Air Force Institute of Technology, Wright-Patterson AFB, Ohio, November 1983.

23. E. Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; Int. ed.,1992, 720-723.

24. D. R. Barstow. Knowledge engineering in Nuclear Physics. Proc. IJCAI-79; vol.1, 1979, 34-36.

25. D. Chester, D. Lamb and P. Dhurjati. Rule-based computer alarm analysis in chemical process plants. Proc. of the Seventh Annual Conf. on Computer Technology, MICRO-DELCON 84, IEEE; March 1984, 22-29.

26. Artificial Intelligence Letter V, no. 10, October 1989, Texas Instruments, Data systems group, Austin, Texas.

27. F. J. Dickey and A. L. Toussaint. ECESIS : an application of expert systems to manned space stations. Proc. of the first Conf. on Artificial Intelligence applications, IEEE Computer Society; December 1984.

28. E. Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; Int. ed.,1992, 703-704.

29. R. Simon. The Morning after. Forbes; vol.140, no.8, October 19, 1987,164-168.

30. D. Dutta Majumder. Artificial intelligence and expert systems : Its role in Industrial System Development. Nat. Symp. on Elec. and Appl.; Sept.,1989.

31. J. L. Kolodnor. Extending problem solver capabilities through case-based inference. Proc. of the fourth Int. workshop on machine learning; Los Altos, CA, Morgan Kaufmann, 1987.

32. E. Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; Int. ed.,1992, 225-226.

33. S. Mitra and S. K. Pal. Neuro-fuzzy Expert Systems : relavance, features and methodologies. JIETE; vol.42, nos.4 and 5, July - Oct 1996, 335-347.

34. D. E. Rumelhart and J. L. McClelland. Parallel distributed processing : Explorations in the microstructures of Cognition. eds., vol.1, Cambridge, MA : MIT Press; 1986.

35. R. P. Lippmann. An introduction to computing with neural nets. IEEE Acoustics, Speech and Signal Processing Magazine; vol.4,1987, 4-22.

36. T. Kohonen. Self-organization and associative memory. Berlin, Springer-Verlag; 1989.

37. J. Hertz, A. Krogh and R. G.Palmer. Introduction to the theory of Neural computation. Reading, MA , Addison-Wesley; 1994.

38. L. A. Zadeh. Fuzzy Sets. Information and Control; vol.8, 1965, 338-353.

39. G. J. Klir and T. Folger. Fuzzy Sets, Uncertainty and Information. Reading, MA, Addison-Wesley; 1989.

40. J. C. Bezdek and S. K. Pal. Fuzzy models for pattern recognition : methods that search for structures in data, (eds.), New York, IEEE Press; 1992.

41. Y. H. Pao. Adaptive pattern recognition and neural networks. Reading, MA, Addison - Wesley; 1989.

42. Proceedings of third International Conference on fuzzy logic and neural networks (IIZUKA 94); Iizuka, Fukuoka, Japan, July 1994.

43. Proceedings of Third International Conference on Fuzzy Systems (FUZZ-IEEE); Florida, USA, June 1994.

44. H. J. Zimmermann. Fuzzy sets, decision making and expert systems. Boston, MA, Kluwer, Academic Publishers; 1987.

45. L. A. Zadeh. The role of fuzzy logic in the management of uncertainty in expert systems. Fuzzy Sets and Systems; vol.11, 1983, 199-227.

46. A. Kandel. Fuzzy Expert Systems. (ed.), Boca Raton, CRC Press; Inc.,1991.

47. S. I. Gallant. Connectionist Expert Systems. Communications of the Association for Computing Machinery; vol.31, 1988, 152-169.

48. B. G. Buchanan, D. Barstow, R. Bechtal, J. Bennett, W. Clancey, C. Kuiilowski, T. Mitchell and D. A. Waterman. Constructing an expert system in F. Hayes-Roth, D. A. Waterman, D. B. Lenat, (eds.), 1983, Building Expert Systems. Addison-Wesley, Reading, Mass.; 1983, 127-167.

49. R. H. Anderson. Four criteria for choosing expert system applications. Asian Computer Monthly; April, 1985, 21-22.

50. D. S. Prerau. Choosing an expert system domain, in G. Guida and C. Tasso, eds., 1989. Topics in expert system design; North-Holland, Amsterdam, 27-43.

51. D. Dutta Majumder. Fuzzy mathematics and uncertainty management for decision making in science and society. Computer Science and Informatics; vol.23, no.3, 1993,1-31.

KNOWLEDGE ACQUISITION AND REPRESENTATION†

4.1. Introduction

Knowledge is as much an essential ingredient to the artificial intelligence of a computer as it is also to the natural intelligence of a person. Knowledge is the essential part of an expert system; it is what distinguishes an expert system from a conventional program. Knowledge acquisition is of critical importance to the ultimate success of the expert system development. Knowledge is a collection of specialized facts, procedures, and judgement rules. Knowledge may be collected from many sources. A representative list of sources includes domain experts, books, computer databases, maps, flow diagrams, pictures, web-sites etc. These sources can be categorised into two types : documented and undocumented sources.

Domain experts are generally considered as the primary source of knowledge for an expert system development. Experts should have developed domain expertise by task performance over a long period of time. One of the objectives of the knowledge acquisition is to find the experts' heuristics related to the task. Project experts should have enough experience to have been able to develop the domain insights that result in these heuristics.

Experts should be capable of communicating their knowledge, judgement, and experience and the methods they use to apply these to the particular task. Experts' temperament, cooperativeness, and working relation with the project team can have a major impact on the success and the speed of the knowledge acquisition.

After the knowledge acquisition, this knowledge has to be put into an objective form for the knowledge base. The proper selection and design of a suitable knowledge representation scheme should be in tune with the requirements of the application domain. In addition, the proper selection should also depend on certain important properties of a scheme like expressive power and adequacy in context to the application domain. In this chapter, we have tried to analyse some of these issues from the viewpoint of an expert system designer.

---

† This is based on the publications [Modelling, Measurement and Control, D, AMSE Press, France, vol. 17, no. 1,2, 1998, 1-19; Converse : A Themis Pharmaceuticals Publication, India, vol. 1, no. 2,1996, 13-22] of the author.

Section 4.2 will be devoted to describe levels of knowledge. Knowledge categories are presented in section 4.3. In section 4.4, we describe different sources of knowledge. Methods of knowledge acquisition are presented in section 4.5. Knowledge acquisition problems and possible ways of overcoming them are discussed in section 4.6. Section 4.7 contains the prime sources used in the present research. Section 4.8 will be devoted to describe some knowledge representation schemes from the literature. In section 4.9, we shall analyse the relative suitability of such schemes as described in section 4.8. In section 4.10, we have presented some representative expert systems and ES-developmental tools along with the KR-schemes and control mechanism, they use. In section 4.11, the knowledge of the present problem domain has been represented in different schemes as discussed in section 4.8. Finally, we end up with some discussions.

4.2. Levels of knowledge

Knowledge can be represented at different levels, of which two extremes are - shallow knowledge and deep knowledge. Shallow knowledges are the surface level informations, that can be used to deal with very specific situations. Deep knowledge refers to the internal and casual structure of a system and considers the interactions among the systems component. Deep knowledge can be applied to different task and different situations. It is based on a completely integrated, cohesive body of human consciousness that includes emotions, common sense, intution etc.

4.3. Knowledge categories

Knowledge can be differentiated into various categories - such as declarative knowledge, procedural knowledge, semantic knowledge, episodic knowledge and metaknowledge.

4.3.1. Declarative knowledge

Descriptive representation of knowledge is a declarative knowledge. It is expressed in a factual statement. Declarative knowledge is especially important in the initial stage of knowledge acquisition.

4.3.2. Procedural knowledge

It includes step-by-step sequences and how-to types of instructions, it may also include explanations.

4.3.3. Semantic knowledge

Sementic knowledge reflects cognitive structure that involves the use of the long term memory.

4.3.4. **Episodic knowledge**

Episodic knowledge is autobiographical, experimental informations organized as a case or an episode.

4.3.5. **Meta-knowledge**

Metaknowledge means knowledge about knowledge. In AI, metaknowledge refers to the knowledge about the operation of knowledge based systems i. e., about its reasoning capabilities.

4.4. **Sources of knowledge**

From many sources knowledge can be collected. We may classify them into two broad categories : (i) classical sources, and (ii) more recently available web-based sources.

4.4.1. **Classical sources**

A representative list of classical sources includes domain experts, books and literature, films, computer databases, pictures, maps, flow diagram, stories, songs, investigating tools such as ECG, ultrasound scan etc. Furthermore, these sources can be divided into two types : documented and undocumented knowledge. Undocumented knowledge resides in people's mind. Worthwhile to mention that in medical domain there are scopes of accumulating undocumented knowledge as gathered by medical practitioners during the examinations of the patients. In this respect, domain medical experts might be considered as a good source of knowledge. Although there is a need for better methods of knowledge acquisition, including techniques to automate the process, but for the forseeable future, most of the knowledge for any practical expert system in a complex domain will be obtained through the interaction of knowledge engineers and domain experts [1]. To mitigate the lack of domain experts, recently web-based knowledge acquisition can be used; the details of which is presented in the following sub- section.

4.4.2. **Web-based knowledge acquisition**

Since the introduction of expert systems in medical domain nearly 30 years back, as a matter of fact, most expert systems have not been found their places in routine clinical use. At per the 1994 reports [2] only 25 systems were used by tapping into the global collecected wisdom of the Artificial Intelligence in Medicine special interest group on the Internet. At per 14th June, 1999, this number increases to 39 as shown in Table 3.3. (**Chapter 3**). The situation was very different a decade ago when most of the systems were in experimental stages. Many peoples were in doubt whether AI technology should find its place in actual medical floor. The current picture is quite different. What is apparent from Table 3.3 is that AI Systems are actively working in many different roles.

But this number is not large enough as expected. Why ? Various investigators have proposed different explanations in this context as follows :

1. Knowledge acquisition and representation have been considered as the prime factor in the development process of a medical consultation system.
2. Some authors point to the inadequay of a formal theory for knowledge engineering demanding a more principled methodologies that can be used by the system developers.
3. Some point to the inadequacy of formal models of knowledge representation.
4. Some authors stress more on the lack of time and poor availability of medical experts for establishing high-quality knowledge-bases.

4.4.2.1. **Internet and WWW**

With the introduction of Internet and world-wide-web(WWW) knowledge acquisition has got a new dimension. WWW has not only revolutionized the dissimination process of information but also it has created novel opportunities for sharing data via Internet. Physicians are now getting acquiantance with web-based computer technologies. A good number of medical web-sites in general and paediatric web-sites in particular are now available via Internet.

4.5. **Methods of Knowledge Acquisition**

Once the problem domain has been selected, knowledge acquisition is very likely the most important task in an expert system development. The elicitation of knowledge from the expert can be done by various ways. We may classify the methods of knowledge acquisition in three categories : manual, semiautomatic and automatic.

4.5.1. **Manual methods**

Manual methods are basically structured around some kind of interview. The knowledge engineer elicits knowledge from the domain expert and / or other sources and then codes it in the knowledge base.The three major manual methods are interviewing (structured, semistructured, unstructured), tracking the reasoning process, and observing.

4.5.1.1. **Interviewing**

4.5.1.1.1. **Structured interview**

Systematic goal-oriented interview is a structured interview. It establishes an organized communication between the expert and the knowledge engineer. It allows the knowledge engineer to prevent the distortion caused by subjectivity of the domain expert and

structured interview reduces the interpretation problem inherent in unstructured interview. Attention to a number of procedural issues are necessary for structuring an interview. To identify major demarcation of the relevant knowledge the knowledge engineer should study avilable material(s) on the domain. During the knowledge acquisition session he / she should identify target question to be asked. Sample question, questioning techniques, question(s) type and level should be written prior to stuctured interview. Knowledge engineer should follow the guide lines for conducting interviews. During the interview the knowledge engineer uses directional control to retain the interview's structure.

In this study the author himself is a doctor having some amount of domain knowledge, and domain experts are well experienced paediatritians. Moreover during structured interview principal supervisor was readily available. So, knowledge elicitation from domain expert(s) was methodical and easier.

4.5.1.1.2. **Unstructured interview**

As a starting point, informal interviews are conducted for many knowledge acquisition requirements. It helps to get quickly to the basic structure of the domain and saves time. According to McGraw and Harboson-Briggs[3], unstructured interviewing seldom provides complete or well organized descriptions of cognitive processes. Firstly, they observed that the expert system domins are generally complex; thus, the knowledge engineer and the expert must actively prepare for interview situations. Unstructured interviews generally lack the organization that would allow this preparation to transfer effectively to the interview itself. Second, domain expert usually find it very difficult to express some of the more important elements to their knowledge. Third, domain expert may interpret the lack of structure as requiring little preparation on their part prior to the interview. Fourth, data aquired from an unstructured interview are often unrelated, exist at varying levels of complexity, and are difficult for the knowledge engineer to review, interpret, and integrate. A fifth problem cited by McGraw and Harbison-Briggs concerns trainings. Because of a lack of training and experience, few knowledge engineers can conduct an efficient unstructured interview. Thus, knowledge engineers appear disorganized and may unwittingly allow the expert to have low confidence in the knowledge engineer. This may decrease the rapport needed to work togather on a large scale development effort. Finally and most importantly, unstructured situations generally do not facilitate the acquisition of specific information for experts.

4.5.1.1.3. **Semi-structured interview**

Semi-structured interviews are obviously a compromise between structured approach and unstructured approach. This approach is required when complete unfolding of the complexity of the problem domain is not possible.

4.5.1.2. **Tracking the reasoning process**

Tracking the reasoning process refers to a set of techniques that attempts to track the reasoning process of an expert. Cognitive psychologists use this technique in discovering the expert's "train of thought" while he / she reaches a conclusion.

4.5.1.3. **Observations**

Sometimes, it may be possible to observe the expert at work and thereby one can acquire knowledge.

4.5.2. **Semi-automatic methods**

Methods intended to help the knowledge engineers by allowing them to execute the necessary tasks in a more efficient and / or effective manner and also intended to support the experts by allowing them to build knowledge bases with little or no help from knowledge engineers are semiautomatic methods.

4.5.3. **Automatic methods**

In this method the role of the expert is minimal (limited to validation) and there is no need for a knowledge engineer. For example, the induction method can be administered by any builder(e.g., a system analyst).

4.6. **Problems in knowledge acquisition**

4.6.1. **Problems with knowledge acquisition in general**

There are a number of problems with knowledge acquisition [4] mainly concentrating on two aspects : (i) availability of source(s) and (ii) Transferring knowledge. To overcome these problems many efforts have been made [5]. For example, research on knowledge acquisition tools are going on [6] focus on ways to decrease the representation mismatch between the human expert and the program under development. Several expert system development software packages such as EXSYS, Level5 and VP expert greatly simplify the syntax of the rules (in a rule-based system) to make them easier for an expert system builder to create and understand without special training. Also, a natural language processor can be used to translate knowledge.

4.6.2. Problems with Web-based Knowledge Acquisition

- **Finding the proper medical web sites of interest**

Searching by "medical web sites" using *'msn'* search engine one may get the number of search results as 2, 12, 243. The number of search results is 5, 01, 70, 810 with 16 directories using 'Infoseek' search engine. It is reported that the number of web sites is increasing 10% in every month. It is sometimes very difficult to have a comprehensive list of the proper web sites of domain interest. Although one may get some good starting places. Pediatrics, for example, we may start some web sites like : *American Academy of Pediatrics, American Board of Pediatrics, Pediatric Points of Interest.*

- **Internet Information is different from printed Information; Why ?**

1. Lack of quality control at stage of production, leading more easily to lack of reliability.
2. It is possible to read a web page without having seen context pages or the cover page containing disclaimers, warnings etc.
3. Authors of web pages, news articles, e-mails, etc., sometimes remain unidentified.
4. Health information that is valid in a specific healthcare context may be wrong in a different one.

- **Judging the quality of medical information**

The quality of medical information is particularly important because misinformation could be a matter of life or death. Thus studies investigating the "quality of medical information" on the various internet venues - websites, mailing lists and newsgroups, and in E-mail communication between patients and doctors - are mostly driven by the concern of possible endangerment for patients by low quality medical information. Thus quality control measures should be enforced during the knowledge acquisition process with first and foremost objective " *first, do not harm".*

Well known sites such as those of BMJ, JAMA, and Human Genome News are dependable, but what about all the material in usenet groups, listserves, and E-mail messages ? In this respect medicine is closer to astrology than to hard sciences - hence the need for assuring quality [7]. So we should encourage doctors and biomedical researchers, as well as institutions, to comment on what they see on the Internet.

So, the filtering of medical information is particularly important. Differernt tools and techniques are now in proposition. PICS (platform for internet content selection) is one such platform meant for supplying professionals and consumers with labels to help them separate valuable health information from dubious information [8, 9].

4.7. Representative sources used in this work

The following prime sources were used during knowledge acquisition connected to the present research :

- Domain experts :
  1. Dr.(Mrs) Mridula Chatterjee, M.D.(PEAD).
  2. Dr. D. Pal, M.D.(PEAD).
  3. Dr. Salil Dutta, F.R.C.S, F.R.C.O.G.
  4. Dr. Durgesh Rastogi, M.D.(Radiologist).

- Books and Literature :
  1. Nelson Text book of Pediatrics, 16th Edition.
  2. Essential Pediatrics by O.P. Ghai, 3rd Edition, Interprint, New Delhi.
  3. Park's Text book of Preventive and Social Medicine.
  4. IAP Text Book of Pediatrics, Jaypee, New Delhi.
  5. Appendix A

- Sonographic Studies : (Chapter 5)

- Web-sites :

Some of the reviewed web-sites of our interest are given below :

Pediatrics, Perinatal & neonatal medicine reviewed Web sites

*** Best **Very Good * Good

***** Pediatric Health**
Part of the Health Answers site, this section focuses on chilsren's health and health-related topics.
http:/www.healthanswers.com/health_answers/search

**** American Academy of Pediatrics(AAP)**
Provides parents and professionals with information on children's health care and the organization.
http:/www.aap.org/

**** Pediatric Points of Interest**
An online encyclopedia of pediatric links covering diseases, hospitals, parenting resources and more.
http:/www.med.jhu.edu/peds/neonatology/

**** Department of Pediatrics-Loyola University Medical Center**
Information about this perinatal center, including research programs, clinical guidelines, residency programs, fellowship programs, Ronald McDonald House, and related topics.
http://www.meddean.luc.edu/lumen/DeptWebs/peds/ped-hm.htm

****National Association of Neonatal Nurses**
Information about NAN-publications, chapters, accessories, meetings, links, and related topics.
http://www.nann.org/

****Neonatology on the Web: Neonatology Teaching Files, Outlines and Guidelines**
A list of links to online resources for parents and medical practitioners.
http://external.csmc.edu/neonatology/syllabus/syll...

**** NICU-WEB: An On-line Neonatology Resource**
Comprehensive resource on neonatal intensive care issues from the University of Washington Children's Hospital Medical Center.
http://weber.u.washington.edu/d08/neonatal/

**** Pediatric-Perinatal Pathology Index**
Image gallery of newborns' pathologies.
http://www-medlib.med.utah.edu/WebPath/PEDHTML/PED...

4.8. Some knowledge representation (KR) schemes

A knowledge representation method is the way a knowledge engineering models the facts and relationships of the domain knowledge. The two types of knowledge that need to be represented in a computer are declarative knowledge and procedural knowledge. Declarative knowledge signifies facts about objects, events, and about how they relate to each other and procedural knowledge signifies the way to use the declarative knowledge.

In the present state-of-the-art of the subject there is no best theory of knowledge representation and so there is no best way to represent knowledge. Each method has its own advantages and disadvantages.

Several common knowledge representation schemes have been discussed in the literature [4, 10-14] including logic, semantic networks, frames, OAV-triplets, scripts and production rule systems as classical methods; and the relatively new paradigm : object-oriented (O-O) approach.

4.8.1. Knowledge representation using Logic

The formal logic systems used to represent declarative knowledge in AI are propositional calculus, predicate calculus, and first order predicate calculus.

An inference in propositional calculus is as follows :

Tommy is a cat.
If Tommy is a cat then she is a mammal.

In such a case according to an inference rule in propositional calculus known as **modus-ponens** the following must be true :

Tommy is a mammal.

In predicate calculus a generalization like the following is possible.

Tommy is a cat.
All cats are bigger than all mice.

If the above two statements are true then

Tommy is bigger than all mice.

First-order predicate calculus is created by adding functions and some other analytical features. For example. a function "is-owned-by" is represented as

(is-owned-by (Tommy John)).

It may be noted that, the first AI program "The Logic Theorist" was written using formal logic. A somewhat less formal systems of logic, such as fuzzy logic are used to reprsent concepts that are relatively vague and approximate such as "moderately expensive", "somewhat tall", "not so beautiful" etc.

There are merits and dimerits of using logic. The idea of logic, having matured for centuries are understood by intellectual community throughout the world. But, everybody will agree that, only a limited portion of intelligent human behaviour can be described in terms of logic [15]. However, different aspects of logic, along with the proofs and rules of inferences such as **modus-ponens**, **modus tolens, hypothetical syllogism** and **resolution** are treated in different text books on AI and Expert Systems.

4.8.2. Knowledge representation using semantic nets

Semantic nets were originally developed for use as psychological models of human memory but are now a standard representation method for AI and expert systems. A semantic network method represents knowledge using two tuples : (N, L). N is a set of nodes representing objects and descriptors. An object may be a physical or conceptual entity. A descriptor provides additional information about an object. L is a set of links connecting the nodes and representing the relations among them. Mainly three types of links are used : has-a link, is-a link, and definitional links. A has-a link shows that a node has a certain property. An is-a link represents class and instance relationships. A definitional link is used for representing declarative relations.

For example, a simple description of a bluebird might be "a bluebird is a small blue-coloured bird and a bird is a feathered flying vertebrate". This may be represented as a set of logical predicates [16] :

size(bluebird, small).
hascovering(bird,feathers).
hascolour(bluebird, blue).
hasproperty(bird, flies).
isa(bluebird, bird).
isa(bird, vertebrate).

Instead of predicates to indicate relations, this description could also be represented graphically by using the links in a graph (fig.4.1). This description, called a semantic network, is a fundamental technique for representing semantic meaning. Because relationships are explicity denoted by the links of the graph, an algorithm for reasoning about the domain could make relevant associations simply by following links. In the bluebird illustration, a system need only follow two links in order to determine that a

bluebird is vertebrate. This is more efficient than exhaustively searching a data base of predicate calculus descriptions of the form isa (X,Y).

Moreover, knowledge may be organized to reflect the natural class-instance structure of the domain. Certain links in a semantic network (the ISA links in fig.4.1) indicate class membership and allow properties attached to a class description to be inherited by all members of the class. Inheritance is a natural tool for representing taxonomically structured information and ensures that all members of a class share common properties.

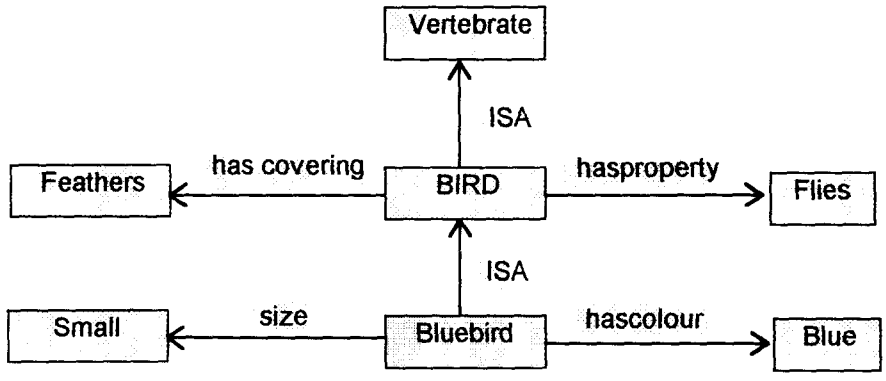


Fig. 4.1. Semantic network description of a bluebird.

4.8.3. **Knowledge representation using rules**

Rules provide a formal way of representing recommendations, directives, or strategies; they are often appropriate when the domain knowledge results from empirical associations developed through years of experience solving problems in an area. A rule has two parts. The first part is a premise of conditions connected by logical-AND or logical-OR relationships. The second part is a conclusion. When the premise of a rule is true, the conclusion of the rule will become true. In some systems rules may be implemented by semantic networks or OAV, as in MYCIN [17], the medical diagnostic system developed at Stanford University. Alternatively, rules may be represented by frames, as in IntelliCorp's knowledge engineering environment [18].

In expert systems jargon the term rule has a much narrower meaning than it does in ordinary language. It refers to the most popular type of knowledge representation technique, the rule-based representation. Rules are expressed as IF-THEN statements, as shown below [19] :

Rule 1. If a flammable liquid was spilled,
called the fire department.

Rule 2. If the pH of the spill is less than 6,
the spill material is an acid.

Rule 3. If the spill material is an acid,
and the spill smells like vinegar,
the spill material is acetic acid.

These are rules that might exist in a crisis management expert system for containing oil and chemical spills. Rules are sometimes written with arrows (→) to indicate the IF and THEN portions of the rules.

Rule 2 in this notation would look like :

If the pH of the spill → the spill material
is less than 6 is an acid.

In a rule-based expert system, the domain knowledge is represented as sets of rules that are checked against a collection of facts or knowledge about the current situation. When the IF portion of a rule is satisfied by the facts, the action specified by the THEN portion is performed. When this happens the rule is said to fire or execute. A rule interpreter compares the IF portions of rules with facts and executes the rule whose IF portion matches the facts, as shown in fig. 4.2.

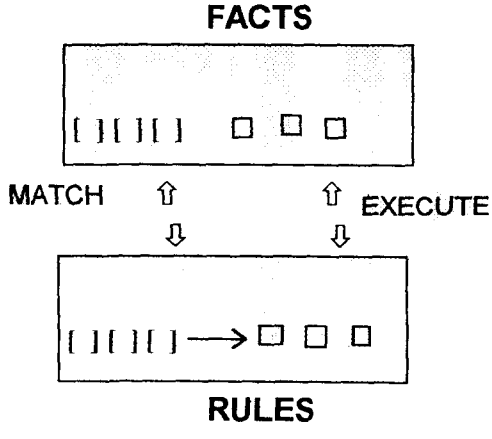


Fig. 4. 2. The rule interpreter cycles through a match-execute sequence.

The rule's action may modify the set of facts in the knowledge base, for example, by adding a new fact, as shown in fig. 4.3. The new facts added to the knowledge base can themselves be used to form matches with the IF portion of rules as illustrated in fig. 4.4. The action taken when the rule fires may directly affect the real world, as shown in fig. 4.5.

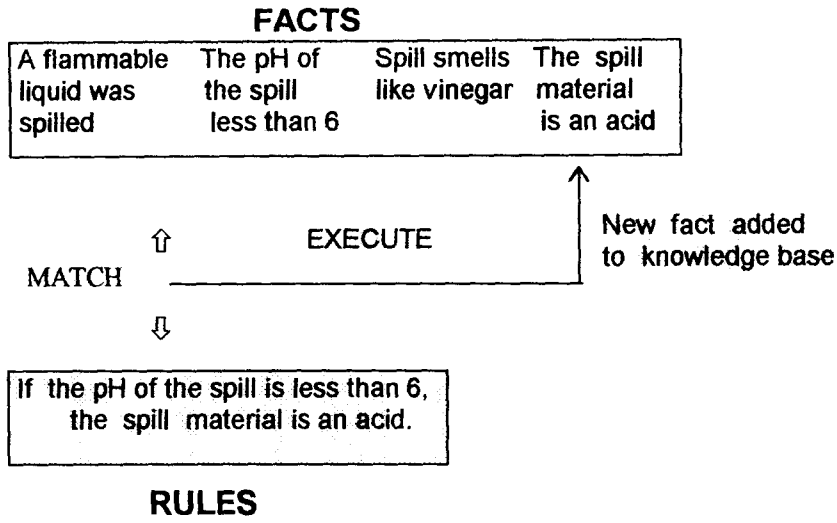


Fig. 4.3. Rule execution can modify the facts in the knowledge base

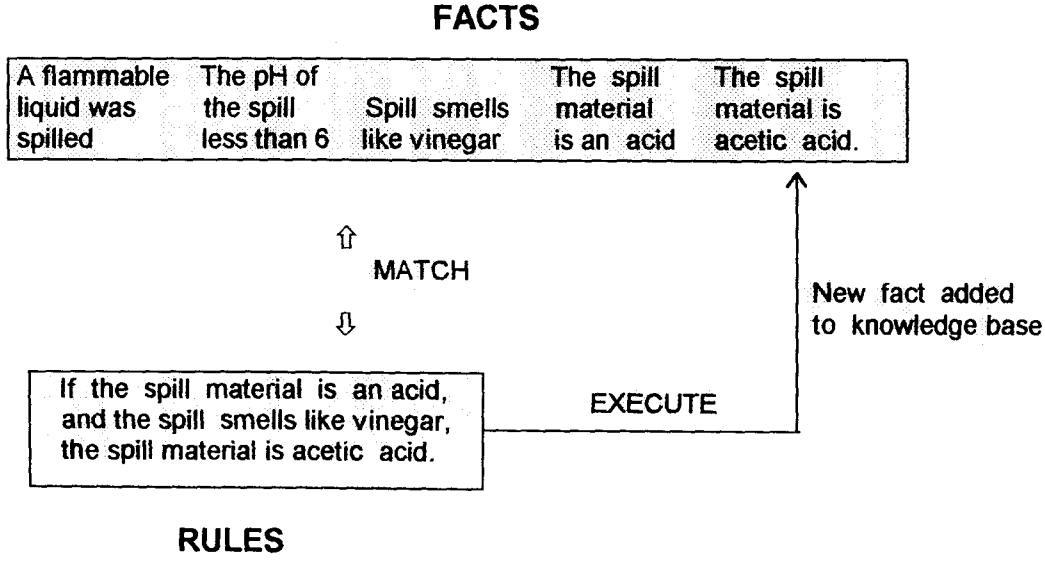


Fig. 4.4. Facts added by rules can match rules

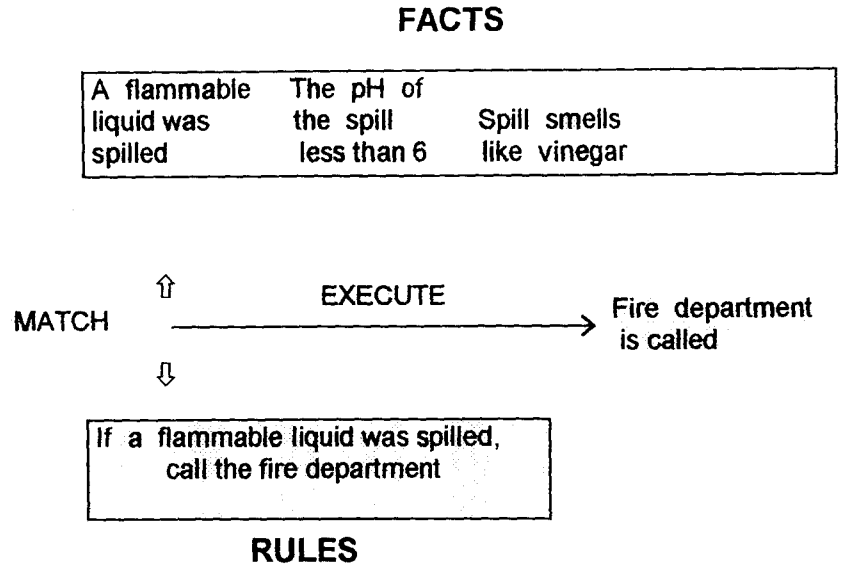


Fig. 4.5. Rule execution can affect the real world

This matching of rule IF portions to the facts can produce what are called inference chains. The inference chain formed from successive execution of rules 2 and 3 is shown in fig. 4.6. This inference chain indicates how the system used the rules to infer the identify of the spill material. An expert system's inference chains can be displayed to the user to help explain how the system reached its conclusions.

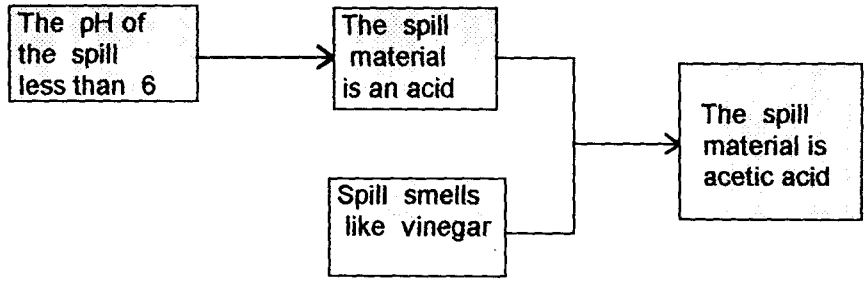


Fig. 4.6. Inference chain for inferring the spill material

There are two important ways in which rules can be used in a rule-based expert system: forward chaining and backward chaining. The spill material example just presented used forward chaining.

Forward chaining is a 'data-driven' approach. In this approach one starts from available information as it comes in, or from a basic idea, then to draw conclusions. Backward chaining is a 'goal-driven' approach in which one starts from an expectation of what is to happen(hypothesis), then seek evidence that supports (or contradicts) his/her

expectation. The inference chain created by backward chaining is identical to the one created by forward chaining; but however, the order and actual number of states searched can differ. The prefered strategy is determined by the properties of the problem itself. These include the complexity of rules, the shape of the state space, and the nature and availability of the problem data. All these vary for different problems.

Depth-first search and Breadth-first search [4]

In addition to specifying a search direction (data-driven or goal-driven), a search algorithm determines the order in which states are examined in the tree or graph. This section considers two possibilities : depth-first and breadth-first search.

Depth-first search

A depth-first search begins at the root node and works downward to successively deeper levels. An operator is applied to the node to generate the next deeper node in sequence. This process continues until a solution is found or backtracking is forced by reaching a dead end.

A simple depth-first search is illustrated in fig. 4.7. The numbers inside the nodes designate the sequence of nodes generated or searched. This process seeks the deepest possible nodes. If a goal state is not reached in this way, the search process backtracks to the next highest level node where additional paths are available to follow. This process continues downward and in a left-to-right direction until the state goal is discovered. Here, the search would actually end at node 13.

When a dead-end node is discovered, such as node 4 in fig. 4.7, the search process backtracks so that any additional branching alternative at the next higher node level is attempted. The search backs up to node 3. It has no alternate paths, so the search backtracks to node 2. Here, another path through node 5 is available. The path through node 6 is explored until its depth is exhausted. The backtracking continues until the goal is reached.

The depth-first serch guarantees a solution, but the search may be a long one. Many different branches will have to be considered to a maximum depth before a solution is reached. (By setting a "depth bound", it is frequently possible to reduce the search.) The method is especially attractive in case where short paths exist and where there are no lengthy sub-branches.

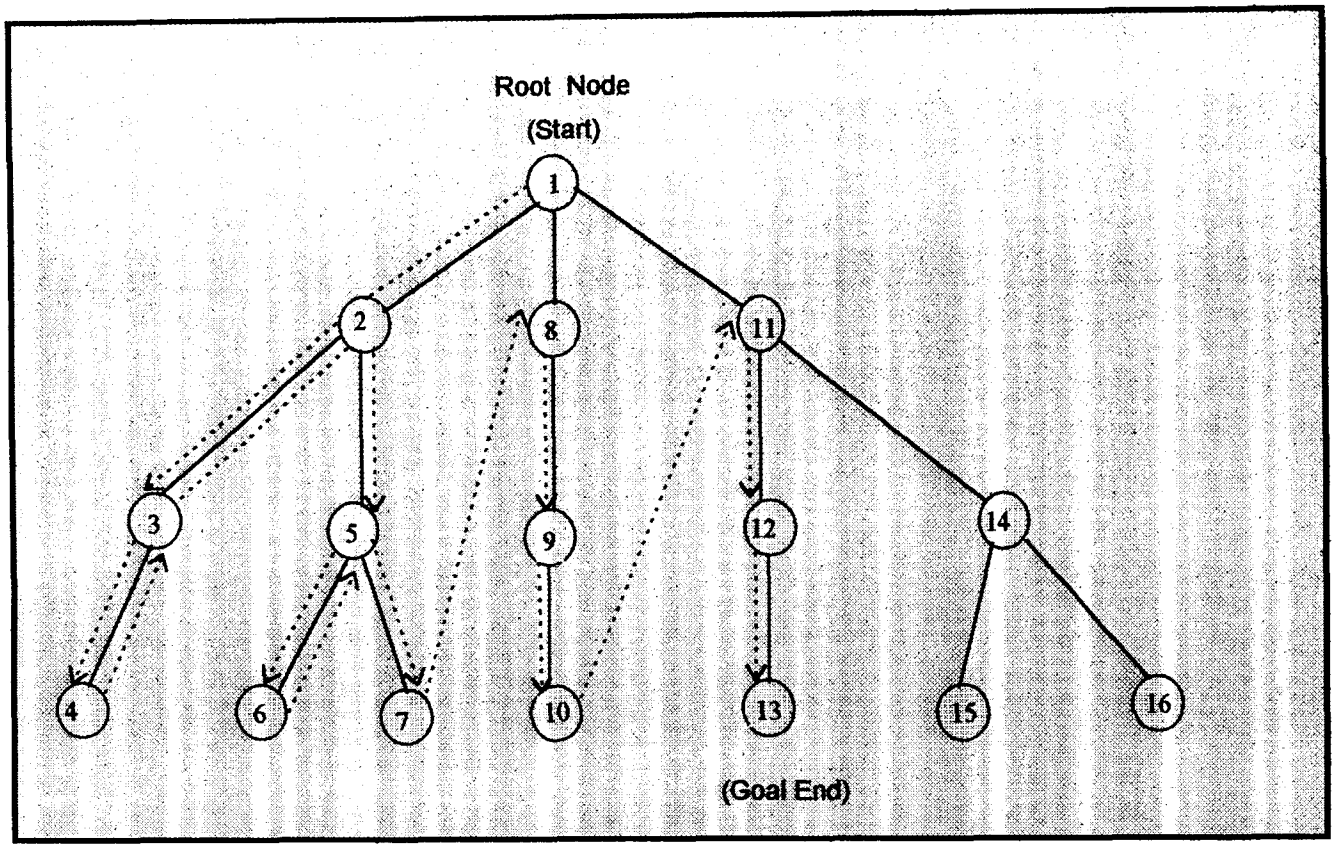


Fig. 4.7 Depth-first search

Breadth-first search

A breadth-first search examines all of the nodes (states in a search tree), begining with the root node. The nodes in each level are examined completely before moving on to the next level. A simple breadth-first search is illustrated in fig. 4.8. The numbers inside the node circles designate the sequence in which the nodes are examined. In this instance, the search (follow the broken line) would actually end at node 7, as that is the goal state.

A breadth-first search of the state space will usually find the shortest path between the initial state and the goal state, with the least number of steps.

The process usually starts at the initial state node and works downward in the tree from left to right. A terminal node is not necessarily a goal node; it can be a dead-end node. Breadth-first procedures are good when the number of paths emanating from each goal is relatively small and where the number of levels in each branch is of a different depth (number of levels).

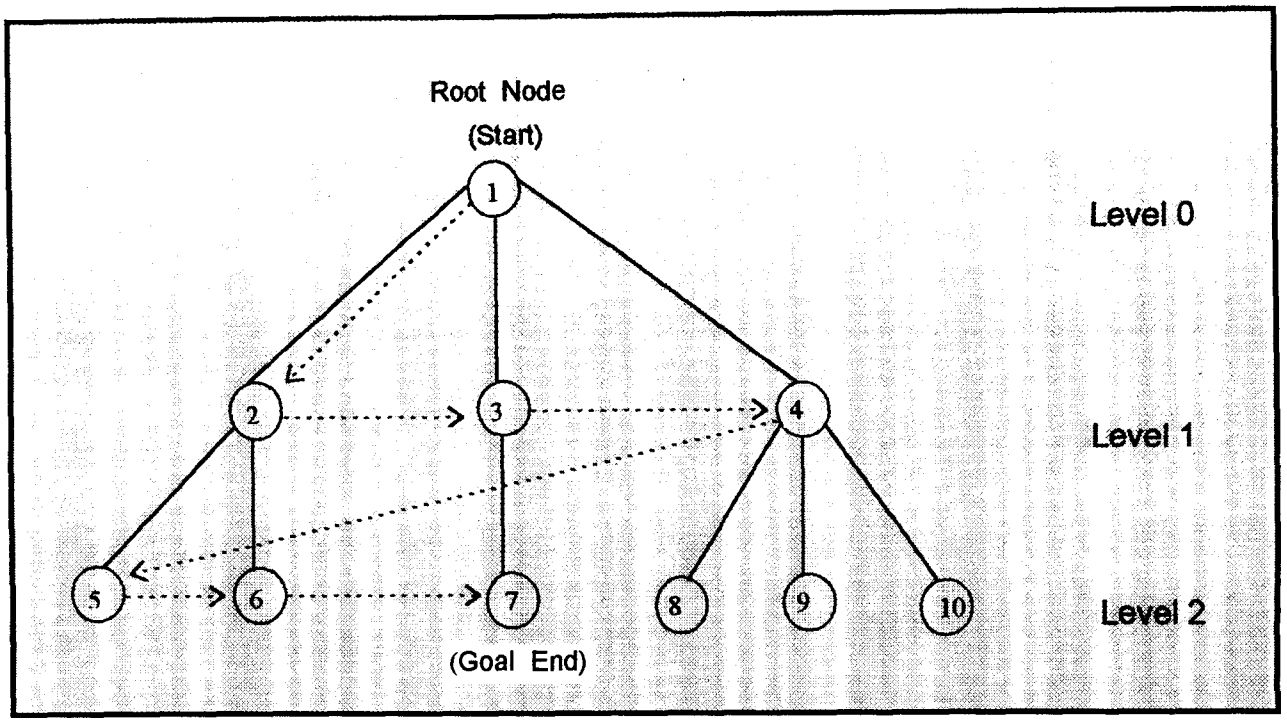


Fig. 4.8 Breadth-first search

4.8.4. **Knowledge representation using frame**

A frame is used to describe an object [20]. It is composed of slots storing information associated with the object. The function of the slots is similar to that of the attributes in OAV. However, frames differ from OAV in that the slots may contain default values, pointers to other frames, sets of rules, or procedures. Frames may also be linked to allow for inheritance. So, frames and OAV can be considered special cases, or subsets, of semantic networks. The representational power of the three systems is the same. The difference lies in the structure and concept of their knowledge organizations.

According to Marvin Minsky [20] when we mentally recall the image of a particular object, we recall a group of typical attributes of that object at the same time - this cohesive grouping of attributes is called Frames. For example, if somebody mentions about a chair, that would "trigger" a series of expectations - such that it is an object with four legs, a seat, a back, and possibly but not necessarily two arms.

A preconception about the colour may not be there but a general expectation of size will be there [12, 13]. A frame of the word " chair" is shown in figure 4.9.

FRAME : CHAIR

PARTS	:	SEAT, BACK, LEGS, ARMS
NUMBER OF LEGS	:	4
NUMBER OF ARMS	:	0 OR 2
DEFAULT	:	0
COLOUR	:	ANY
SIZE (in feet)	:	
HEIGHT	:	2.5 - 5
WIDTH	:	1 - 3
DEPTH	:	1 - 3
STYLES	:	ROCKING, RECLINING, OFFICE

Fig. 4.9 A frame for the word "CHAIR"

4.8.5. **Knowledge representation using scripts**

Another knowledge representation system that is especially useful in the area of natural language understanding is a system called scripts, proposed by Roger Schank [21] at Yale University. Scripts are composed of a series of slots that describe, in sequence, the events that we expect to take place in familiar situations. Just as the concept of frames is based on the assumption that we have a set of expectations about objects, the use of scripts assumes that we also expect certain sequences of events to occur in particular times and places. Schank and Childers in their book [21] the Cognitive Computer uses a resultant script. Visit to a restaurant is composed of a series of scenes, for example, an entering scene, a sitting scene, an ordering scene, an eating scene, a bill payment scene etc. Fig. 4.10 shows a script for entering scene.

SCRIPT : RESTAURANT
SCENE : ENTERING
GO INTO THE RESTAURANT
LOOK AT THE TABLES
DECIDE WHERE TO SIT
GO TO TABLE
SIT.

Fig.4.10 A script for entering a restaurant

4.8.6. Object-attribute-value triplets as KR scheme

The object-attribute-value triplets (OAV-triplets) method represent knowledge using three tuples : (O, A, V). O is the set of objects, which may be physical or conceptual entities. A represents the set of attributes characterising the general properties associated with objects. V (values) specify the nature of the attributes.

The OAV method is actually a special form of semantic network. The relation between an object and an attribute is a has-a link, and the relation between an attribute and a value is an is-a link. The objects, attributes, and values of OAV are equivalent to the nodes in semantic networks. Knowledge can be divided into dynamic and static portions. The triplet values are the dynamic portion. These values may change, but the static portion (usually facts and rules) remains unchanged for different consultations.

An expert system stores data about real-world entities. In knowledge representation theory, the real-world entities are objects. Each object has one or more attributes or properties, and the attributes have values; for example,

Object	Attribute	Value
Bus	Colour	Blue
Amal	Fever	High

The object is **bus**, the attribute is **colour**, and the value is **Blue**. Another example, the object is **Amal**, the attribute is **fever**, and the value is **high**.

OAV is more structured than a semantic network. However, when the number of objects increases, an OAV system becomes difficult to manage.

4.8.7. Object-Oriented approach

The world consists of different objects. An object is an independent entity represented by some data and a set of operations (methods and capabilities) [22]. Therefore, an object can be used to represent a variety of knowledge. Knowledge (K) can be formally represented by three tuples, K = (C, I, A). C is a set of classes represented by class objects. I is a set of instances represented by instance objects. A is a set of attributes possessed by the classes and instances.

4.8.7.1. **Classes**

A class is a description of a group of similar instance objects [22]. It is a mold that determines the behavior of its instances. Each class has a unique name and a set of attributes that define the properties of the class. A class may be a sub-class of another class and may inherit properties from its parent class as discussed in **sub-section** 4.8.7.4.

4.8.7.2. **Instance objects**

Instance objects are members of classes. Their properties are defined by their parent classes. Each instance object consists of three sets of attributes :

(a) Name - the name of the object, which is unique in the system. It is used to identify the object.
(b) Class - the class that contains the object.
(c) Instance attributes - attributes belonging to the instance object. Some operations may be performed on these attributes. The behaviour of the object is determined by the values of these attributes.

4.8.7.3. **Attributes** and **methods / operations**

The property of an attribute is determined by its type and value. The type of an attribute is defined by its class, while the value may be defined in its class or its instance object. A set of generic attributes can be associated with the every object in a class, say furniture, for example. All furnitures has a cost, dimensions, weight, location, and colour among many possible attributes.

Methods are a kind of attribute belonging to objects. They are used to represent capabilities, not to store data, and are defined in classes. Methods cannot be modified during consultations.

4.8.7.4. **Inheritance**

Properties of a class can be inherited from its parent's class. This feature permits factoring knowledge into a class hierarchy. Thus, it encourages modular design of knowledge. The system adopts the inheritance rules, which are similar to those in smalltalk [22], as follows :

(a) If class A inherits from class B, then the objects of class A support all operations supported by objects of class B.
(b) If class A inherits from class B, then class A's attributes are a superset of class B's attributes.

For example, Chair is a member of the class furniture. Chair inherits all attributes defined for the class. This concept is illustrated schematically in fig. 4.11.

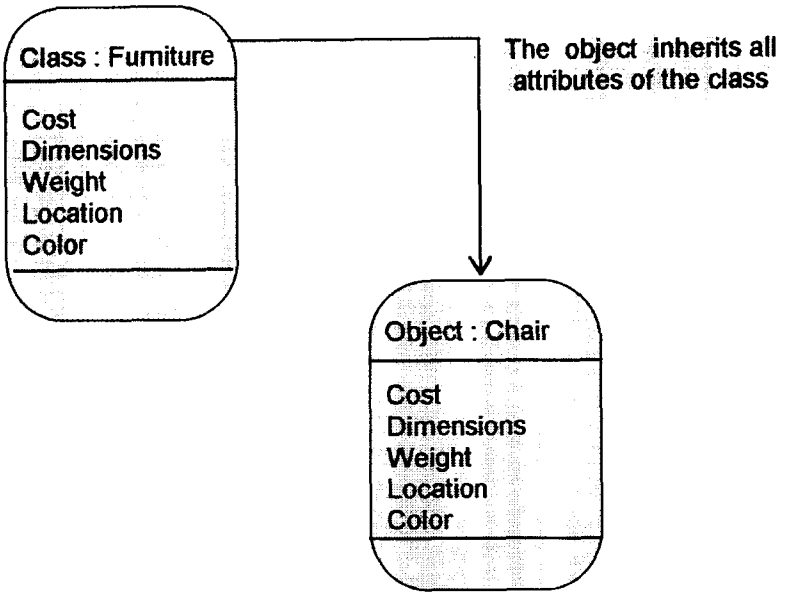


Fig. 4.11 Inheritance from class to object

Once the class has been defined, the attributes can be reused when new instances of the class are created. For example, assume that we were to define a new object called **chable** (a cross between a chair and a table) that is a member of the class **furniture**. **Chable** inherits all of the attributes of **furniture** [23].

Every object in the class furniture can be manipulated in a variety of ways. It can be bought and sold, physically modified, or moved from one place to another. Each of these operations will modify one or more attributes of the object. For example, if the attribute **location** is actually a composite data item defined as :

Location = building + floor + room

Then an operation named **move** would modify one or more of the data items (**building**, **floor**, or **room**) that comprise the attribute **location**. To do this, **move** must have "knowledge" of these data items. The operation **move** could be used for a chair or a table, as long as both are instances of the class **furniture**. All valid operations (e.g. **buy**, **sell**, **weigh**) for the class **furniture** are "connected" to the object definition as shown in fig. 4.12 and are inherited by all instances of the class.

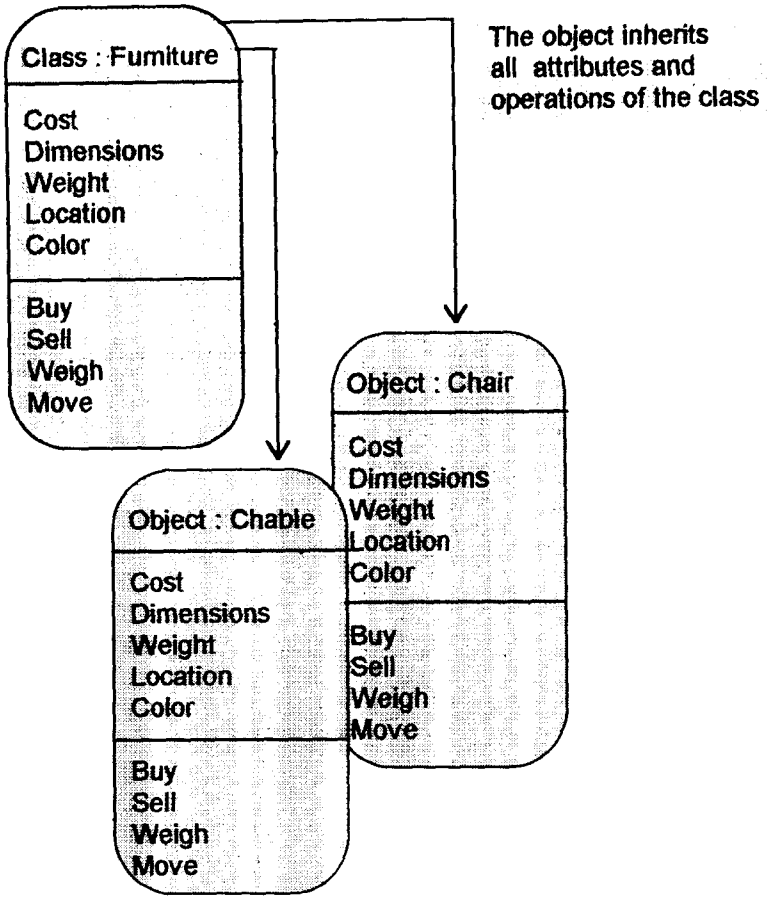


Fig. 4.12 Inheritance of operations from class to object.

The object **chair** (and all objects in general) encapsulates data (the attribute values that define the chair), operations (the actions that are applied to change the attributes of chair), other objects (composite objects can be defined [24]), constants (set values), and other related information. Encapsulation means that all of this information is packaged under one name and can be reused as one specification or program component.

4.8.7.5. **AI, Expert systems** and **O-O technology**

In recent years, the term or adjective 'object-oriented' has become a popular slogan of any kind of systems regardless of its actual qualities. But however, we must admit that behind the slogan there must be some interesting ideas and concepts people suggest for developing large, integrated systems. Although there is the lack of standard definition of what is the O-O approach, but however, the properties like encapsulation, inheritance, polymorphism are considered useful for O-O approach. O-O is now being exploited for analysis and design, and people are sharing their experiences.

The ideas behind object oriented programming (OOP) and object oriented technology (OOT) date back to the forties [25]. These ideas, however, were not put into practice until the introduction of the Simula_67 programming language [26]. Simula, a superset of Algol, was designed for describing a wide class of discrete event simulations and implementing them for simulations. Simula objects represent data and an operation on the data. These objects communicate with each other through messages to determine their next action. Although primitive by today's standards, Simula provided the first insight into the value of OOP.

The form of OOP we are accustomed to seeing took shape in the seventies with the development of Smalltalk at the Xerox Palo Alto Research Centre. Although Smalltalk is used to develop expert systems, its real value is that it offers a user-friendly programming environment.

What made Smalltalk easy to use and conceptually appealing was the extensive use of techniques commonly found today in OOP languages : class / object representations, inheritance, message-passing and encapsulation, to name a few. Researchers at Xerox Palo Alto Research Centre found that these techniques enabled a programmer to easily perceive an object system's structure and operation and to use this understanding to efficiently develop an interface, or for that matter, an entire functioning program. OOP's intuitive approach was the key to Smalltalk's success. Programming solutions frequently followed the methods that humans use to address everyday problems.

Given Smalltalk's intuitive programming environment, coupled with AI researchers' interests in computers representing and reasoning with knowledge similarly to humans, it was only natural for these researchers to adopt object-oriented techniques. This trend was most noticeable during the eighties.

One of the most important events during the eighties that spurred the interest in AI was the marketing of expert system development shells. Most of the early shells were rule-based. However, given the appeal of object-oriented systems, as demonstrated by Smalltalk's success, the demand pushed vendors to offer tools with object-oriented techniques. These tools, commonly called frame-based development programs (but sometimes called hybrid tools), usually combine object-oriented techniques with rule-based programming. New procedural languages with object-oriented techniques also surfaced, such as Objective C, C++, Pascal Object, Modula-2 and Lisp extensions such as Scoops, Flavors, Loops and the Common Lisp Object System(CLOS).

Armed with powerful object-oriented shells and languages, expert system developers took aim at problems that were often out of the reach of rule-based approaches. A review of systems developed during the later eighties and early nineties clearly shows a swing toward object-oriented techniques [27]. This trend was due partly to the

availability of relatively inexpensive frame-based shells that ran on a variety of platforms. Two of the earliest frame-based shells, the knowledge engineering environment from IntelliCorp and the automated reasoning tool from Inference, offered AI researchers powerful tools, but were costly and ran on mainframes or workstations, preventing their widespread use. In the mid-eighties, vendors began marketing cheaper object tools, many of which ran on a PC. This situation led to the accelerated development of frame-based expert systems. Most important, it opened the door at most universities for teaching object-oriented technology (OOT) techniques to the next generation of AI researchers. Flourishing development of object-oriented knowledge-based systems continues. Vigorous development of object-oriented knowledge-based systems continues. Most corporations - including many in the Fortune 500 - are focusing on client-server and object-oriented problems. These organizations have come to recognize AI in general and OOT in specific, as a standard way of doing business. Whereas many of these companies first ventured into AI by forming a dedicated group of AI specialists, most of these specialists now work in the more traditional programming departments, where they routinely carry on their trade of knowledge-based programming.

A look at the recent marketing approach of vendors of AI object-oriented tools is also revealing. As any good business would do, these vendors have kept a finger in the air to sense the direction of their clients interests. They found that although terms such as "AI" and "expert systems" might have fallen out of favour in some circles, their clients' still wanted the object-oriented capability of their products. To go with the flow, these vendors began to advertise their products as "intelligent applications tools". AI capability was still there, but the idea of AI faded into the background. This presents an interesting situation : companies using AI but not promoting it, and vendors marketing products with AI capabilities but not advertising it. Although abandoning the AI label, both have created a new infrastructure on which to build the knowledge-based technology that should flourish in the latter part of the nineties. The irony : even if the spotlight is no longer on AI, AI's contributions will continue to positively affect future information processing, only under other labels [28].

4.9. **Analysing relative suitability**

The major advantage of semantic networks is flexibility, since new nodes and links can be defined as required without restriction. This flexibility also exists in object-oriented (O-O) knowledge representations where, by storing the names of their objects as the attributes of an instance object, relations between instance objects can be established dynamically. These relations have the same power as links in semantic networks; in fact, this object-oriented (O-O) construct can be viewed as dynamic semantic network. The is-a links of semantic networks can be implemented in object-oriented (O-O) representations by relationships between classes and sub-classes or between classes

and instances. Has-a links can be implemented by the relationships between classes and attributes. Therefore, object-oriented knowledge representation has the same power as a semantic network but is much more structured.

A common disadvantage in semantic networks, rules, and OAV representations is that they are not structured enough. A significant increase in the number of objects or rules makes the system difficult to manage. This is because the knowledge cannot be modularized and interactions among rules and objects become too complex. When the value of an object or an attribute is modified, it is difficult to pinpoint the effects on the whole system. Therefore, such knowledge representations are difficult to develop and maintain, especially for a large knowledge base. The encapsulation property and structuredness of object-orienred (O-O) knowledge representations give them a distinct edge over these three representations.

Frames are more structured than semantic networks, rules, and OAV representations, since related attributes and rules can be grouped into frames hierarchically. However, modularity of knowledge represented in frames cannot be clearly defined, and frame representation lacks flexibility. In a frame system, relationships between frames may be member or subclass links and thus are not unique. Moreover, in some systems, a rule is represented by a frame linked to another frame with a special relationship. These factors greatly reduce the structure in a frame system. In object-oriented (O-O) knowledge representation, which is quite similar to frames, knowledge can be arranged in a hierarchical form using classes. However, a subclass link is the only possible relationship between two classes, an is-a link is the only possible relationship between a class and an instance object, and rules are defined as methods in classes - clear cut distinctions that reduce ambiguity and improve understandability.

In tune with our identified key requirements the domain lays on an expert system, we now analyse the relative suitability of different KR schemes discussed in **section** 4.8. When the domain knowledge is vast and varied, the knowledge can become unmanageable. To handle a large knowledge base it is suggested [29] that the structuredness and modularity is necessary where knowledge is varied. A common disadvantage in Semantic Networks, Rules and OAV representations is that they are not structured enough [14]. It is very difficult to manage a system with these representations when the number of objects and rules increases significantly. According to some researchers [30], some applications such as engineering processes, manufacturing and communications are expected to contain 100,000 rules or more. It is then very difficult to pinpoint the effects on the whole system if a value of an object or an attribute is modified.

The major advantage with semantic networks is its flexibility in defining new nodes and links as when required. The type of flexibility is also with the O-O approach which may

be viewed as a dynamic semantic networks. The O-O knowledge representation has the same power as of semantic networks but is much more structured. Frames are more structured than Semantic nets, rules and OAV representations, since related attributes and rules can be grouped into frames hierarchically. This is a passive data structure which lacks flexibility and the relationships within this system are not unique. The active data structure, the O-O representation of knowledge where declarative as well as procedural knowledge can be mixed, is structured and is much more meaningful semantically. The O-O form of KR encourages modular designs supporting the improvement of the efficiency of knowledge acquisition and management. The properties like encapsulation and inheritance of O-O approach are really attractive for large, integrated information systems. The encapsulation property prevents object manipulation except by defined operations. Inheritance is a valuable mechanism which enhances reusability and maintainability of software. Because this approach minimizes object interdependency [31] the knowledge can be structured.

A common disadvantage in OAV triplets and rules is that there may be some redundancy in information which may lead to some inconsistency. There is no such redundancy problem with Semantic Networks, frames and O-O forms. Moreover, O-O approach to KR supports high level of knowledge abstraction, an important advantage over other classical approaches. Considering all these factors, we advocate O-O representation to improve consistency, understandibility, maintainability and modifiability of knowledge base. Last, but not least, in the evolution of an expert system [19], prototyping may have an adverse impact on modifiability and maintainability of knowledge bases since these may be patched and modified several times. This may, however, be overcome by the use of O-O approach. As the system grows, the major changes will be with the addition of new objects or deletion of old objects rather than modifying the old objects. In this respect O-O approach is considered very useful for rapid prototyping, an added advantage.

4.10. **Representative expert systems** and **ES-development tools**

The following table 4.1 represents some expert systems and ES-developmental tools [19, 32-34] with the kind of knowledge representation scheme(s) and control for knowledge base scanning.

Table 4.1 Some ES/ES-tools using different KR-schemes.

ES/ES-tools	Representation	Control
MYCIN	Rule-based	Backward chaining
EMYCIN	Rule-based	Restrictive backward chaining
PROLOG	Logic-based	Backward chaining

EXPERT	Rule-based	Forward chaining
LISP	Procedure-oriented, functional, symbolic expressions	Forward, backward chaining
ADVISOR II	Rule-based	Backward chaining
EXSYS	Rule-based	Backward chaining
GURU	Rule-based	Backward chaining, Limited forward chaining
KES II	Rule-based, classes	Backward chaining, Limited forward chaining
LEONARDO	Rules, Frames, Procedures	Backward and forward chaining
XI PLUS	Rules, Induction	Control over direction
GOLDWORKS	Rules, Frames, Objects	Control over direction
NEXPERT	Rule-based	Forward, backward chaining
ESE	Rule-based	Backward chaining
OPS5	Rule-based	Forward chaining
ART	Rule-based, Frame-based	Forward and backward chaining
DUCK	Logic-based, Rule-based	Forward and backward chaining
GUSS/1	Rule-based	Backward and forward chaining
KES	Rule-based, Frame-based	Backward chainig
M.1	Rule-based	Backward chainig
OPS5	Rule-based	Forward chaining
RITA	Rule-based	Forward and backward chaining
SAVOIR	Rule-based	Backward and forward chaining
S.1	Rule-based, Frame-based	Backward chaining
ARBY	Rule-based	Backward chaining
Plant/cd	Rule-based	Backward chaining
XCON	Rule-based	Forward chaining
XSEL	Rule-based	Forward chaining
YES/MVS	Rule-based	Forward chaining
TALIB	Rule-based	Forward chaining
DELTA	Rule-based	Forward and backward chaining

SPERIL-I	Rule-based	Forward chaining
SPERIL-II	Rule-based	Forward and backward chaining
DIPMETER ADVISOR	Rule-based	Forward chaining
DRILLING ADVISOR	Rule-based	Backward chaining
MUD	Rule-based	Forward chaining
CODES	Rule-based	Backward chaining
FOLIO	Rule-based	Forward chaining
PROJCON	Rule-based	Backward chaining
DSCAS	Rule-based	Forward chaining
SAL	Rule-based	Forward chaining
TAXADVISOR	Rule-based	Backward chaining
IMACS	Rule-based	Forward chaining
PTRANS	Rule-based	Forward chaining
AI/RHEM	Rule-based	Forward chaining
BABY	Rule-based	Forward chaining
CLOT	Rule-based	Backward chaining
MEDICO	Rule-based	Forward chaining
MI	Rule-based	Forward chaining
NEURAX	Rule-based	Forward and backward chaining
ONCOCIN	Rule-based	Forward and backward chaining
PUFF	Rule-based	Backward chaining
SPE	Rule-based	Forward chaining
THYROID MODEL	Rule-based	Forward chaining
WHEEZE	Frame-based	Backward and forward chaining
MES	Rule-based	Forward chaining
TATR	Rule-based	Forward chaining
PDS	Rule-based	Forward chaining
FAITH	Frame-based	Backward and forward chaining
LEVEL5 (OBJECT)	Large-hybrid-object-oriented, Rule-based	Forward and backward chaining

4.11. Paediatric Problem domain

OAV-tripiets

The general form of OAV-triplets representation of the knowledge of **Appendix A** is shown as [35] : (age-of-the-baby, activity, value).

For examples :

(1-month, Axial-muscle-tone, Head-drops),
(1-month, Axial-muscle-tone,Turns-head-from-side-to-side),
(1-month, Spontaneous-Gestures, Athetoid),
(1-month, Spontaneous-Gestures, Stays-lying-on-side),
(1-month, Rhythms-sleep, 21 hours),
(1-month, Rhythms-meals, 5 meals).

Semantic networks

From the **Appendix A**, we show a small portion of the total semantic networks drawn as an example.

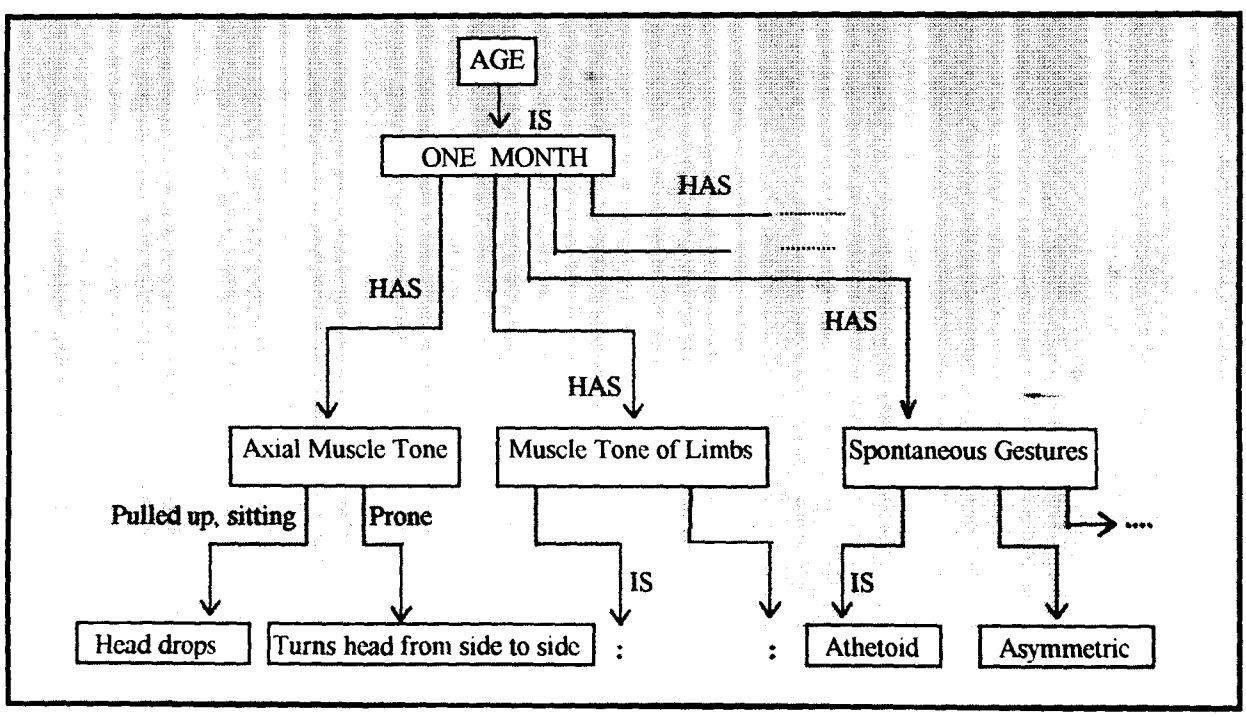


Fig. 4.1 3. Semantic networks.

Rules

In the form of rules the knowledge is represented as :

Rule 1.

If the baby of age between 1 day and 1 month drops his / her head when he / she is pulled up sitting and turns head from side to side when he / she is proned
Then the axial muscle tone is normal.

:

Rule 8.

If the baby of age between 1 day and 1 month sleeps 21 hours and takes 5 meals
Then rhythms are normal.

Frames

The knowledge of **Appendix A** may be represented using frames as :

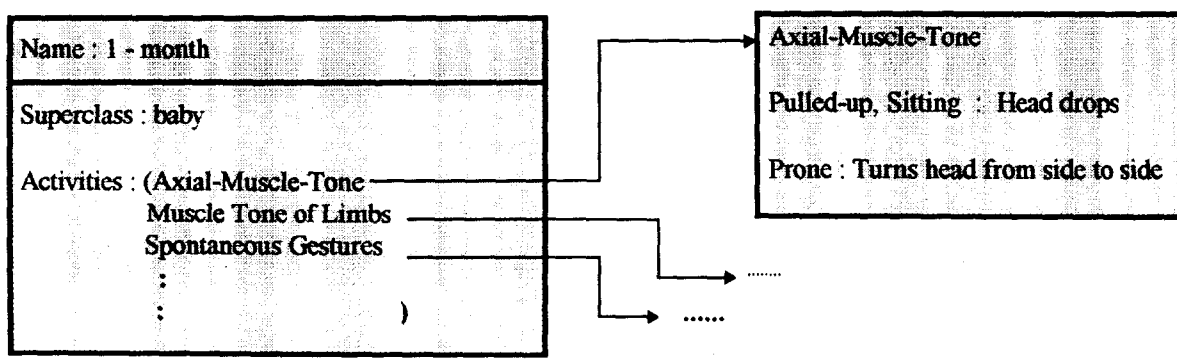


Fig. 4.14. Frame description of baby.

Object-Oriented approach

The knowledge of **Appendix A** may be represented using O-O approach as :

Class name : age
Super class : baby
Instance variables
Head : Limbs : Gestures : Reflexes : Vision : Pulled-up-sitting : Prone : :
Instance methods
Axial-muscle-tone ( ) : begin message (Pulled-up-sitting,head-drops); message (Prone,head-turns-side-to-side); : end. : :

Fig. 4.15. object - oriented representation.

4.12. Discussions

In this chapter, we have considered the vital issues of knowledge acquisition - types and sources. In this connection, we have tried to explore the difficulties associated with knowledge acquisition. Potential sources used in this research have been pointed out. We have tried also to analyse the relative suitability of different KR schemes from the viewpoint of an expert system designer for the paediatric domain. Our analysis finds O-O approach more suitable for the problem domain.

While above analysis and consequent results might lead one to believe that the O-O paradigm is a panacea for all the woes of knowledge engineering / abstraction / representation, the paradigm does have some drawbacks [36] :

- One of object-oriented technology's disadvantage is its long learning curve. The classical developers have to devote several months before they are skilled enough to start a project.

- Second problem may be that is expected in the initial stages of any relatively new technology is the unavailability of robust and reliable tools such as AI-language or a shell. However, at present, there are some ES-shells using object technology (e.g. Level5 object) are coming into the market.

- The third problem may come from the very nature of abstraction. The reliability of the abstraction layer(s) should be sufficiently high so that there should not be any bug with these layer(s). These bugs are rarely trapped by the application layer due to the shielding property of abstraction. A careful design is, obviously, required to overcome this problem.

At this stage, there is no doubt that O-O technology should certainly assist us : (i) in developing a complex system; (ii) in maintaining the system; and (iii) in modifying the knowledge base of a system.

References

1. David S. Prerau. Developing And Managing Expert Systems. Addison-Wesley Publishing Company, inc.; 1990, 200.

2. E. Coiera. Medical Informatics. Australian Medical Journal; 160, 1994, 438 - 440.

3. K. L . McGraw,., and B. K. Harbison-Briggs. Knowledge Acquisition, Principles and Guidelines. Englewood cliffs, N.J. : Prentice-Hall;1989.

4. Efraim Turban. Expert Systems and Applied Artificial Intelligence. Macmillan Publishing Company; New York, 1992.

5. J. H. Boose. A survey of Knowledge Acquisition Techniques and Tools. Knowledge Acquisition; 1, March 1989.

6. K. Bogel. Design and Implementations of a Web-based knowledge acquisition toolkit for medical expert consultation systems. Doctoral Thesis, technical University of Vienna; 1997.

7. S. Arunachalam. Assuring quality and relevence of Internet information in the real world. British Meddical Journal; 317, 1998, 1496-1502.

8. J. Miller, P. Resnick and D. Singer. Rating services and rating systems (and their machine-readable descriptions). World Wide Web J; 1996, vol.1, no.4, 28 th August 1997, 23-43.

9. P. Resnick and J. Miller. PICS : Internet access controls without censorship. Commun ACM 1996; vol. 39, no.10, htm 28 Aug 1997, 87-93.

10. N. J. Nilsson. Principles of Artificial Intelligence. Narosa Publishing House; New Delhi, 1990.

11. Elaine Rich and Kevin Knight. Artificial Intelligence (2nd ed.). Tata McGraw-Hill; New Delhi,1991.

12. Henry C. Mishkoff. Understanding Artificial Intelligence. H. W. Sams and Co.; 1985, 9-10.

13. Eugene Charniak and Drew McDermott. Introduction to Artificial Intelligence. Addison-Wesley; 1985.

14. K. S. Leung. and M. H. Wong. An expert system shell using structured knowledge : An object-oriented approach. IEEE Computer; vol. 23, no.3, March 1990, 38-47.

15. D. Dutta Majumder. Artificial Intelligence and Expert Systems : its role in Industrial System Development. National Symp. on Elec. and Appl; September 1989.

16. George F. Luger and William A. Stubblefield. Artificial Intelligence and the design of expert systems. The Benjamin / Cummings Publishing Co.; Inc., 1989, 33-35.

17. M. Van et al. Emycin manual. Tech. report. Heuristic programming project; Stanford University, 1981.

18. F. R. Kehler. The role of frame-based representation in reasoning. Comm. ACM; vol.28, no.9, Sept. 1985, 904-920.

19. D. A. Waterman. A Guide to expert systems. Addison-Wesley, Reading, Mass; 1986.

20. Marvin Minsky. A framework for representing knowledge. The Psychology of Computer Vision, (ed.), P. H. Winston. McGraw Hill; New York, 1975.

21. Roger C. Schank and Peter G. Childers. The Cognitive Computer. Addison-Wesley; 1984.

22. A. Goldberg and D. Robson. Smalltalk-80 : The language and its Implementation. Addison-Wesley; NY, 1983.

23. R. S. Pressman. Software Engineering : A practitioner's approach. 3rd edn., McGraw-Hill: Inc., 1992.

24. Object-Oriented Requirements Analysis (course notebook). EVB Software Engineering. 1989

25. K. Nygaard. Basic concepts in object-oriented programming. Sigplan Notices; vol.21, no. 10, October 1986, 117.

26. O. Dahl and K. Nygaard. Simula - A language for programming and description of discrete event systems. Introduction and Users Manual. Norwegian Computing Centre, Oslo, Norway, 1967.

27. J. Durkin. Expert Systems : Catalog of Applications. Intelligent Computer Systems; Inc., Akron, Ohio, 1993.

28. J. Durkin. Expert Systems : A view of the field. IEEE Expert; April 1996, 56-63.

29. W. B. Gevarter, The nature and evaluation of commercial expert system building tools. IEEE Computer; vol.20 no.5, 1987, 24-41.

30. F. Hayes-Roth. Invited Talk. IEEE Compcon; San Francisco, CA, February 1987.

31. A. Synder. Encapsulation and inheritance in object-oriented programming languages. Proc. OOPSLA, ACM; New York, 1986, 38-45.

32. James L. Alty. Expert System Building Tools. Topics in Expert System Design; G. Guida and C. Tasso,(ed.), Elsevier Science Publishers B. V.; North-Holland, 1989, 193.

33. F. Hayes-Roth, Donald A. Waterman and Douglas B. Lenat. An overview of Expert Systems. Building an Expert System, Part II; 26.

34. P. H. Winston and B. K. Horn. LISP. (2nd ed.). Addison-Wesley, Reading, Mass.; 1985.

35. M. Chatterjee, A. K. Saha, M. G. Goswami and R. K. Samanta. Artificial intelligence and expert systems technology : Its role in medicine. Converse : A Themis Pharmaceuticals; India, vol.1, no.2 1996, 13-22.

36. R. Gupta, W. H. Cheng, R. Gupta, I. Hardonag and M. A. Breuer. An object-oriented VLSI CAD Framework : A case study in rapid prototyping. IEEE Computer; May 1989, 28-37.

FOETAL GROWTH PERFORMANCE USING SONOGRAPHIC APPROACH †

5.1. Introduction

For its safe and good outcome the determination of foetal status is very important. For example, accurate determination of gestational age is required in some critical cases. Ultrasonographic imaging technique is now being used for more accurate determination of foetal status and doctors rely more on this technique.

The foetal status of a concieved mother is determined mainly by the sonographic parameters such as BPD, AC, FL, CRL, HC. The measurable parameters are compared with the standard taken at different weeks of gestation of Western pregnant women. From this one can identify the discrepancy from the set standard of the foetus under observation. At present we have standard data from Osaka University, Japan, Britain, USA, Germany on which some important features like gestational age (GA), foetal body weight (FBW) etc. are derived. Regression models are used in different sonographic equipments based on Western data for such prediction. But the question arises whether these standards are appropriate for all population groups ? Some studies insist that the same growth standards should be adopted for all population [1], ethnic factors seem to have little influence, if any, on growth potential before puberty [2]. But, various other studies have shown that maternal size has a pronounce effect on foetal growth [3]. Widdowson[3] suggested that the placental blood flow might be the regulator of foetal growth. This blood flow, the regulator of oxygen supply to the foetus, depends on the surface area which in turn depends on the size of the mother. This factor i.e. maternal size is not uniform for different population group. So one cannot expect similar foetal growth statistics in populations with different maternal size. Keeping all those factors in mind, this study has been undertaken to compare the statistics on foetal growth by ultrasonographic measurements of Indian mothers with Western studies. It should be noted here that during the knowledge acquisition phase for our foetal module (chapter 10), we found no earlier studies on foetal growth performance of North-Bengal districts of India using sonographic technique. This study was conducted as a part of knowledge acquisition phase in connection to the expert system development incorporating the features of the region.

---

† This is based on the publications [41st All India Conf. on Obst. & Gynaecology, Dec,1997, N.B. Medical College, Siliguri, 95-96; Proc. XXIInd Int. Cong. of Pediatrics, Amsterdam, 9-14 August, 1998, 091] of the author.

5.2. Study Design and Sample Size

The present study has been carried out in an upgraded Clinic and Research Centre, Siliguri, Darjeeling. The study has been carried out during last 4 years. One handed (the author of the thesis) experiment was undertaken. Only the pregnant mothers of affluent population segment were selected without any obvious socio-economic constraints. Criteria of affluence is mainly on the income level, life style, educational status and awareness of basic principles of health. Only the "singletone normal" mothers having no ailments or disabilities were included in the study. The analysis of the data was done in the Department of Computer Application, North Bengal University.

The total sample consists of 1747 cases from 14 to 40 weeks of gestational age. The relation between the BPD, FL, AC, HC and menstrual age were determined by cross sectional analysis of these 1747 cases using real time sonography.

The predicted values at various points of gestation were compared with the results of investigations of Western studies and one Indian study; i.e. Johnson [4], Rajan et al. [5], Jeanty et al.[4,6], Chitty [7], Tamura et al.[8], Hadlock et al. [8-10], Sabbagha et al.[5] Shepard [5]. The data were only taken from those mothers who were sure of their LMP. The patient with maternal diseases known to have effect on foetus e.g. diabetes, cardiac disease, hypertension, renal diseases etc. and patient with multiple gestation were excluded from the study.

The examination was done by Linear array (real type of 3.5 MHZ) transducer. The mean value and standard deviations were calculated at weekly interval using standard methods. Each interval were centered on the week (e.g. 20 weeks interval = 19.5 weeks to 20.49 weeks).

Although it is emphasized that the growth curves obtained from longitudinal studies are better than cross-sectional studies, but, however, it is practically not feasible to do the sonographic studies on statistically significant number of mothers every week from 14 week gestational age till the birth of the baby. The error caused by cross-sectional data can be minimised by taking a good number of data.

5.3. Smoothing of Data

Sonographicallly measured foetal parameters were smoothed using polynomials in gestational age. That is, a polynomial of the form

$$Y = a + bx + cx^2 + dx^3 + \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots + kx^n$$

was fitted to the estimated mean values, where Y denotes the estimated mean values of different foetal parameters and x denotes the specified gestational age. Necessary computations were done using orthogonal polynomials. Using analysis of variance technique, goodness of fit of the polynomial of each degree was made through testing for additional information in the orthogonal polynomials of higher degree. Optimal equations as found are presented as follows :

Y (BPD of foetus) = $-23.391 + 3.523x + 0.015x^2 - 0.0007x^3$; 1 SD = 0.503; $R^2 = 0.999$ ..... (1)

Y (FL of foetus) = $-39.525 + 4.304x - 0.0365x^2$; 1 SD = 0.342; $R^2 = 0.999$ ..... (2)

Y (AC of foetus) = $-11.545 + 1.468x - 0.0084x^2$; 1 SD = 0.145; $R^2 = 0.999$ ..... (3)

Y (HC of foetus) = $-12.556 + 1.803x - 0.0162x^2$; 1 SD = 0.276 ; $R^2 = 0.999$ ..... (4)

GA(using BPD) = $1.945 + 5.836\,BPD - 0.652\,BPD^2 + 0.0495\,BPD^3$; 1 SD = 0.198; $R^2 = 0.999$ ..... (5)

GA(using FL) = $9.611 + 3.349\,FL - 0.127\,FL^2 + 0.030\,FL^3$; 1 SD = 0.212 ; $R^2 = 0.999$ ..... (6)

GA(using AC) = $7.451 + 0.905\,AC - 0.006\,AC^2 + 0.0002\,AC^3$; 1 SD = 0.156 ; $R^2 = 0.999$ ..... (7)

GA(using HC) = $4.242 + 1.238\,HC - 0.031\,HC^2 + 0.0008\,AC^3$;1 SD = 0.216 ; $R^2 = 0.999$ ..... (8)

5.4. Results and Discussions

5.4.1. Importance of growth charts

Before we put our results and comparative statements, it would be pertinent to have some discussions on the importance of the following growth charts used in this study.

- **General**

1) The growth chart depict average and permissible range of variation for the particular gestational age.

2) Growth measurement if recorded over a period of time and plotted, the deviation in the growth profile of a term baby from normal pattern for that gestational age can easily be interpreted.

3) Intra-uterine growth chart is a good epidemiological tool for early diagnosis of IUGR, occult intra-uterine infection, or other adverse intra-uterine environmental factor.

4) The standard deviation, the degree of dispersion or the scatter of observation away from the mean and a value beyond two standard deviation is unusal in a normal foetal growth.

- **Importance of the growth velocity**

1) The measurement of velocity of growth in unit time better indicates early identifications of factors affecting growth;

2) It is also important for medical, psychological, social, and remedial measures;

3) This is important because comparison of a foetal growth with the norms based on analysis of the anthropometric data obtained from cross-sectional studies reflects whether the particular foetus is within the expected normal range for gestational age. But it does not give a clue whether the growth rate of the foetus has been normal in the recent past. This is even more clear while visualizing a chart of acceleration / retardation. It also recognizes recent disturbance in the growth performance.

- **The importance of acceleration / retardation and growth shock chart**

1) These are tangible, visible attributes;

2) It creates a felt need demand for growth;

3) It can illustrate the adverse effects of various negative events on growth; for example, infection, maternal depression, nutritional factors etc.;

4) The acceleration/retardation chart may be helpful for both the mother as well as the physician to visualize and motivate the concern for growth of foetus. This could also act as operational strategy for the assessments of growth and development of the community in order to promote the optimal health.

5.4.2. **Femur Length (FL)**

The distribution of the mean femur length data (± 2 SD) as a function of menstrual age is indicated in fig. 5.1. The predicted femur length values for specific points in gestation (equation 2) are compared with the data from other investigators in fig. 5.2. Comparisons of femur length velocity values, acceleration/retardation values, shock values, and trend values are presented in figs. 5.3, 5.4, 5.5, and 5.6 respectively.

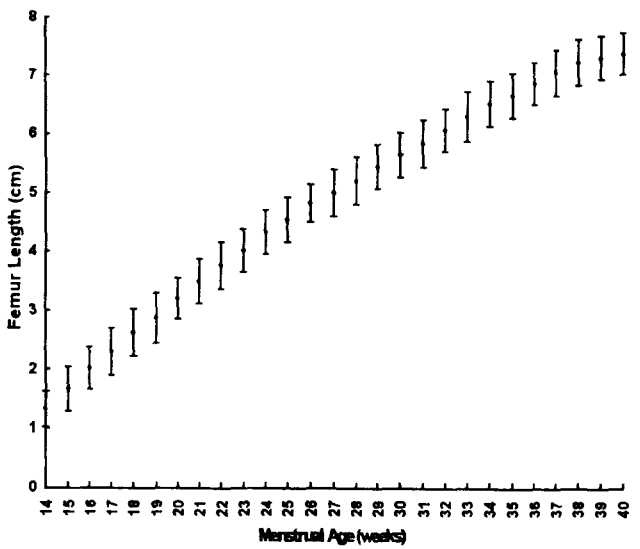


Fig. 5.1. Mean (± 2SD) Femur Length values observed at specific weeks in gestation.

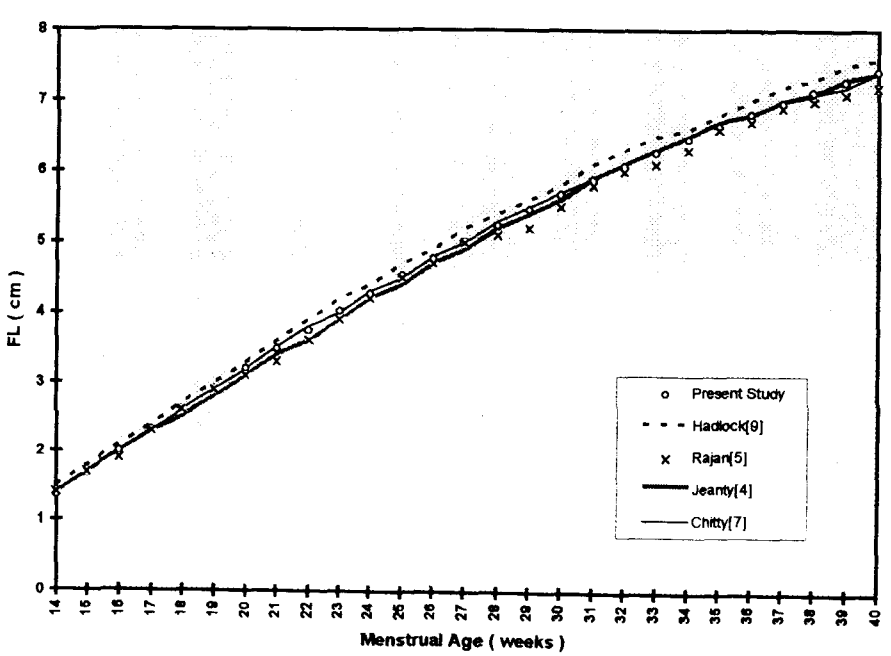


Fig.5.2. Comparison of Femur Length values with Indian and Western Studies.

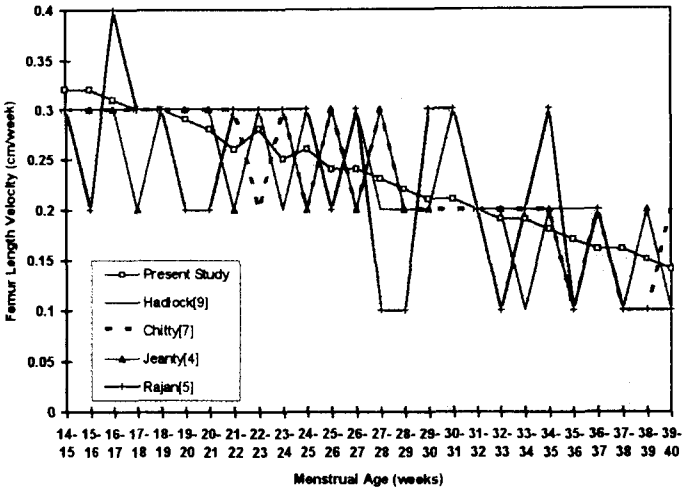


Fig. 5.3. Comparison of Femur Length Velocity values with Indian and Western Studies.

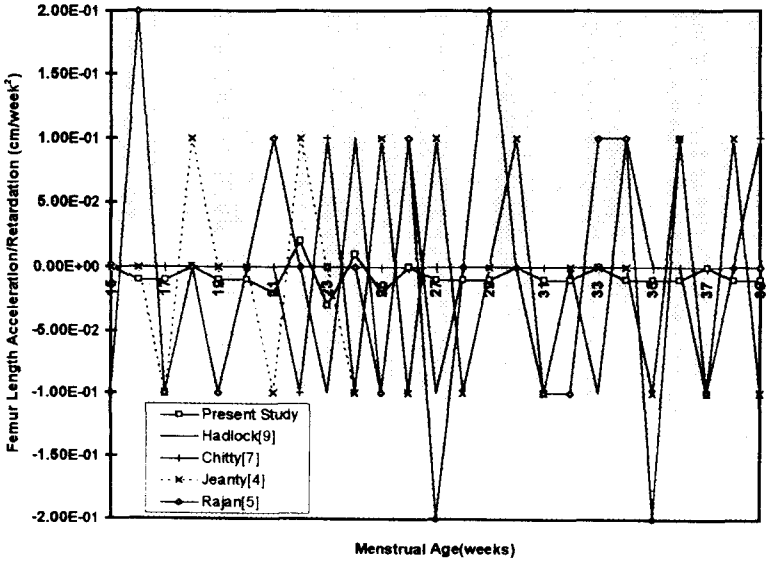


Fig. 5.4. Comparison of Femur Length Acceleration/retardation values with Indian and Western Studies.

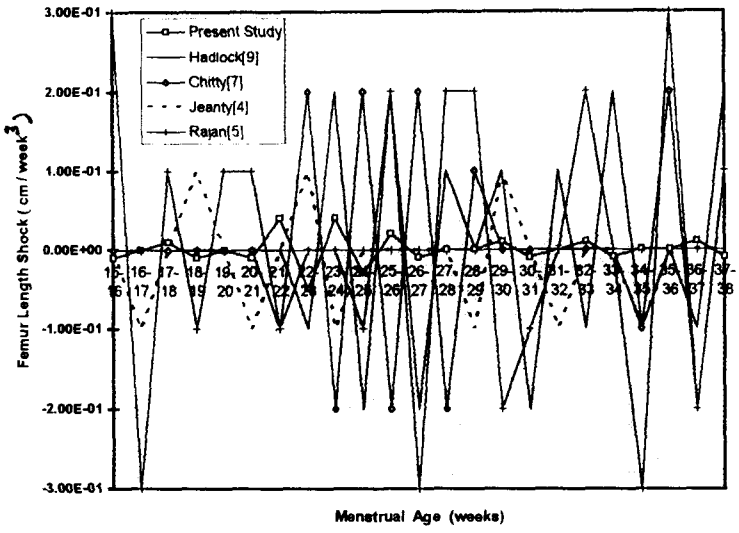


Fig. 5.5. Comparison of Femur Length Shock values with Indian and Western Studies.

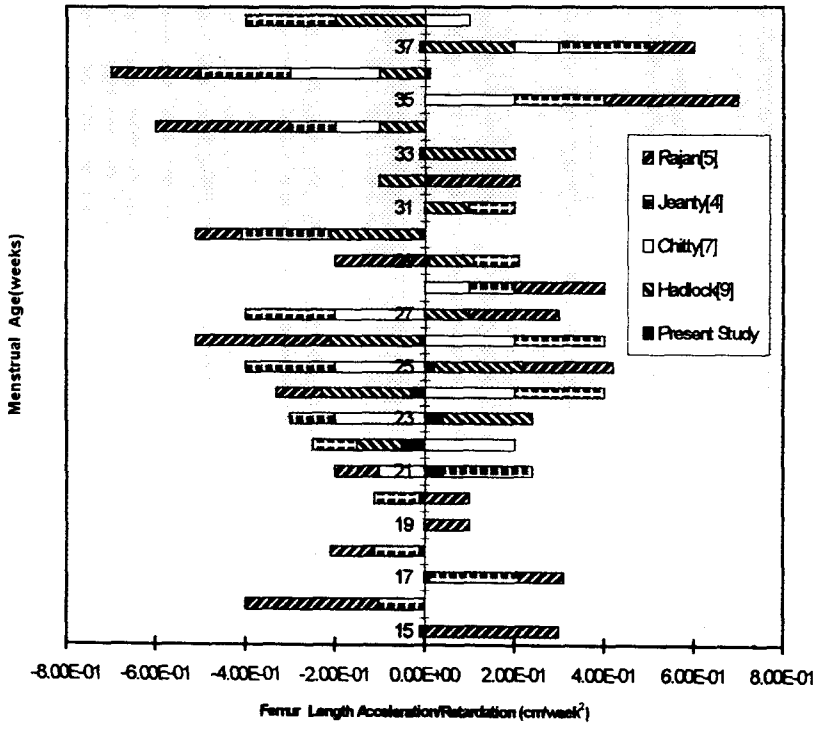


Fig. 5.6. Comparison of Femur Length Trend values with Indian and Western Studies.

The comparison of femur length shows some difference specially at the later weeks of gestation in comparison with Hadlock et. al. But it is almost similar with other studies. The rate of change of femur length per week shows almost similar pattern when compared with Western studies.

5.4.3. **Biparietal Diameter (BPD)**

The distribution of the mean biparietal diameter data (± 2 SD) as a function of menstrual age is indicated in fig. 5.7. The predicted biparietal diameter values for specific points in gestation (equation 1) are compared with the data from other investigators in fig. 5.8. Comparisons of biparietal diameter velocity values, acceleration/retardation values, shock values, and trend values are presented in figs. 5.9, 5.10, 5.11, and 5.12 respectively.

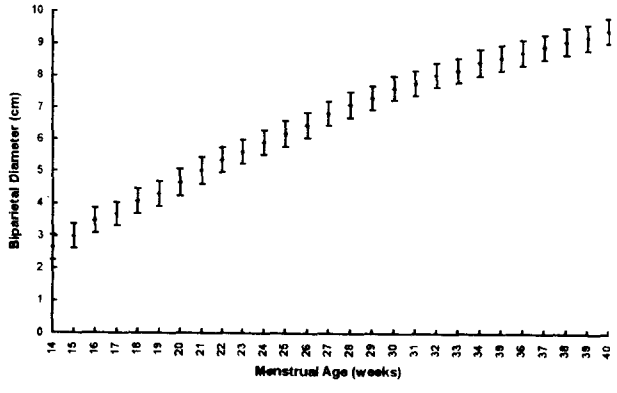


Fig. 5.7. Mean (± 2SD) Biparietal Diameter values observed at specific weeks in gestation

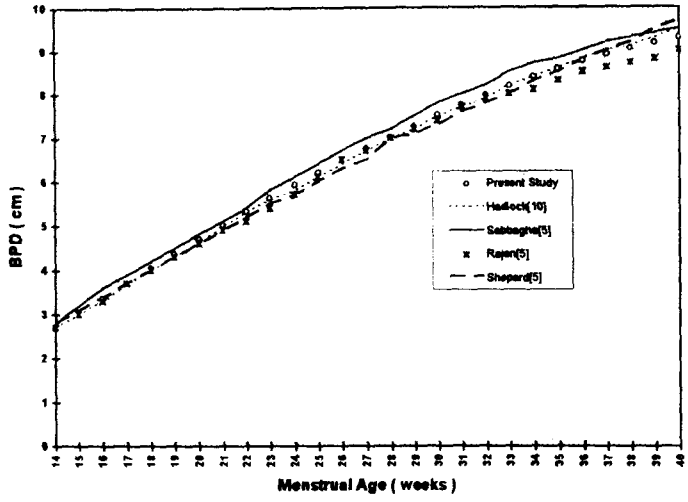


Fig. 5.8. Comparison of BPD values with Indian and Western Studies.

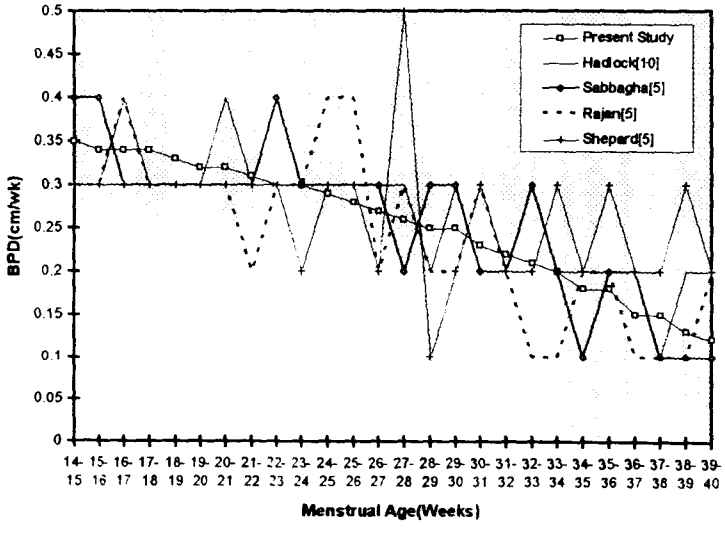


Fig. 5.9. Comparison of BPD velocity values with Indian and Western Studies.

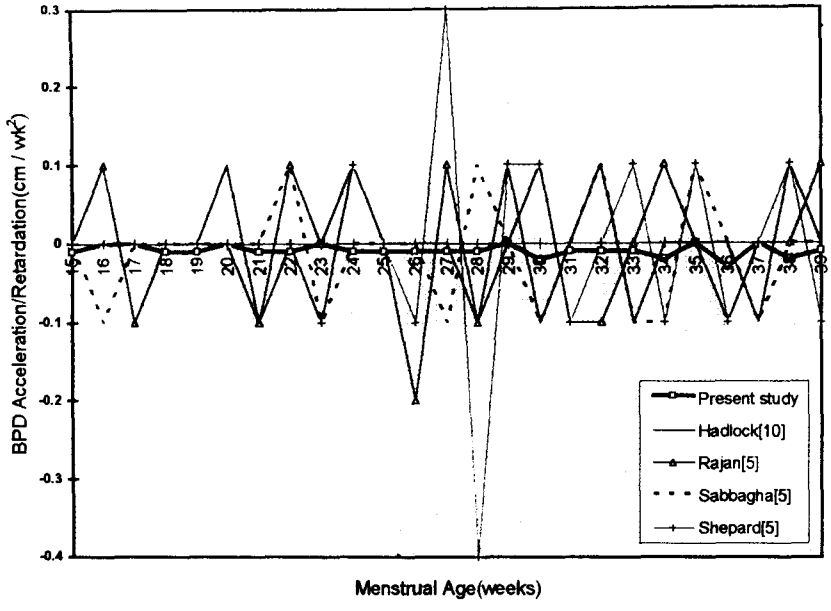


Fig.5.10. Comparison of BPD Acceleration/Retardation values with Indian and Western Studies.

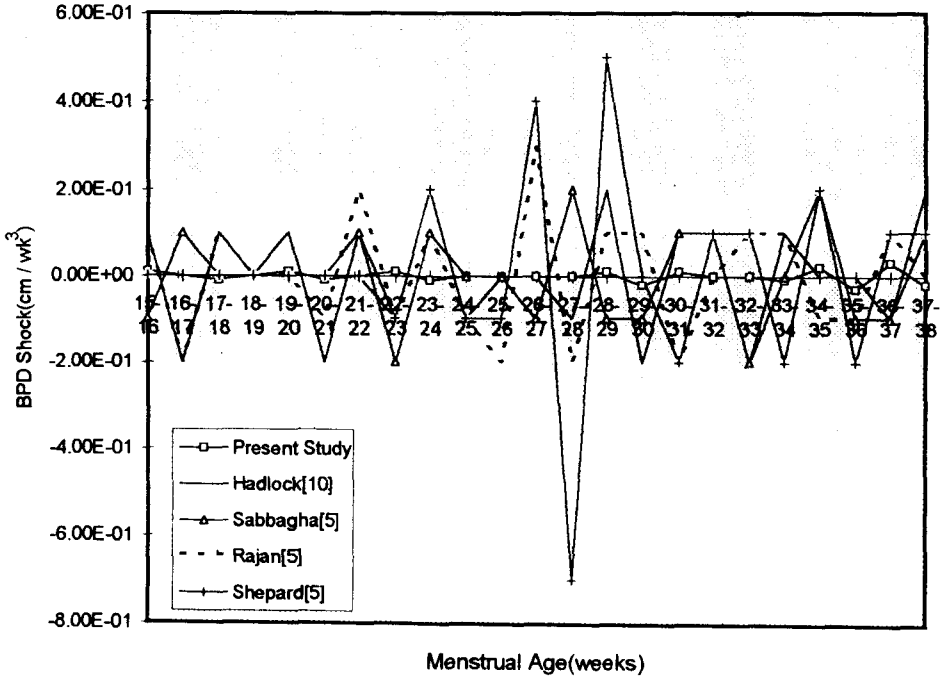


Fig. 5.11. Comparison of BPD Shock values with Indian and Western Studies.

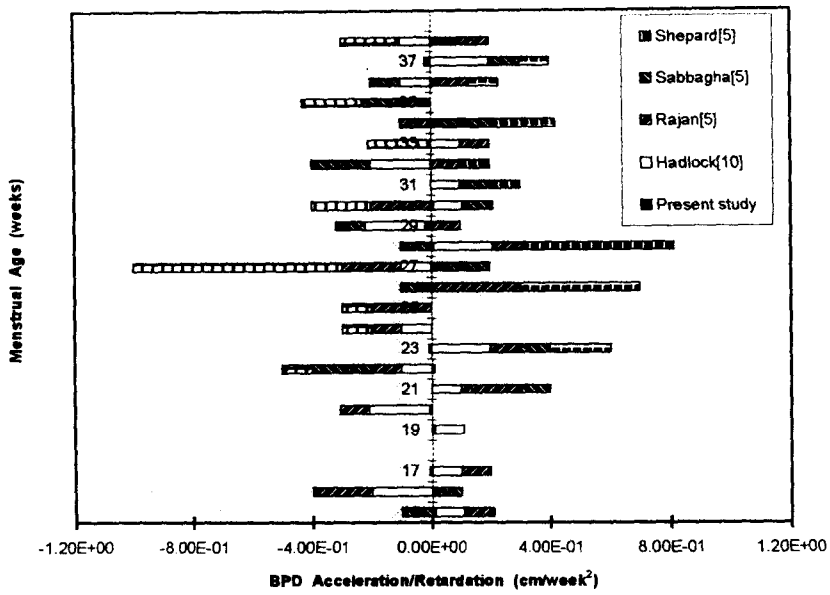


Fig. 5.12 Comparison of BPD Trend values with Indian and Western Studies.

Proper imaging of biparietal diameter requires foetal head to be in an occipito transverse position. The measurement of biparietal diameter is marked with electronic calipre from the outer margin of the proximal skull table to the inner margin of distal skulltable that is leading edge to edge. A comparison of the present study (fig. 5.8) with the others shows that biparietal diameter remains almost same (except Sabbagha et. al.[5] which shows higher values of BPD than Indian studies) till 36 weeks of age. After which there is a faultering which is marked at 39 and 40 weeks. When compared with Rajan et. al. [5], the present study shows slightly higher value. The reason could be present study of higher socio-economic group limiting the nutritional constraint. The comparison of biparietal diameter velocity (fig.5.9) also shows uniform mild faultering at the later weeks of gestational age.

5.4.4. **Abdominal Circumference**

The distribution of the mean abdominal circumference data (± 2 SD) as a function of menstrual age is indicated in fig. 5.13. The predicted abdominal circumference values for specific points in gestation (equation 3) are compared with the data from other investigators in fig. 5.14. Comparisons of abdominal circumference velocity values, acceleration/retardation values, shock values, and trend values are presented in figs. 5.15, 5.16, 5.17 and 5.18 respectively.

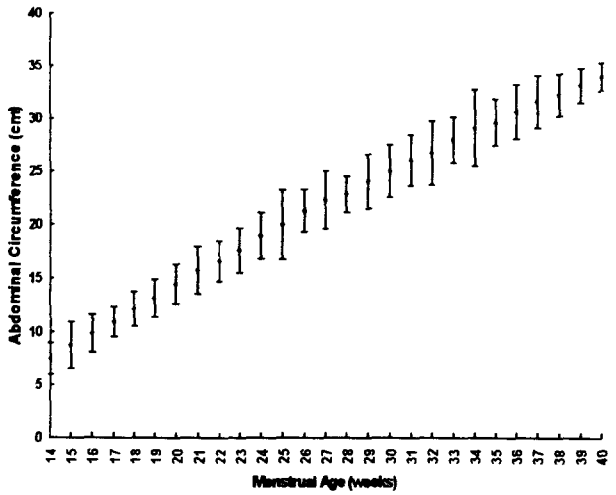


Fig.5.13. Mean (± 2SD) Abdominal Circumference values observed at specific weeks in gestation.

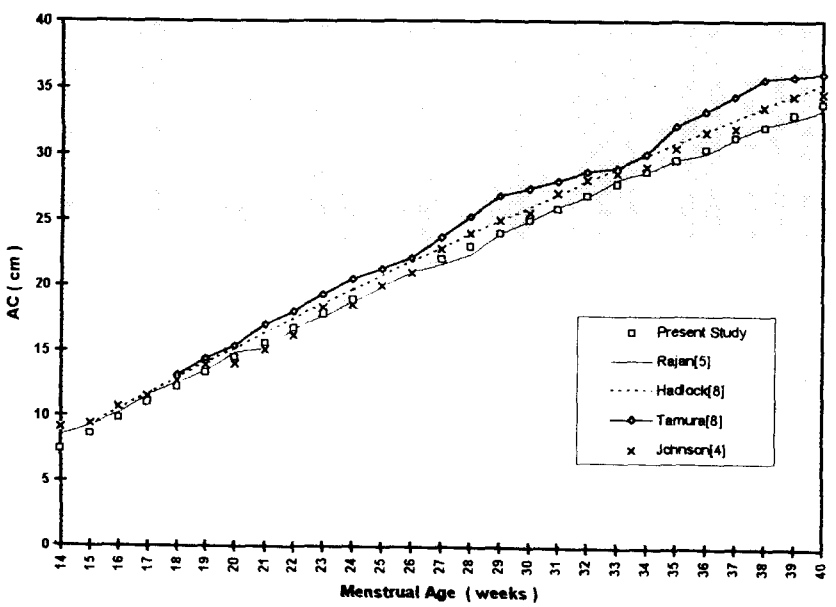


Fig. 5.14. Comparison of Abdominal Circumference values with Indian and Western Studies.

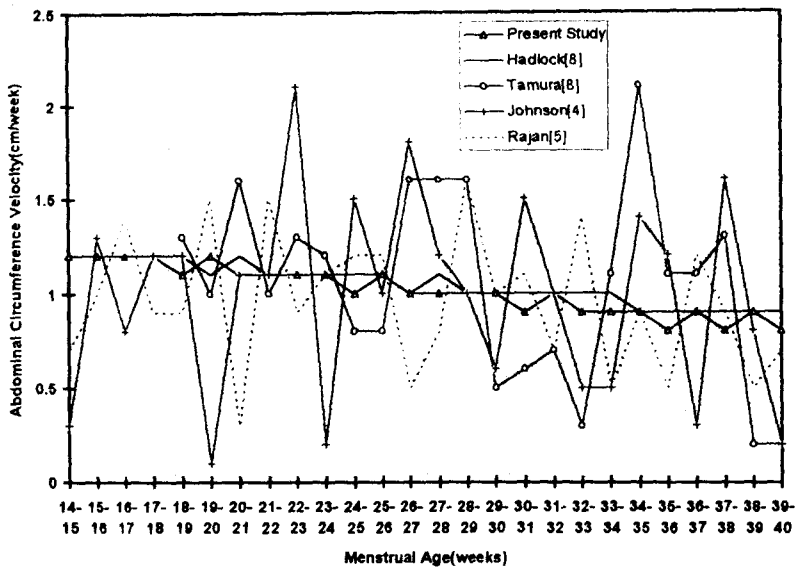


Fig. 5.15. Comparison of Abdominal Circumference Velocity values with Indian and Western studies.

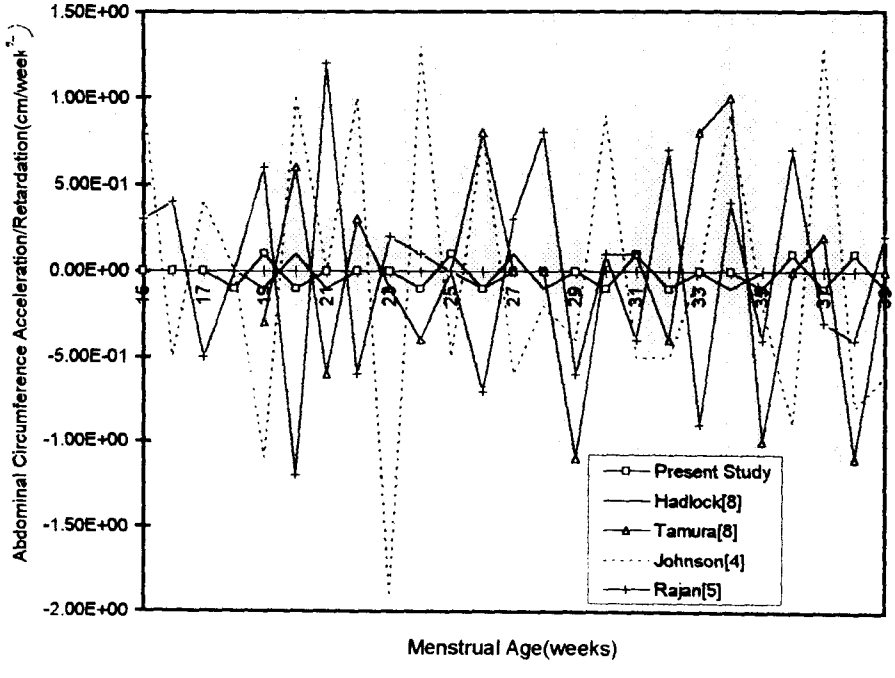


Fig. 5.16. Comparison of Abdominal Circumference Acceleration/Retardation values with Indian and Western Studies.

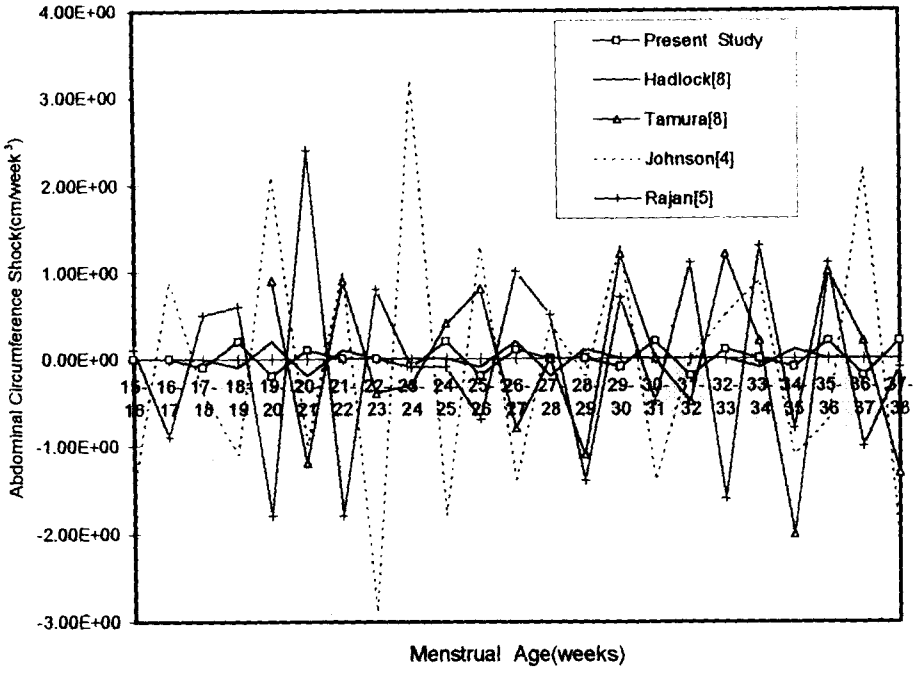


Fig. 5.17. Comparison of Abdominal Circumference Shock values with Indian and Western Studies.

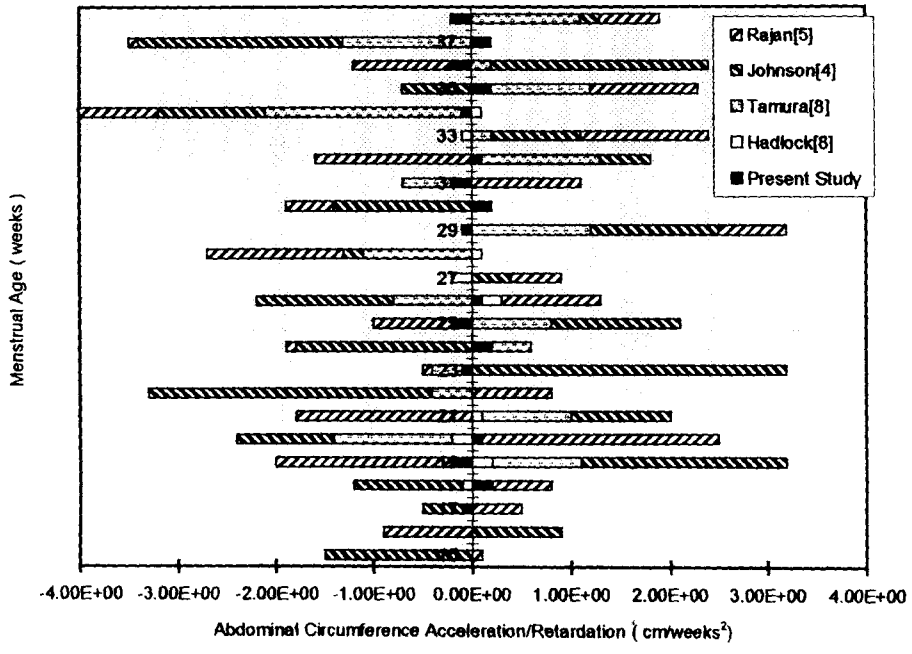


Fig. 5.18. Comparison of Abdominal Circumference Trend values with Indian and Western Studies.

Abdominal circumference shows the marked difference with Western studies which is even evident with the early weeks of gestational age. At the end of the last tri-mester, this deviation can give error upto 3-4 weeks. However, this is similar to Rajan et. al.

5.4.5. **Head Circumference**

The distribution of the mean head circumference data (± 2 SD) as a function of menstrual age is indicated in fig. 5.19. The predicted head circumference values for specific points in gestation (equation 4) are compared with the data from other investigators in fig. 5.20. Comparisons of head circumference velocity values, acceleration/retardation values, shock values, and trend values are presented in figs. 5.21, 5.22, 5.23 and 5.24 respectively.

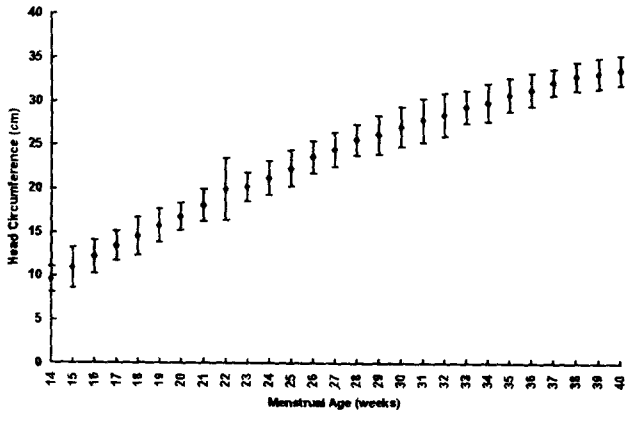


Fig. 5.19. Mean (± 2SD) Head Circumference values observed at specific weeks in gestation.

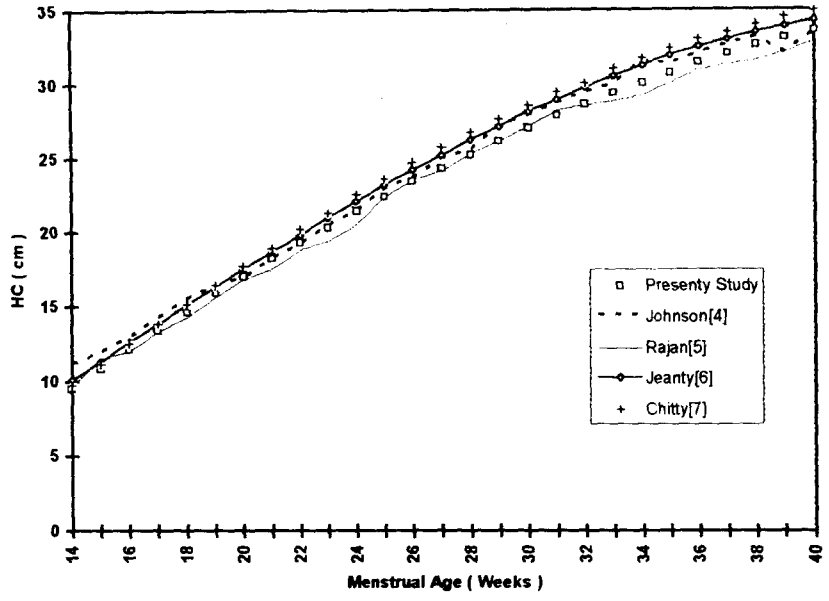


Fig. 5.20. Comparison of Head Circumference values with Indian and Western Studies.

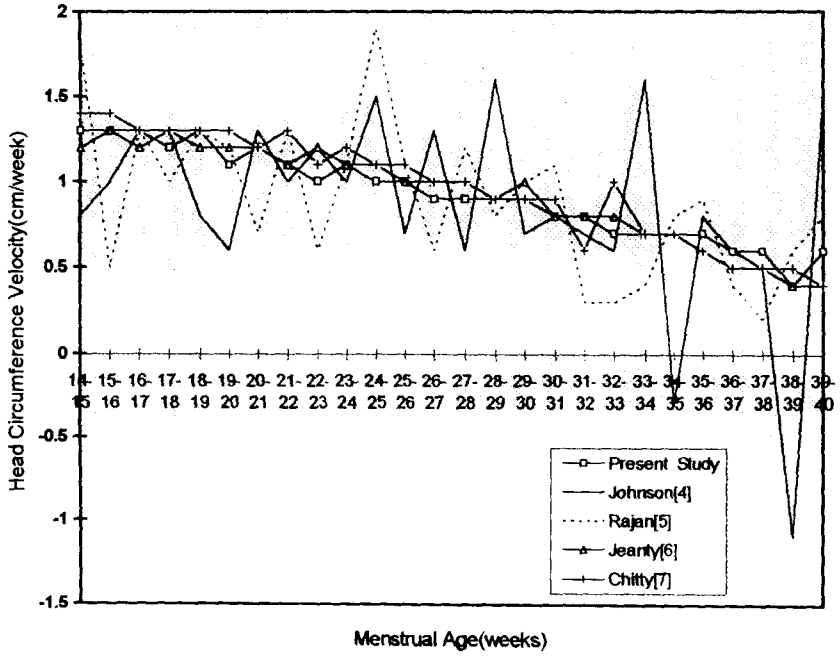


Fig. 5.21. Comparison of Head Circumference Velocity values with Indian and Western Studies.

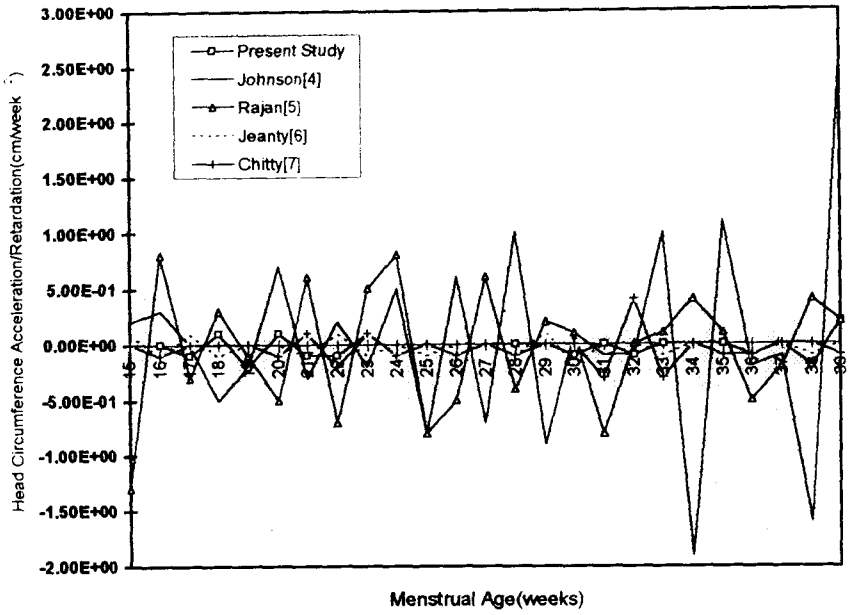


Fig. 5.22. Comparison of Head Circumference Acceleration / Retardation values with Indian and Western Studies.

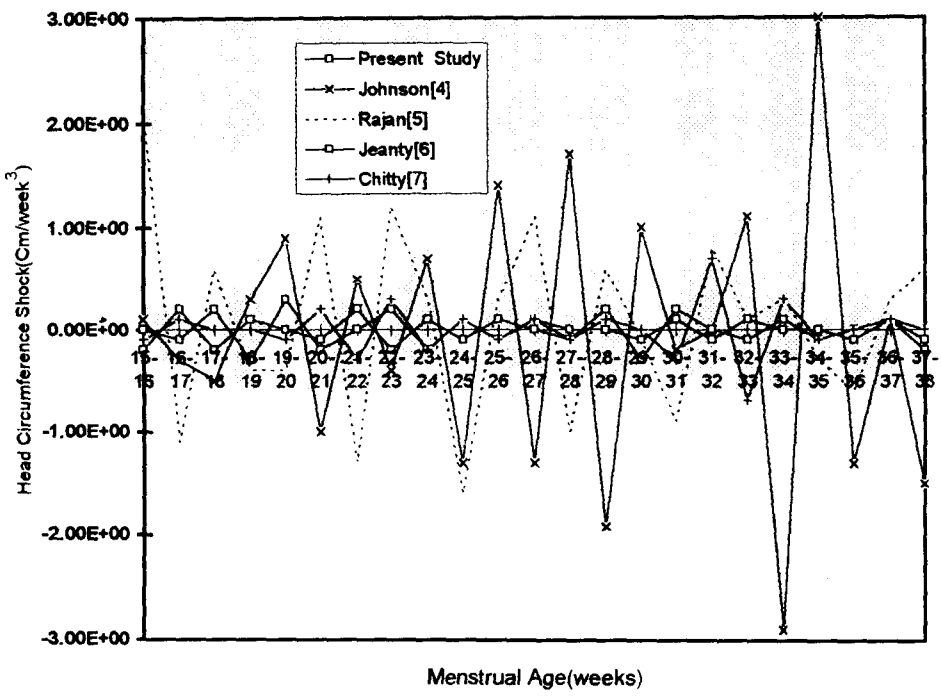


Fig. 5.23. Comparison of Head Circumference Shock values with Indian and Western Studies.

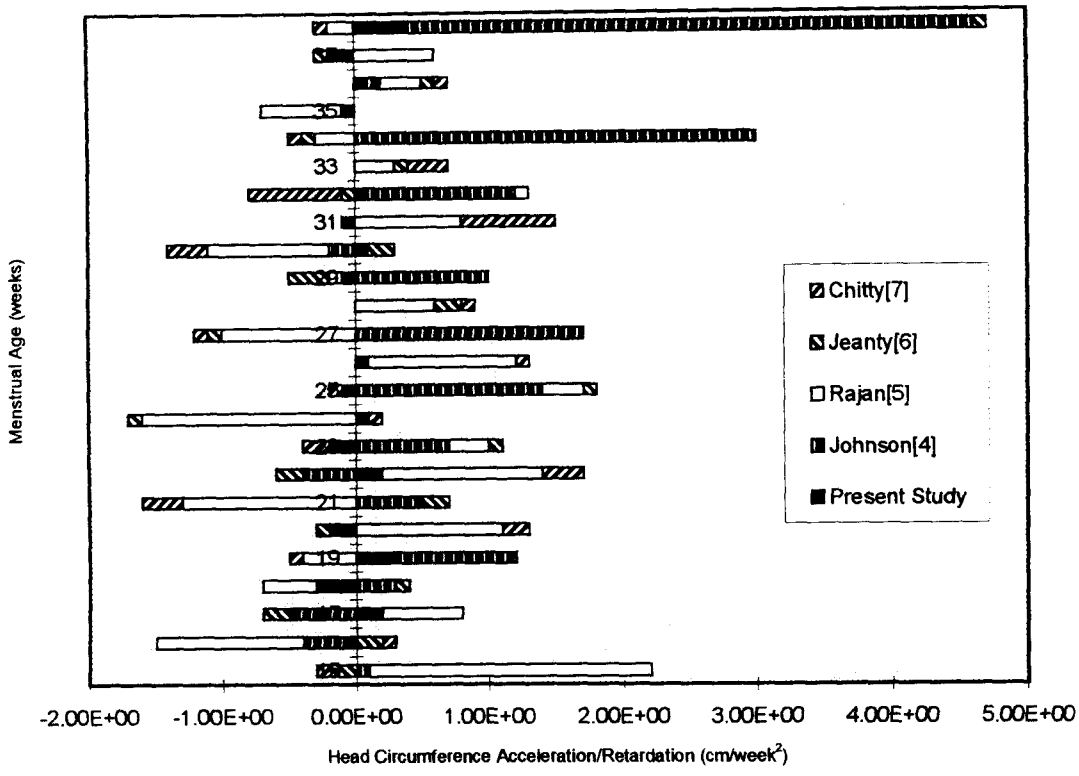


Fig. 5.24. Comparison of Head Circumference Trend values with Indian and Western Studies.

A comparison of the present study shows that there is some difference specially at the later weeks of gestation. However, compared with Rajan et. al. it gives higher value after 32 weeks of gestation. The rate of change of head circumference per week shows the similar pattern i. e. maximum growth of gestation with gradual faultering after the end of last tri-mester.

5.4.6. **Prediction of Gestational Age**

Predicted gestational age values (equation 6) for specific femur length measurements along with a comparison with Hadlock et al. [9] are shown in table 5.1 and in fig. 5.25. Predicted gestational age values (equation 5) for specific biparietal diameter measurements along with a comparison with Hadlock et al.[10] are shown in table 5.2 and in fig. 5.26. Predicted gestational age values (equation 7) for specific abdominal circumference measurements along with a comparison with Hadlock et al. [10] are shown in table 5.3 and in fig. 5.27. Predicted gestational age values (equation 8) for specific head circumference measurements along with a comparison with Hadlock et al. [10] are shown in table 5.4 and in fig. 5.28.

Table 5.1. Predicted menstrual ages for Femur Lengths values

FL(cm)	Menstrual Age (weeks) Hadlock	Menstrual Age (weeks) This study	FL (cm)	Menstrual Age (weeks) Hadlock	Menstrual Age (weeks)This study
1.0	12.8	12.9	4.5	24.5	24.9
1.1	13.1	13.2	4.6	24.9	25.3
1.2	13.4	13.5	4.7	25.3	25.7
1.3	13.6	13.8	4.8	25.7	26.1
1.4	13.9	14.1	4.9	26.1	26.5
1.5	14.2	14.4	5.0	26.5	27.0
1.6	14.5	14.8	5.1	27.0	27.4
1.7	14.8	15.1	5.2	27.4	27.8
1.8	15.1	15.4	5.3	27.8	28.3
1.9	15.4	15.7	5.4	28.2	28.7
2.0	15.7	16.0	5.5	28.7	29.2
2.1	16.0	16.4	5.6	29.1	29.7
2.2	16.3	16.7	5.7	29.6	30.2
2.3	16.6	17.0	5.8	30.0	30.7
2.4	16.9	17.3	5.9	30.5	31.2
2.5	17.2	17.7	6.0	30.9	31.7
2.6	17.6	18.0	6.1	31.4	32.2
2.7	17.9	18.3	6.2	31.9	32.7
2.8	18.2	18.7	6.3	32.3	33.2
2.9	18.6	19.0	6.4	32.8	33.8
3.0	18.9	19.3	6.5	33.3	34.3
3.1	19.2	19.7	6.6	33.8	34.9
3.2	19.6	20.0	6.7	34.2	35.4
3.3	19.9	20.4	6.8	34.7	36.0
3.4	20.3	20.7	6.9	35.2	36.6
3.5	20.7	21.1	7.0	35.7	37.2
3.6	21.0	21.4	7.1	36.2	37.8
3.7	21.4	21.8	7.2	36.7	38.4
3.8	21.8	22.2	7.3	37.2	39.1
3.9	22.1	22.5	7.4	37.7	39.7
4.0	22.5	22.9	7.5	38.3	40.3
4.1	22.9	23.3	7.6	38.8	41.0
4.2	23.3	23.7	7.7	39.3	41.7
4.3	23.7	24.1	7.8	39.8	42.4
4.4	24.1	24.5	7.9	40.4	43.1
			8.0		43.8

Table 5.2.Predicted menstrual ages for BPD values

BPD (cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks) This study	BPD (cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks) This study
2.0	12.2	11.4	6.0	24.6	24.2
2.1	12.5	11.8	6.1	25.0	24.5
2.2	12.8	12.2	6.2	25.3	24.9
2.3	13.1	12.5	6.3	25.7	25.2
2.4	13.3	12.9	6.4	26.1	25.6
2.5	13.6	13.2	6.5	26.4	25.9
2.6	13.9	13.6	6.6	26.8	26.3
2.7	14.2	13.9	6.7	27.2	26.7
2.8	14.5	14.3	6.8	27.6	27.0
2.9	14.7	14.6	6.9	28.0	27.4
3.0	15.0	14.9	7.0	28.3	27.8
3.1	15.3	15.2	7.1	28.7	28.2
3.2	15.6	15.6	7.2	29.1	28.6
3.3	15.9	15.9	7.3	29.5	29.0
3.4	16.2	16.2	7.4	29.9	29.5
3.5	16.5	16.5	7.5	30.4	29.9
3.6	16.8	16.8	7.6	30.8	30.4
3.7	17.1	17.1	7.7	31.2	30.8
3.8	17.4	17.4	7.8	31.6	31.3
3.9	17.7	17.7	7.9	32.0	31.7
4.0	18.0	18.0	8.0	32.5	32.2
4.1	18.3	18.3	8.1	32.9	32.7
4.2	18.6	18.6	8.2	33.3	33.2
4.3	18.9	18.9	8.3	33.8	33.7
4.4	19.2	19.2	8.4	34.2	34.3
4.5	19.5	19.5	8.5	34.7	34.8
4.6	19.9	19.8	8.6	35.1	35.4
4.7	20.2	20.1	8.7	35.6	35.9
4.8	20.5	20.4	8.8	36.1	36.5
4.9	20.8	20.7	8.9	36.5	37.1
5.0	21.2	21.0	9.0	37.0	37.7
5.1	21.5	21.3	9.1	37.5	38.3
5.2	21.8	21.6	9.2	38.0	39.0
5.3	22.2	21.9	9.3	38.5	39.6
5.4	22.5	22.2	9.4	38.9	40.3
5.5	22.8	22.5	9.5	39.4	41.0
5.6	23.2	22.9	9.6	39.9	41.6
5.7	23.5	23.2	9.7	40.5	42.4
5.8	23.9	23.5	9.8	41.0	43.1
5.9	24.2	23.8	9.9	41.5	43.8
			10.0	42.0	44.6

Table 5.3. Predicted menstrual ages for Abdominal Circumference values

AC(cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks) This study	AC (cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks)This study
10.0	15.6	16.1	23.5	27.7	28.2
10.5	16.1	16.6	24.0	28.2	28.7
11.0	16.5	17.0	24.5	28.7	29.2
11.5	16.9	17.4	25.0	29.2	29.7
12.0	17.3	17.8	25.5	29.7	30.2
12.5	17.8	18.3	26.0	30.1	30.7
13.0	18.2	18.7	26.5	30.6	31.2
13.5	18.6	19.1	27.0	31.1	31.7
14.0	19.1	19.6	27.5	31.6	32.2
14.5	19.5	20.0	28.0	32.1	32.8
15.0	20.0	20.4	28.5	32.6	33.3
15.5	20.4	20.9	29.0	33.1	33.8
16.0	20.8	21.3	29.5	33.6	34.4
16.5	21.3	21.7	30.0	34.1	34.9
17.0	21.7	22.2	30.5	34.6	35.5
17.5	22.2	22.6	31.0	35.1	36.1
18.0	22.6	23.1	31.5	35.6	36.6
18.5	23.1	23.5	32.0	36.1	37.2
19.0	23.6	24.0	32.5	36.6	37.8
19.5	24.0	24.4	33.0	37.1	38.4
20.0	24.5	24.9	33.5	37.6	39.0
20.5	24.9	25.4	34.0	38.1	39.6
21.0	25.4	25.8	34.5	38.7	40.2
21.5	25.9	26.3	35.0	39.2	40.8
22.0	26.3	26.8	35.5	39.7	41.5
22.5	26.8	27.2	36.0	40.2	42.1
23.0	27.3	27.7	36.5	40.8	42.7
			37.0		43.4

Table 5.4. Predicted menstrual ages for Head Circumference values

HC(cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks) This study	HC (cm)	Menstrual Age (wks) Hadlock	Menstrual Age (wks)This study
8.0	13.4	12.5	22.5	24.4	25.3
8.5	13.7	13.0	23.0	24.9	25.8
9.0	14.0	13.4	23.5	25.4	26.4
9.5	14.3	13.9	24.0	25.9	26.9
10.0	14.6	14.3	24.5	26.4	27.5
10.5	15.0	14.7	25.0	26.9	28.1
11.0	15.3	15.1	25.5	27.5	28.7
11.5	15.6	15.5	26.0	28.0	29.3
12.0	15.9	16.0	26.5	28.1	29.9
12.5	16.3	16.4	27.0	29.2	30.5
13.0	16.6	16.8	27.5	29.8	31.2
13.5	17.0	17.2	28.0	30.3	31.9
14.0	17.3	17.6	28.5	31.0	32.5
14.5	17.7	18.0	29.0	31.6	33.3
15.0	18.1	18.5	29.5	32.2	34.0
15.5	18.4	18.9	30.0	32.8	34.7
16.0	18.8	19.3	30.5	33.5	35.5
16.5	19.2	19.7	31.0	34.2	36.3
17.0	19.6	20.1	31.5	34.9	37.1
17.5	20.0	20.6	32.0	35.5	37.9
18.0	20.4	21.0	32.5	36.3	38.8
18.5	20.8	21.5	33.0	37.0	39.7
19.0	21.2	21.9	33.5	37.7	40.6
19.5	21.6	22.4	34.0	38.5	41.5
20.0	22.1	22.8	34.5	39.2	42.4
20.5	22.5	23.3	35.0	40.0	43.4
21.0	23.0	23.8	35.5	40.8	44.4
21.5	23.4	24.3	36.0	41.6	45.5
22.0	23.9	24.8	36.5		46.5
			37.0		47.6

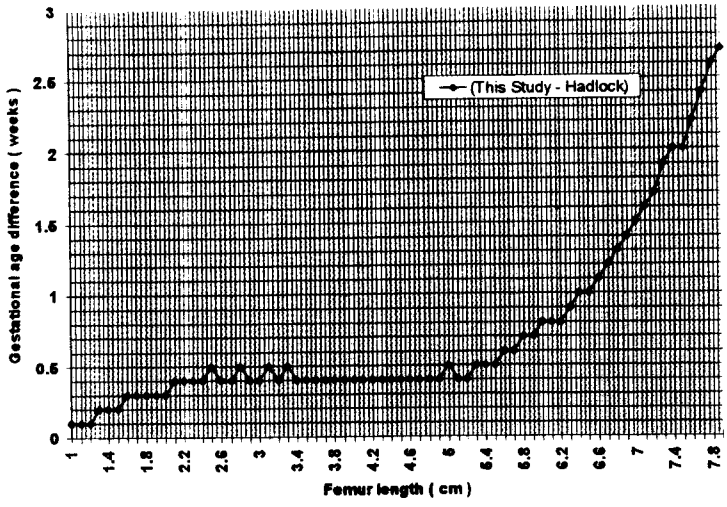


Fig. 5.25. Week difference of calculated gestational age from femur length.

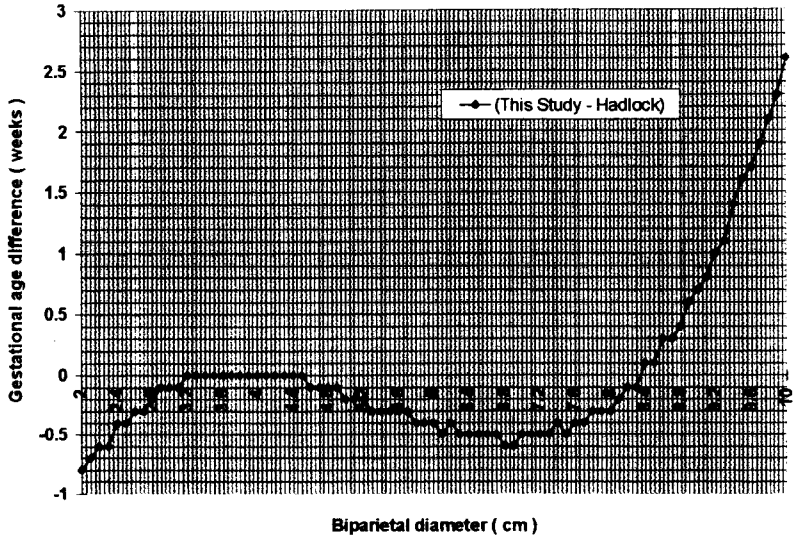


Fig. 5.26. Week difference of calculated gestational age from biparietal diameter.

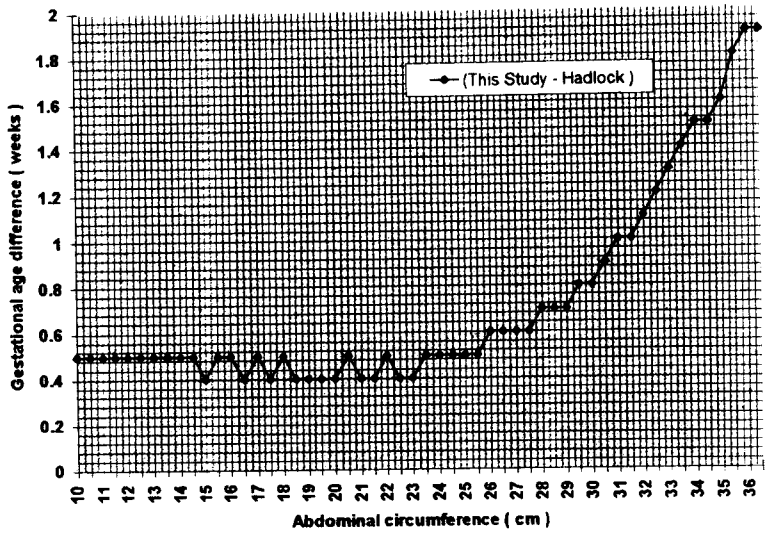


Fig. 5.27. Week difference of calculated gestational age from Abdominal circumference.

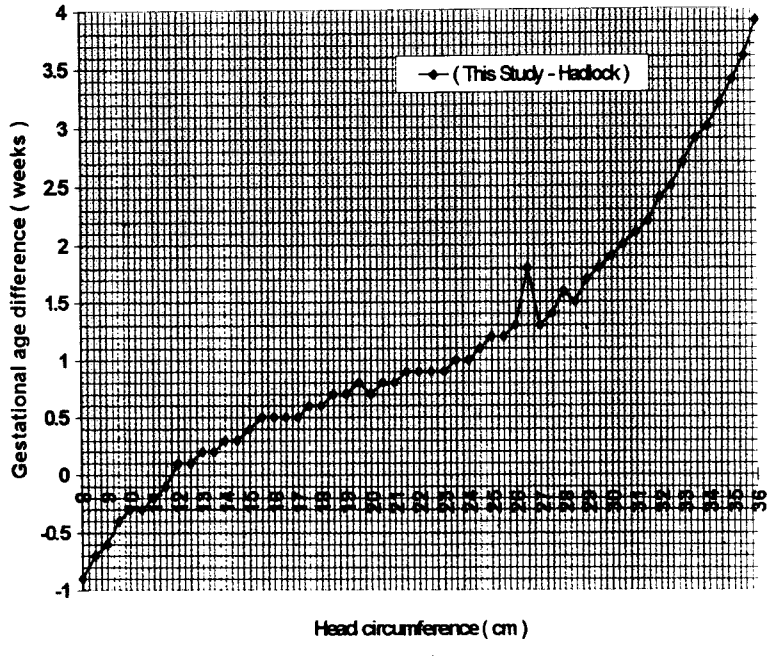


Fig. 5.28. Week difference of calculated gestational age from Head circumference.

5.4.7. **Conclusions**

It is evident from the curves and the above tables that there is a difference of 1-3 weeks gestational age derived from femur length, 2-3 weeks derived from abdominal circumference, 1-2 weeks derived from biparietal diameter, and 2-4 weeks derived from head circumference in 3rd trimester. An overall calculation gives a difference of about 2 weeks of menstrual age in 3rd trimester. So, it would be pertinent to consider zonal effects during the development of such expertise.

References

1. J. P. Habicht, R. Martorell, C. Yarbrough, R. M. Malina, R. E. Klein. Height and Weight Standards for Pre-School Children. How relevant are Ethnic differences in Growth Potential ? Lancet ; 1974, I, 611- 615.

2. P. B. Eveleth, J. M. Tanner. World Wide Variations in Human Growth. Cambridge University Press; 1976.

3. E. M. Widdowson. How the fetus is fed. Proc. Nutr. Soc.; 28, 1969, 17-24.

4. E. A. Reece, I. Goldstein and J. C. Hobbins. Fundamentals of Obstetrics & Gynecologic Ultrasound. PHI Inc.; 1994, 141- 181.

5. P. Kumar, M.Y. Rawal, S. Dasgupta, R. Rajan, N. Malhotra. Imaging in Obstetrics & Gynecologogy. Federation of Obstetric & Gynecological Societies of India, Jaypee,2nd. Ed.; 1998, 298 - 299.

6. M. Hansmann, B. -J. Hackelöer, A.Staudach. Ultrasound Diagnosis in Obstetrics & Gynecology. Narosa Pub. House; 1990, 176 - 177.

7. K. Dewbury, H. Meire, D. Cosgrove (Eds.). Ultrasound in Obstetrics & Gynecology. Churchill Livingstone; 1993, 526 - 536.

8. F. P. Hadlock, R. L. Deter, R. B. Harrist, S. K. Park. Fetal Abdominal Circumference as a Predictor of Menstrual Age. AJR; vol. 139, Aug. 1982, 367 - 370.

9. F. P. Hadlock, R. B. Harrist, R. L. Deter, S. K. Park. Fetal Femur Length as a Predictor of Menstrual Age : Sonographically Measured. AJR; no. 136, May 1982, 875 - 878.

10. P. A. Athey, F. P. Hadlock. Ultrasound in Obstetrics & Gynecology. B. I. Publications Pvt. Ltd., 2nd. ed.; 1992, 22-37.

SELECTING AN APPROPRIATE EXPERT SYSTEM TOOL[+]

6.1. Introduction

For a successful development of an expert system, a number of factors are responsible. Notably, selection of an appropriate expert system problem domain, knowledge acquisition and formal representation, system construction, validation and testing are typically important. Not only the degree of complexity of these tasks are fairly high but also it is fairly easy to mismanage. Different expert system tools have been developed to improve the situation. More specifically, these tools are used to less the burden on a developer and in a more cost-effective way. Although, there are tools to assist in different phases of such development, we are confined here to the matter of selection of tools during actual system construction process after the selection of a problem domain, and the knowledge acquisition process.

In the next section, we discuss some potential inconveniences in selecting a tool for implementation during an expert system development. In section 6.3, we have analysed the features and capabilities of two potential tools at our disposal : (i) Level5 (object), and (ii) Turbo Prolog. Lastly, our conclusions and discussion are summarised.

6.2. Potential inconveniences

This section presents some potential inconveniences faced by an expert system designer and discusses the classification of tools. This section also intends to highlight the inconveniences particularly faced by the author in the selection process in connection with the present work.

6.2.1. No general purpose tool

The headache of selecting an appropriate tool would have been relieved if we have been provided a general purpose tool for our use. Unfortunately, it is not possible to construct such a general purpose tool simply because human beings commonly utilise a knowledge based approach rather than a general purpose approach for problem solving [1]. As a result, a range of tools are provided applicable in a wide

---

[+] This is based on the publication [CSI Communications, vol. 20, no.7, January 1997; 24-32; ibid vol. 20, no. 8, February 1997, 19-22] of the author.

variety of domains. For a successful development of an expert system, it is essential that an appropriate tool is being selected.

6.2.2. **Single or multiple tools**

In a particular situation, a single tool may not be adequate to fulfil the requirements the problem domain lays on an expert system. Due to the evolutionary nature of an expert system development, one may find it worthwhile to use one tool for prototype system and the other for target system. Obviously, this adds one confusion - which type(s) of tools would be suitable for prototype development and which would be suitable for target system development. It is rather difficult to offer sound guidelines here. Two typical approaches may be useful : (i) solving the chicken-egg problem [2], and (ii) use one of the large, hybrid object-oriented toolkits [3] throughout the total developmental phases. In the first approach, it is assumed that few expert systems development efforts are well formulated in advance. The problem domain is complex, domain knowledge and expertise are ambiguous and the nature of the problem is such that complete unfolding is only possible after considerable exploration and experimentation. This is where one has to address the chicken-egg problem. One can choose an appropriate tool only after understanding the total requirements the problem domain lays on an expert system; but this may only be possible after experimentation on building a prototype which requires choosing a tool. This is where two types of tools may be considered useful.

During demonstration / research prototypes, one should use a tool having fast prototyping property e.g. Prolog [4]. The second approach assumes that selection of such appropriate implementation tool should only be done after complete requirement analysis of the problem domain lays on an expert system. In most of the situations this may prevail. Even though the total requirement analysis is not complete, it might be advantageous to use a comprehensive tool assuming that the simpler rule-based production systems do not qualify; model-based reasoning would be more appropriate. Potential difficulties in using multiple tools (discussed below) can be reduced here substantially. The objectives of demonstration / research prototypes can be fulfilled using the simple structures of the tool. More complex structures of the tool can be used for production system. Use of multiple tools may lead to the following problems : (i) Financial investment; may not be practical in many cases, (ii) Man-years investment; before using any such tool, one has to be conversant with the intricacies of the tool. The amount of man-years would be increased simply by a factor of two or three depending on the number of tools used. Moreover, once domain experts and knowledge engineers have been trained on a particular system, it is not simply feasible to be retrained on another potentially suitable system; (iii) Interfacing problem; usually the architectural design and functionality of each tool is different, one has to interface such two or more tools. This would require an extra

degree of expertise involving a fair amount of man-years. The original objectives of using tools e.g. less time, less effort are thereby challenged.

As a worst situation, no such existing higher level tools might be adequate to satisfy the requirements of the problem domain. This is where one has to design his / her own architecture. Here, one has to fix up his / her mind whether the goal of the system development is to develop a system for actual use or to make major advances in the state-of-the-art of expert system technology. As observed by Prerau [5], it is not wise to attempt to achieve both of these goals simultaneously simply because it is really a formidable task. So, a system development for actual use may demand pruning of certain characteristics of the problem domain within the confines of current expert system technology.

6.2.3. **In search of a bird after constructing a cage**

With the advent of increasing number of higher level tools (e.g. shells or toolkits) one can observe an interesting phenomenon [6]. In most cases, peoples are in search of birds after the construction of a cage. That is, the members of a project team are forced to in search of a problem domain which suits the solution techniques of the tool already chosen. This approach, obviously, restricts the natural flow of selection of a problem domain of our society. Therefore, the reverse approach, i.e. selecting an expert system problem domain and then go for selecting an appropriate tool for implementation, is much more natural. Obviously, this may lead to a situation of having no higher level tool suitable for the problem domain.

6.2.4. **Exaggerated claims from vendors / agents**

It is rather very difficult for the end users to distinguish between facts and hyperbole. Vendor literature, demonstrations and reference manuals are subject to exaggerated claims [7]. It may not always be possible to go through the detailed experimental verification of these claims before the actual procurement. Even, sometimes, agents do suppress the potential demerits of the tools for selling his / her product.

6.2.5. **Non-standard terminologies**

Standard terminologies with their standard definitions and actions are really useful for a better comparison of tools and thereby ease the process of selection of such tools. Unfortunately, some tools do not agree in terminologies. For example, in KEE, frames are called units, properties of units are called slots, and properties of slots are called facets. But, in S1, frames are called classes, properties of classes are called attributes, and properties of attributes are called slots. Similarly, the term rule used in ROSIE, ART and RULE MASTER are different in performing actions. So, the

terminologies with non-standard definitions and actions add an extra degree of difficulty, the prospective users face when selecting tools.

6.2.6. **Miscellaneous issues : Price, training and documentation support**

"High price, good quality - more facility" is, however, considered true. But, from functionality and performance point of view, price is not necessarily an indicator of suitability. A tool costing less may be more suitable at per the requirements of the problem domain at hand than a high cost tool. For the ease of use and for the quick exploitation of the potential features of the tool, a comprehensive training should be considered mandatory. The trainer should be an expert preferably a core member of the group responsible for the development of the tool. In different developed countries, the source of most advanced tools, this can easily be observed. Most of the vendors have their online facilities to support the customers; this facility can easily be enjoyed by the customers having advanced communication tools. For us, in India, the situation is not so favourable. Usually, the vendors sell their products through agents; the agents are not well-equipped with the properly trained experts. Even, the agents are, sometimes, reluctant to offer an on-site demonstration. Frequent on-line assistance is again a costly affair; may not be feasible for every customer. The situation may be improved with good documentation including a user's guide, a reference manual and an architecture manual and some demonstration examples with clarification.

6.2.7. **Language, shell or toolkit**

The selection of an appropriate tool should also be dictated by the relative merits and demerits of these three classes of tools. Generally speaking, the shells provide the upper level of a stratum of tools, the lower level is being provided by the languages and the middle level is being provided by the toolkits. The metrics like applicability, abstraction, facilities and costs of hardware, software and training may be considered useful in the comparison process [1] as discussed below.

6.2.7.1. **Applicability**

Languages are applicable quite generally and virtually any type of expert system can be designed. On the other hand, shells are rather specific in this context. The good matching of the requirements the domain lays on an expert system and the facilities the shell offers is the key to success of the development of an expert system. The toolkits should have the generality of the language approach but also contain specific representations and control strategies.

6.2.7.2. **Abstraction**

The level of abstraction is low in the language approach and medium in the shell approach. On the other hand, toolkits provide a rich set of abstraction.

6.2.7.3. **Facilities**

In the toolkit, the facilities are most rich. Shells offer medium facilities. In a language approach, we get limited facilities but, however, any facility missing in a language may be provided by programming.

6.2.7.4. **Costs**

(i) **Hardware**

In the case of languages and shells, the costs of hardwares are generally quite low compared to toolkits. This is simply because toolkits often demand specialised hardwares whereas languages and shells run on PC or on workstations. Although, recent versions of some toolkits run on PC but the memory / backup size is reasonably high.

(ii) **Software**

The costs of languages and shells are more or less same. But, toolkits are normally more costly than other two.

(iii) **Training**

In general, the shells require a fairly less efforts and short time for learning. But, however, languages require a more extensive efforts and training period.

With due consideration of the observations, we summarize that the toolkit approach appears to be superior to other two approaches on the consideration of applicability, abstraction and facilities offered. But, however, it appears to be inferior to language and shell approaches on the consideration of costs of hardware, software and training. Moreover, although, a toolkit offers a good number of facilities, these may not fulfil all the requirements of the problem domain lays on an expert system. The programming facility, if any, of a toolkit is expected to fulfil such requirements. Once again, one has to be a master of a language like LISP or PROLOG which is provided by a toolkit. One may now have an idea on the sales of expert system software tools per year [8] as depicted in fig. 6.1. From fig. 6.2, one can have an idea how expert system applications have developed : (a) on different platforms, and (b) with different softwares.

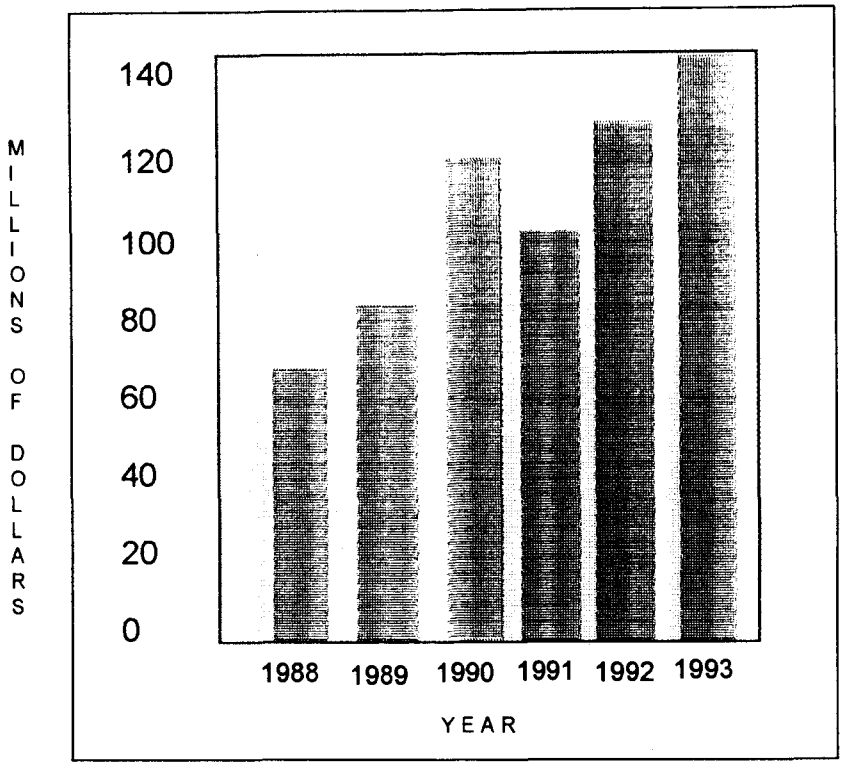


Fig.6.1 Sales of expert system software tools per year

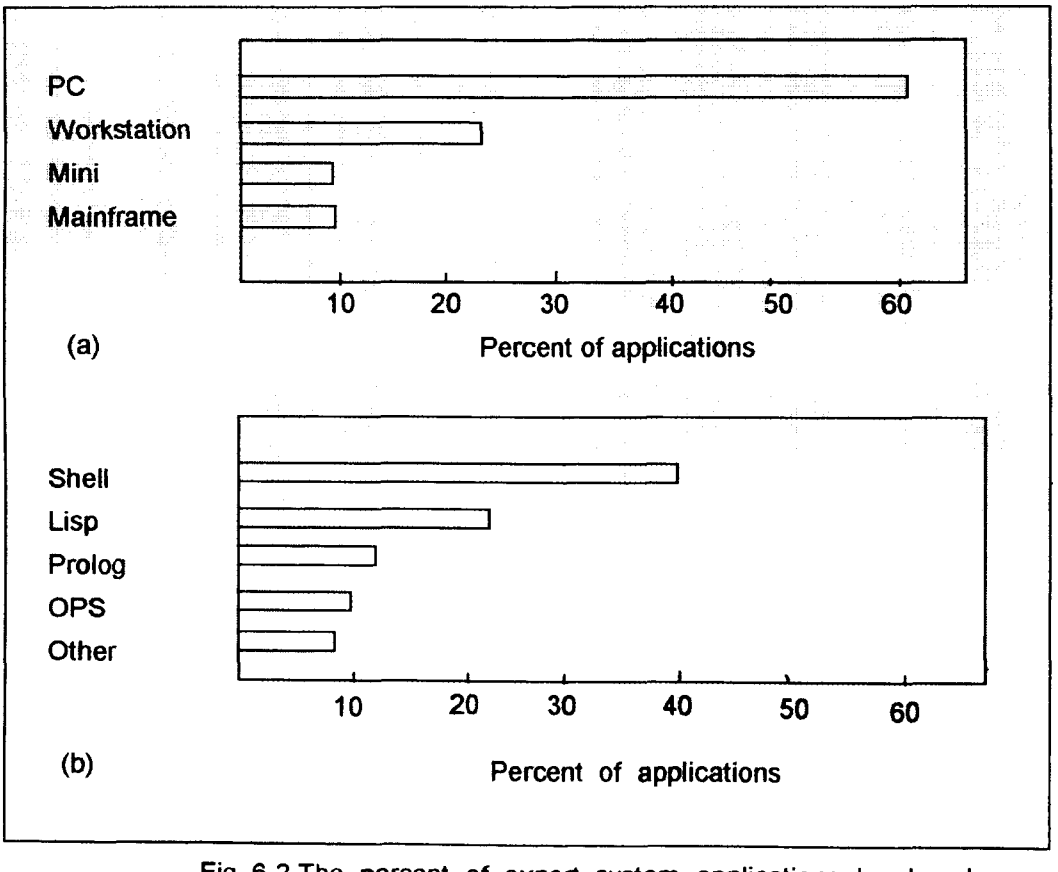


Fig. 6.2 The percent of expert system applications developed (a) on different platforms; (b) with different softwares

6.2.8. **Left no stone unturned - Is it practically feasible ?**

A decade back literature [9] would be adequate to explain the above head line. Bundy [9] provides a catalog of over 250 software products and AI techniques. Hopefully, this number should be over 450 at the end of 1999. In such a situation, it may be pertinent to ask whether it is practically feasible or not for an end user to turn each and every stone. This should not be an impossible task but this may lead to an unacceptable delay in achieving the ultimate objective of selecting such a tool. Obviously, this demands a fast pruning mechanism.

6.2.9. **Potentially active research field**

It may not be possible to select the best one but a better one due to the evolutionary nature of this potentially active research field. One may find a better tool tomorrow satisfying more need of the problem domain. But, to develop a system for practical use one has to be confined to the present ES-technology where all the requirements may not be satisfied.

6.2.10. **Any unique framework ?**

Because of the so many turbulant features just creating the unstability in the selection process, the tool evaluation and selection can not entirely be mechanical. Here, human expertise and judgement will certainly play a significant role especially for the pruning process. But, obviously, this might lead to different solutions of the same problem which again demands a more formal mechanism. To what extent this formalisation would be achieved ? Answering this question and suggesting a formal method are really formadible tasks. It is, rather little bit easier to suggest a general framework for the problem. Rothenberg [2] suggests a framework with eight fold steps emphasising "matching a tool to its intended use" rather than simply "matching a tool to a problem". This framework might be worthwhile in many cases but it involves a larger number of criteria, may not lead to a manageable situation.

For example, 'contexts' dimension might demand five different tools suitable for five different contexts such as conceptualization, prototyping, development, fielding and operation / maintenance; leading to an unmanageable situation. But, however, pruning and prioritizing should make the situation manageable which again may depend on individual's experience and judgement. This might suggest another more simple framework. However, the matter is yet to be settled. But, it may be of general acceptance that such an evaluation and selection should primarily depend on the requirements the domain lays on an expert system and on the tool capabilities rather than tool features. This demands a thorough analysis of the problem domain, the

problem itself and the anticipated project including, even, the potential users of the proposed system.

6.3. ES-building tools' capabilities

It might be more important to focus on the capabilities of a tool, rather than the specific features the tool provides for achieving or supporting those capabilities. Highlighting capabilities means highlighting the functionalities of a tool rather the specific implementation of a functionality. Users are generally interested in different capabilities (and it is also more convenient for less experienced end users) without knowing the technical features supporting those capabilities. Some representative potential capabilities and the corresponding supporting features are identified here as presented in table 6.1 which includes the suggestions of Rothenberg [2].

Table 6.1. Capabilities of tools with supporting features.

Capability	Examples of Supporting Features
Arithmetic processing	Arithmetic operators, extended floating point
Built-in-functions	Mathematical, statistical, string, type conversion
Certainty handling	Certainty factors, fuzzy logic
Concurrency	Distributed processing, parallel processing
Consistency checking	Knowledge base syntax checking
Data type handling	Numeric, string, time, simple, compound
Documenting development	Assumption/rationale history, code / data annotation
Explanation	Execution trace, knowledge base browsing
Inference & control	Iteration, forward / backward chaining, inheritance
Integration	Calling other languages, interprocess calls
Internal access	Tool parameter setting functions, source code
Knowledge acquisition	Rule induction, model building aids
Knowledge-base editing	Structure editors, graphic rule lattice
Life Cycle	Tool support for target system life cycle support
Menus	Goals, Reasoning, tools
Meta-knowledge	Rules controlling inference, self-organizing data
Optimization	Intelligent look-ahead, caching, rule compilation
Presentation (I/O)	Text, graphics, Windows, forms, mouse, keyboard
Representation	Rules, frames, procedures, objects, simulation

Let us now consider two potential tools at our disposal : (i) a large-hybrid-object-oriented toolkit - Level5 Object, and (ii) Turbo-prolog for further analysis in the light of the above discussed tool capabilities. We shall be trying to draw a comparison of these two tools according to their capabilities at per the above identified representative list. This discussion is based on Level5 object (Rel 3.0) for Micro-soft Windows and Turbo-prolog (Rel 1.1).

6.3.1. Level5 Object

6.3.1.1. What is Level5 Object ?

Level5 object is a software development tool kit from Information Builders, USA. It is the application development tool to combine client / server technology, object-oriented programming, graphical user interfaces, and knowledge based systems [10].

Level5 object is a software development tool kit. Even if one has little programming experience, he / she can use it to create complex applications in an easy, consistent, and maintainable fashion.

Level5 object is an environment. It contains all the tools necessary to solve a very wide range of problems. The Level5 object toolbox contains an integrated set of editors that help one rapidly create software solutions. From rapid prototypes to large, mission-critical applications, Level5 object is a proven winner as demanded by its developer.

Level5 object is a development tool and a delivery vehicle. Once an application has been created using the Development System, it can be delivered to end-users with the Run-Only System. The Run-Only System provides a variety of delivery choices emphasizing smaller. faster systems. These systems can be encrypted to provide enhanced security.

Level5 object uses a high-level language called PRL. PRL is designed to be simple to learn and read, and is similar to natural English. Although we rarely see it because of the interactive editors, it is there to provide maximum flexibility and accuracy when we develop an application. All of the elements of the software we wish to create can be expressed in the PRL language. The text of the application can be sent to other hardware platforms and operating systems, compiled, and executed. We can therefore create our software solutions where it is most convenient and inexpensively deliver them to other hardware platforms.

Level5 object is well-connected. It contains built-in access to over 60 local and remote data-bases and servers, access to all the common local database formats and SQL servers, interfaces to external programs, communication paradigms, text files, timers, and custom interfacing options.

Level5 object is an expert system. One can use it to create "smart" systems. It can solve real world problems. Using simple rules that can reason or pattern matching triggers that can react to situations, Level5 object can provide consistent, educated answers to the people who need it.

Level5 object contains the following integrated array of powerful tools :

- True objects providing the efficiency of object-oriented programming.
- Graphical User Interface (GUI) development editors, forms and display builders, and control over all aspects of the user interface.
- Complex logic capabilities, business rules, triggers, agendas, procedural and non-procedural modules.
- Robust and seamless database access, SQL, object-oriented databases, and client / server architectures.
- Complete set of integrated debugging tools, stepping, breakpoints, traces, and reasoning.
- 100% portability to other hardware and operating system platforms.
- Compiled execution for efficient application speed and size.

6.3.1.2. **What kinds of problems are best solved with Level5 object** ?

Level5 object has consistently shown itself to be an effective tool for solving certain classes of application problems. Here is a short list of the kinds of problems one can solve with Level5 object. However, because it is a general purpose tool with a broad range of capabilities, one may not be limited to this list.

Scheduling

These applications solve difficult scheduling problems that exceed the capabilities of conventional off-the-shelf solutions. Level5 object's event-driven architecture allows one to create scheduling applications that adjust "on-the-fly" to the changing events of our day-to-day business activities.

Resource and **Constraint Management**

These applications juggle resources and resource constraints to find a best-fit scenario that can mean the business efficiency difference between one and one's competition. Modern business is moving to "just-in-time" operations in order to operate with the least cost and the highest productivity. Innovative organizations use Level5 object as the critical decision-making link in these resource management applications.

Regulation Compliance and **Conformance**

Worldwide, government organizations are using Level5 object to create "smart" forms systems that enforce consistent compliance with government regulations and ensure higher quality decision making by the work force. By codifying regulations and guidelines into business rules and embedding them into automated business process applications, many organizations have improved the quality and fairness of the services they provide.

Diagnostics

Level5 object has the unique ability to resolve complex diagnostic problems across a broad range of industries. Level5 object has proven itself as the quickest and easiest path to a solution, whether one is developing fault detection systems, real-time process monitoring and control applications, or business evaluation and decision support applications.

Client / Server

Level5 object has built-in interfaces to all the common and important local and remote database servers on mainframes and mid-tier systems. Its robust and flexible support of the user interface environment on the client side, as well as its ease of use and development features, make Level5 object an excellent tool to field client / server applications and cooperative processing.

6.3.1.3. **Capabilities of Level5 object**

Let us now have a look on the capabilities of Level5 object as follows :

Arithmetic Processing

Level5 object provides arithmetic processing to the attributes of numeric type of different objects. The types of operators such as assignment, numeric, relational along with numeric functions and numeric value editor etc. are used for the purpose.

Built-in-functions

It provides on-system functions that operate on numeric, string, time, interval attributes. The typical functions are : Mathematical, Reference, Statistical, String, Time / Interval, Trigonometric. Value type conversion.

Certainty handling

Level5 provides a way to build reasoning with partial or uncertain information into a knowledge base. When confidence prompting is 'on', the user will assign a confidence value. In response to customer requests, confidence in Level5 object can be expressed as a numeric value (-2..100). -2 indicates 'unknown' and -1 indicates 'undetermined'. Confidence values can also be assigned from within the rules of a knowledge base, which allows developers to quantify the degree to which they are confident in the accuracy of a conclusion or the degree of accuracy required to make a conclusion. This version supports product space confidence. With this technique, confidence is calculated by multiplying the confidence value of the antecedent and the value of the conclusion. Confidence management strategies other than product space can be developed with the help of confidence function.

Consistency checking

Level5 object closely monitors all respects of knowledge base, it maintains complete referential integrity across rule, file and object management. Level5 goes beyond monitoring to prevent any attempt by a developer to change or accidentally delete classes, data, attributes, rules, instances or demons referenced elsewhere in a knowledge base.

Data type handling

It describes the type of information represented by an attribute of a class using attribute type. The attribute types are : colour, compound, instance reference, interval, multicompound, numeric, picture, rectangle, simple, string and time. In addition, the attribute types may be of single value or an array of different sizes.

Documenting development

In Level5 object, the process of documentation can be achieved using the display editor. Named rules and objects facilitate the documentation process in the system.

Explanation

The Level5 report system is a comprehensive explanation facility that provides complete access to the inferential reasoning process while running a knowledge base. In the case of the human mind, how a decision is reached is usually as important as the determination itself. Similarly, Level5 maintains full audit trails of how it arrives at its conclusions. Analytical reasoning facilities, such as session monitor, historical traces, single-stepping and breakpoint, enables developers and when necessary, end

users to view all the current states of the inference engine; examine and change the state of any fact in a knowledge base; review the answers provided to Level5 quaries; and follow the line of reasoning being pursued. By activating a debug window while running an application, developers can observe and trace the reasoning process. In single-step mode, Level5 pauses after each event the inference engine processes, allowing the developer to view the action before resuming the session.

Inference and Control

Level5 can process information in a variety of ways : back chaining, parallel inference processing, dynamic agendas, rules, blackboard techniques and object-oriented programming. This versatility gives developers ultimate flexibility over the way in which data is presented and processed. It uses rules, demons, methods of WHEN NEEDED and WHEN CHANGED styles. These methods may contain some repetitive structures like IF....THEN.....ELSE; WHILE; DO... UNTIL and FOR.... TO. In addition, LOOP-statement is also available but not applicable to two those methods of WHEN NEEDED and WHEN CHANGED styles.

The class property INHERITS allows one to transfer the structure and behavior of a parent class to a child class. Consequently, a child class receives all of the capabilities of its parent classes and uses not only the attributes of the parent class, but also its methods, rules and demons. Therefore, one can reuse an application's code down the class hierarchy, increasing productivity and accuracy.

In Level5 object, a class can inherit from more than one parent. This feature, called 'multiple inheritance' allows one to create hierarchical chains of inheritance.

Integration

Direct database access enables Level5 to read and write to file types directly from within the knowledge base. Level5 object supports dBASE III and III PLUS, Lotus 1-2-3 WKS and WK1, SQL and ASCII file formats. Using object-oriented database management techniques, developers build and maintain powerful applications that are able to access large, heterogeneous data structures. By committing much of the underlying complexity of data access to system classes, developers can limit end-user access to only pertinent data and make the process of sourcing attribute values from databases virtually transparent. In addition, its DDE system class provides direct program-to-program communication capabilities. In this release Level5 object functions only as a client.

Internal access

Level5 object provides different commands and facets for different parameter values at the begining of a consultant session or when running an application e.g. RESIZE, INIT, REINIT etc. It also provides the source code of a knowledge base in a language what is known as Production Rule Language (PRL).

Knowledge acquisition

Level5 object does not provide any facility for automatic knowledge acquisition for a system development.

Knowledge-base editing

Level5 object is equipped with five editors such as Objects editor, Database editor, Display editor, Methods / Rules / Demons editor and Windows editor. With these editors one can edit different parts of the total knowledge base.

Life Cycle

It is in the nature of systems that share a common life cycle pattern. After a system has been in operation for a number of years, it gradually decays and becomes less and less affective because of the changing environment to which it has to adapt. For the time being it may be possible to overcome problems by amendments and minor modifications to the system but eventually it will be necessary to acknowledge the need for fundamental changes which demands a new system.

In Level5 object the required minor modifications in knowledge base in terms of objects, attributes, rules, methods, demons etc are possible. It is also possible to delete any object or add new objects as when required. The system automatically takes care of the validity of any deletion or modification in the knowledge base. For a new system development the created objects can be used. This expedites the new system development.

Menus

The menu-driven facilities of Level5 object makes it handy to the system developer as well as to the end users.

Meta-knowledge

Different facets of Level5 object tell the inference engine how to process an attribute. Facets like BREAKPOINT, EXHAUSTIVE, REINIT etc. as well as Rules / Methods / Demons are used to control inference in an application.

Optimization

Separate compilation of source knowledge base is not required in Level5 object. The object code is automatically created after addition of source text. It supports compiled execution for efficient application speed and size. One can expedite processing using the SMARTDrive concept. SMARTDrive is a disk-caching program provided with Windows. In addition, writing the software in C increases performance speed.

Presentation (I / O)

Considering I / O presentation, Level5 object supports text, graphics, windows, forms, mouse and keyboard. Where mouse is not available, it runs fully using a keyboard.

Representation

Level5 object is a fully object-oriented system. The objects are managed using objects editor. It also supports rules (single or group), methods (WHEN NEEDED and WHEN CHANGED) and demons.

6.3.2. **Turbo Prolog**

Prolog is short for programming in logic. The language was originally developed in 1972 by Alain Colmerauer and P. Roussel at the university of Marseilles in France for writing natural language translation systems, and it quickly became one of the preferred languages for other artificial inteliigence applications. It is a computer language that was created especially for answering questions about a knowledge base that consists of rules and facts. Prolog has built-in 'backward chaining' and also utilizes another technique known as backtracking. Backward chaining is a technique in which a conclusion or consequence is assumed to be true and then knowledge base of rules and facts is examined to see if it supports the assumption. If the assumption turns out 'not' to be correct, backtracking is said to get rid of the original assumption and replace it with a new one.

Like procedural languages, prolog does not use algorithms. Algorithms are objective procedures that when followed, guarantee a solution. Object-Oriented languages, such

as prolog [11], use heuristics. Heuristics are rules of thumb that are useful in reaching a goal. They do not guarantee a solution and are useful when no algorithms exists. Prolog uses only data about objects and their relationships. Prolog also emphasizes symbolic processing.

There is no standard for the prolog language but the closest thing to an accepted standard is the prolog described in Clocksin and Mellish's book which is generally known as C and M prolog [12]. Most commercial prologs have evolved as super sets of this core language. Turbo prolog is both a superset and a subset of C and M prolog, yet it does not support many features that are supported in the core language. In addition, turbo prolog supports some features (such as input and output) by structures more similar to those of the C language than those of C and M prolog. Programs written in C and M prolog will not run under Turbo prolog without modification. Similarly, turbo prolog programs must be extensively modified to run under C and M prolog.

6.3.2.1. **Features and Capabilities**

Let us now have a look on the features and capabilities of Turbo Prolog language.

Knowledge in turbo prolog can be expressed in terms of either facts or rules. The basic units for building facts or rules are predicates, i.e., expressions that say simple things about the individuals in our universe. For instance, the piece of information 'the left speaker of the stereo system is not emitting sound (is dead)' is represented in turbo prolog as

IS (left_speaker, dead).

The factual expression in prolog is called a clause.

In the above example, left_speaker and dead are objects. An object is the name of an element of an entity in the real world. Turbo prolog permits the user to use six different object types viz, char, integer, real, string, symbol and file. The word IS is the relation in the example. A relation is a name that defines the way in which a collection of objects (or objects and variables referring to objects) belong together.

The entire expression before the period is called a predicate. A predicate is a function with a value of true or false. Predicates express a property or a relationship. The word before the parentheses is the name of the relation. The elements within the parenthses are the arguments of the predicate, which may be objects or variables.

Here are a few examples of predicates :

Car(Maruti)
Country(India)
M_tone(Patient)

- **Rules**

Rules permit prolog to infer new facts from existing facts. A rule is an expression that indicates that the truth of a particular fact depends upon one or more other facts. Let us consider the following example.

IF the Muscle tone of limbs is flaccid AND
Heart-rate is low AND
Respiratory effort is slow irregular AND
Reflex stimulation is grimace AND
Colour is periphery blue but body pink
THEN the patient probably has to be undergone with type II resuscitation.

The rule could be expressed as the following turbo-prolog clause :

```
rescus(Patient, type II) if
    m_tone(Patient, flaccid) and
    h_rate(Patient, low) and
    res_eff(Patient, slow_irregular) and
    ref_sti(Patient, grimace) and
    colour(Patient, p_blue_body_pink).
```

The conditions upon which conclusion depends are stated next, each connected by the word and. The conclusion can be viewed as a prolog goal. The goal is true if all the conditions specified for the goal are true.

Every rule has a conclusion (or head) and an antecedent (or body). The antecedent consists of one or more premises. The premises in the antecedent form a conjunction of goals that must be satisfied for the conclusion to be true. If all the premises are true, the conclusion is true; if any premise fails, the conclusion fails.

Since rules are such an important part of any prolog program, a shorthand has been developed for expressing rules. In this abbreviated form the previous rule becomes

```
rescus(Patient, type II) :-
    m_tone(Patient, flaccid),
    h_rate(Patient, low),
    res_eff(Patient, slow_irregular),
    ref_sti(Patient, grimace),
    colour(Patient, p_blue_body_pink).
```

The :- operator is called a break. A comma expresses **and** relationship and semi-colon expresses an **or** relationship. The process of using rules and facts to solve a problem is called formal reasoning.

- **The Turbo-Prolog Program**

Turbo prolog program consists of two or more sections. The main body of the program, the clauses section, contains the clauses and consists of facts and rules. The relations used in the clauses section are defined in the predicates section. Each relation in each clause must have a corresponding predicate definition in the predicates section. The only exceptions are built-in predicates that are an integral part of turbo prolog.

The domains section is also as part of most turbo prolog programs. It depends on the type of each object.

The goal section of the program defines the internal goal. If turbo prolog can match the goal with a fact or rule in the data base, it succeeds otherwise it fails. Variables are used in a turbo prolog clause or goal to specify an unknown quantity. A variable name must begin with a capital letter and may be from 1 to 250 characters long. Except for the first character in the name, one may use upper case or lower case letters, digits or the underline character. If a variable has a value at a particular time in a program, it is said to be bound. If a variable does not have a value at a particular time, it is said to be free. For a goal to succeed, all variables in the goal must become bound.

The turbo prolog program can consist of upto eight sections. Most programs only use a few of these sections, but the complete list gives the extensive flexibility : global predicates for modular programming, dynamic databases, internal goals and compiling directives.

Although not all sections will be used in all programs, those that are used must be in the following order for the compile operation.

1. Compiler directives
2. Domains
3. Global domains
4. Data base
5. Predicates
6. Global predicates
7. Goal
8. Clauses.

- **Unification**

The process by which prolog tries to match a term against the facts or the heads of the other rules in an effort to prove a goal is called unification. Unification is a pattern matching process.

Predicates unify with each other if

1. They have the same relation name.
2. They have the same number of arguments.
3. All argument pairs unify with each other.

- **Prolog execution rules**

The rules for prolog execution are given below :

1) Prolog executes using a matching process.

2) When the original goal is specified, prolog tries to find a fact or the head (conclusion) of a rule that matches the goal.

3) If a fact is found, the goal succeeds immediately.

4) If a rule is found, prolog then tries to prove the head of the rule by using the antecedent (the body of premises) as a new compound goal and proving each premise of the antecedent. If any premise of the antecedent fails, prolog backtracks and tries to solve the preceding premises with other bindings. If prolog is not successful, it tries to find another fact or rule that matches the original goal. If all premises succeed, the original goal succeeds.

5) Turbo prolog continues to execute until all possible solutions for the goal are tested.

There are no global variables; that is, variables that maintain a value throughout the entire programs execution. All variables are local to the clause of which they are a part. Even if the same variable name is used in another clause, it is not the same variable. If a particular clause fails, prolog backtracks and tries to solve the same goal another way using clauses.

- **Built-in Predicates**

Turbo-prolog offers dozens of built-in predicates to support not only input and output operations, but graphics, file operations, string handling and type conversion. These

predicates can be used in rules or facts, just like any other predicate. They do not need to be defined in the predicate section, as they are integral to turbo prolog.

The turbo prolog predicates are classified into eight groups :

1) Reading predicates :
   For reading data from the keyboard or file to a variable.

2) writing predicates :
   For writing data to the screen, printer or a file.

3) Control predicates :
   For controlling program execution, forcing or preventing backtracking.

4) File system predicates :
   For managing disk files from a turbo-prolog program.

5) Screen-handling predicates :
   For graphic control of the display and sound control.

6) String-handling predicates :
   For various operations on string data.

7) Type-conversion predicates :
   For converting data types from one form to another.

8) System-level predicates :
   For access to MS-DOS functions from within a turbo prolog program.

A few examples of the built-in predicates are given below :

- **The read predicate**

Turbo-prolog provides several built-in input predicates. One of these is a read predicate. There are four types of read predicate : readin, readchar, readint and readreal.

The readln predicate permits a user to read any string or symbol into a variable. Let us consider the following example :

```
Symptom(Child, fever) :-
        write ("Did the ", Child, "have vomiting last night (yes / no)?"),
        readln (Reply),
        Reply = "yes".
```

When this is invoked, it displays the question and then pauses for an answer. The rule will succeed if the user enters yes and fail if the user enters anything else. In the same way, **readchar** predicate permits to read any character, **readint** to read any integer and **readreal** to read any real value into a variable.

- **The write predicate**

The write predicate displays the value of the variables or objects. The variable must be bound before the write predicate is invoked. For example, The clause

```
test :-
        write ("Sorry, I don't find "),
        write ("no more new findings").
```

displays

```
Sorry, I don't find no more new findings.
```

The output may be written in two lines using the control predicate nl. The predicate nl indicates a new line. Any output to the display can be directed to a printer or a file. To redirect output, the writedevice built-in predicate is used.

The following code directs the output of the write predicate to the printer and then redirects further output back to the screen.

```
writedevice (printer),
write ("This will print on the printer"),
writedevice (screen).
```

Likewise Turbo-prolog provides the readdevice predicate for redirecting input.

- **The fail predicate**

In prolog, forcing a rule to fail under certain conditions is a type of control and is essential to good programming. Failure can be forced in any rule by using the built-in fail predicate. The fail forces backtracking in an attempt to unify with another clause. Whenever this predicate is invoked, the goal being proved immediately fails, and backtracking is initiated. The predicate has no arguments, so failing at the fail predicate is not dependent on variable binding; the predicate always fails.

Example :

```
go:-
        test,
        write ("you will never get here").
test :-
        fail.
```

If the goal is specified as go, prolog will unify with the head of the first rule and then try to prove it premises. The test premise will unify with the head of the second rule, whose premise is fail, and the goal will fail. Prolog will backtrack to the first rule and the go goal will fail again. The write predicate will never be executed. The same program may be tried again, reversing the two premises and changing the test string slightly :

```
go :-
        write ("you will get here"),
        test.
test :-
        fail.
```

Again, the goal will fail, but the text string will be displayed this time.

- **The not predicate**

If we want to express explicitly in the database that a particular fact is not true, the built-in not predicate has to be used. The not predicate cannot be used to express a fact or appear in the head of a rule. It can only be used in a premise, as in

```
replace (left_speaker) :-
not (is (left_speaker, functional)).
```

In this case, if

```
is (left_speaker, functional).
```

is in the data base, the rule will fail.

- **The cut predicate**

The cut is one of the most important, and also one of the most complex, features of prolog. The primary purpose of the cut is to prevent or block backtracking based on a specified condition. The cut predicate is specified as an exclamation point (!). It has no arguments. The cut predicate always succeeds, and once it succeeds, it acts as a fence, effectively blocking any backtracking beyond the cut. If any premise beyond the cut fails, prolog can only backtrack as far as the cut to try another path. If the rule itself fails and the cut is the last premise, no other rules with the same head can be tried. Prolog must accept failure or success of the predicate based on that particular clause.

There are two types of cuts. In Turbo prolog, these are called the **green** and the **red** cuts. The green type of cut is used to force binding to be retained, once the right clause is reached. Green cuts are used to express determinism. A program is nondeterministic if it is capable of generating multiple solutions on backtracking. The red type of cut is used to omit explicit conditions.

The use of any type of cut in a prolog program is controversial. It implies a type of procedural control, which is in sharp contrast to the declarative style of prolog programming. If used with caution, however, cuts improve the clarity and efficiency of most programs. Of the two types of cut, the green cut is the more acceptable type. One can often use the not predicate instead of the red cut.

Example :

Let us assume a rule of the form :

```
go :-
        premise1,
        premise2, ! ,
        premise3,
        premise4.
```

Prolog will try to prove the go by backtracking between premise1 and premise2 as necessary until both are true. Once this occurs, the cut is reached. The cut predicate always evaluates as true, so testing begins on premise3. Once the cut is crossed, prolog cannot backtrack across the cut. If premise3 fails, the rule fails. If premise4 fails, prolog backtracks to premise3 to try another path for this premise, but prolog goes no further. If either premise3 or premise4 does not evaluate as true, the rule fails without going back to premises before the cut and attempting to prove them in a different way, for example, with different bindings. All variables in premise1 and premise2 are bound when the cut is crossed, and prolog is committed to all choices before the cut.

- **The makewindow predicate**

A window may be created at any time in a program by using the makewindow predicates. The general form of the makewindow predicate is

makewindow (WindowNo, ScrAttr, FrameAttr, Header, Row, Col, Height, Width)

All except the header arguments are integers. The header is a string or a symbol. The arguments are explained as follows :

WindowNo	Assigns a number to the window.
ScrAttr	Defines the video attributes of the window display.
FrameAttr	Defines the video attributes of the border.
Header	Defines a header for the window.
Row, Col	Defines the starting point (upper left corner) for the window.
Height, Width	Defines the size of the window.

Example :

```
makewindow (1, 31, 31 "A FUZZY KNOWLEDGE BASED NEONATAL RESUSCITATION
            MANAGEMENT SYSTEM ", 0, 0, 25, 80)
```

Several windows can be created with separate number. The shift between windows can be performed using shiftwindow predicate :

```
shiftwindow (WindowNo)
```

When a window is shifted, the window will be clear, with the cursor at the upper left. The cursor (Row, Col) predicate positions the cursor at (Row, Col) in the currently active window.

- **Prolog databases**

A prolog program is a collection of facts and rules about a particular knowledge domain. The program really is a database and prolog is very powerful query language for this database, permitting one to select facts from the database through unification. The program, however, is a static database; that is, the database does not change over time. To solve the medical diagnostic problem, prolog permits us to add a dynamic database to the program. Built-in predicates permit to add facts to or remove facts from this dynamic database during program execution. To store information in a dynamic database, one or more database predicates have to be created. Facts can then be stored in these predicates during program execution using the built-in **asserta** or **assertz** predicates. To creat database predicates, a database section must to be added to the program. The database predicates are defined in this section. The section must follow the domains section and precede the predicates section.

Example :

```
Domains
        disease, symptom = symbol
        query  = string
        reply  = char
database
        xpositive (symptom)
        xnegative (symptom)
predicates
        hypothesis (disease)
        symptom (symptom)
        :
        :
```

Here, two database predicates are used in the database section : xpositive and xnegative. The database predicates are not defined in the predicates section. With

these database predicates one can store the facts, learned from the questions, in the databae (facts proven true in xpositive and facts proven false in xnegative) and query the database before asking the same question again.

- **The asserta and assertz predicates**

The asserta (fact) predicate stores a fact at the begining of the database. The assertz (fact) predicate stores a fact at the end of the database. Which of these will be used in the program depends upon where one wishes to put the fact in the database.

- **The retract predicate**

Once a fact is stored in the database, it can be removed only using the built-in retract (fact) predicate.

Example :

retract (xpositive (fever))

removes the fact fever from the database xpositive (fact). To save the current dynamic database, the built-in predicate save (filename) predicate may be used.

- **File opening predicates**

There are four file-opening predicates in turbo-prolog. These are :

1.	Openread (SymbolicFilename, Filename)	Opens file for reading. If the file is not there, the predicate fails.
2.	Openwrite (SymbolicFilename, Filename)	Opens the file only for writing. Any previous file with the same name is deleted.
3.	Openappend (SymbolicFilename, Filename)	Appends any new output to the end of file-name. If filename does not exist, the program terminates execution.
4.	Openmodify (SymbolicFilename, Filename)	Opens file for writing and reading using random access.

All the predicates discussed here have the same number and type of arguments. Any of them can be used to open a file. The first argument is the symbolic file name that will be used for the file in the program. The second argument, filename, is the name of the file on the disk. A file may be accessed randomly using the filepos built-in predicate. The general form of the filepos predicate is

filepos(SymbolicFilename, Position, Mode)

where Mode determines how position is measured as follows :

0 = from begining of the file
1 = from current position
2 = from end of file.

The end of the file during a read operation may be checked using

eof (SymbolicFilename).

The test will be succeed if current position is at the end of the file; otherwise, it will fail. To prevent loss of data, closefile predicate is used to close a file after using it. The general form of the closefile predicate is closefile (SymbolicFilename).

6.3.3. **Requirements vs. Capabilities**

Knowledge bases can be exported to a text file using PRL syntax in Level5 object. PRL (Production Rule Language) is Level5 object's application development language. One can see the underlying PRL structure of his / her application when one exports it to a text file. One can edit this file and also send it to Level5 object running on other hardware platforms and operating systems. This process lets one create applications where it is most convenient and deliver them to the platforms one wants. But, however, when export of an application to be transported to character-based platforms, such as VAX / VMS or MVS, some elements will import, but will have no effect when the application is run. For example, pictureboxes do not appear in character-based displays. So, with the PRL structure one can achieve the portability.

From the PRL source text, one can have the facility of quick and easy modification of existing knowledge base. So, the proper patching work is not difficult here. With the different editors one can easily add or delete an object, its attributes, methods and the consistency checking is monitored by the Level5 object itself.

Probably, one of the limitations of current tools is in their handling of inexactness of information. Level5 object manages only one form of inexactness i.e. uncertainty, in the form of certainty factors(-2..100). MYCIN style of approach has been used here. However, it is not capable of handling other forms of inexactness as identified for our problem domain e.g. fuzzy information, simultaneous occurence of uncertainty and fuzziness, and uncertain-fuzzy. But, however, implementing NMR may be more or less easy, although, it is very difficult to pinpoint the objects / attributes / rules / methods / demons affecting the absurd conclusion. But, once, they are identified it is easy to upgrade the information in knowledge base using different convenient editors.

Level5 object provides a complete access to the inferential reasoning process while running a knowledge base. Its analytical reasoning facilities, such as season monitor, historical traces, single-stepping and breakpoint enable developers (and when necessary, end users) to view all the current status of the inference engine; examine and change the state of any facts in a knowledge base; review the answers provided to Level5 queries; and follow the line of reasoning being pursued. By activating a debug window while running an application, developers can observe and trace the reasoning process. In single-step mode, Level5 pauses after each event the inference engine processes, allowing the developer to view the action before resuming the session. So, a comprehensive explanation facility is being provided by Level5.

Although Level5 objects manages objects and attributes from editor's panel which essentially freeze the knowledge base before running an application, but, however, it can manage instances of an object dynamically i.e. during running an application using MAKE and FORGET commands. The system can learn the situations of using MAKE and FORGET for the defined rules / methods / demons. The repetitive questionnaire of same kind during interrogation with a child and/or with parents / guardians can be avoided by suitably navigating the question-answer sequence. One can save recommendations of a typical session using an external database (typically dBASE III +). The required structuredness and modularity are being assured by the object-oriented design strategy of Level5 object.

Level5 supports the development of large applications through the use of knowledge-based subroutines that allow knowledge to be grouped modularly or chained to knowledge bases. Besides being easier to maintain, subroutines enable a host knowledge base to resume processing where it left off. Developers can navigate between modules and even redefine views of the knowledge domains.

Now considering Turbo prolog, we found the following attractive features :

- In-built inference mechanism which expedites initial prototype development of the system,
- Separation between knowledge base and inference mechanism,
- Static and dynamic knowledge bases,
- Declarative as well as some procedural knowledge can be intermixed,
- Supports modular and structured programming to achieve good modifiability,
- Table look-up scheme for fuzzification and the defuzzification processes can be easily implemented using TURBO-PROLOG's list-structure,
- Backward chaining favours the medical diagnosis problems, and
- Object-Oriented language[11].

6.4. **Conclusions** and **Discussion**

In this chapter we have tried to explain some of the potential inconveniences faced by an expert system designer in selecting an appropriate expert system implementation tool. In connection with the present work, the author has expressed his observations on two tools :

(i) Level5 Object, and
(ii) Turbo Prolog.

An attempt has been made to judge the features and capabilities as discussed in this chapter. Obviously, Level5 object has certain good features and capabilities to act as an implementation tool. But, however, it is fair to state that it has rudimentary capability of handling inexactness. In this respect, Turbo prolog may be considered suitable since it is a language where one can develop his / her own architecture.

References

1. J. L. Alty. Expert system building tools. In Topics in expert system design. G. Guida and C. Tasso. (ed.). Elsevier Science Publishers B. V.; (North Holland), 1989, 181-204.

2. J. Rothenberg. Expert system tool evaluation. In Topics in expert system design. G. Guida and C. Tasso. (ed.). Elsevier Science Publishers B. V.; (North-Holland), 1989, 205-229.

3. P. Harmon (ed). The cost effectiveness of the major U.S. Expert Systems Building Tools. Expert Systems Strategies; vol.3, no.11, 1987, 14-16.

4. I. Bratko. Fast prototyping of expert systems using Prolog. In Topics in expert System Design. G. Guide and C. Tasso. (ed.). Elsevier Science Publishers B. V.; (North Holland), 1989, 69-86.

5. D. Prerau. Selection of an appropriate domain for an Expert System. The AI Magazine; Summer 1985, 26-30.

6. A. B. Baskin and R. S. Michalski. An integrated approach to the construction of knowledge-based systems : Experience with advise and related programs. In Topics in expert system design. G. Guida and C. Tasso. (ed.). Elsevier Science Publishers B. V.; (North-Holland), 1989, 111-143.

7. W. Mettrey. A comparative evaluation of expert system tools. IEEE Computer; February, 1991, 19-31.

8. J. Durkin. Expert Systems : A view of the field. IEEE Expert; April 1996, 56-63.

9. A. Bundy (ed). Catalogue of artificial intelligence tools. Second Revised ed., Springer-Verlag; N. Y.,1986.

10. LEVEL5 OBJECT for Microsoft Windows Getting Started Guide (Release 3.0). Information Builders, Inc.; USA, 1993.

11. R. S. Pressman. Software Engineering : A practitioner's approach. 3rd. ed., Mc-Graw Hill Inc.; 1992, 531-532.

12. W. F. Clocksin and C. S. Mellish. Programming in Prolog. Springer Verlag; Berlin, 1981.

UNCERTAINTY MANAGEMENT[†]

7.1. Introduction

Sometimes either we have no knowledge about an object or we have some incomplete, vague and imprecise knowledge about an object [1, 2]. Software / tool e.g. an expert system, to aid in human decision making needs to take into account the inexact nature of information expecting to lead to rational decisions. For a problem domain, different forms of inexactness may surface. Lacking any unique theory to manage all the forms as a whole, different approaches have been proposed with their own zone of applicability. The proper design and selection of a suitable scheme should be in tune with the requirements of the application domain. Moreover, the proper selection should also depend on certain important properties of a scheme like expressive power and adequacy in context to the application domain.

In section 7.2, the sources and nature of inexactness have been identified for the present problem domain. We have classified the associated inexactness of the domain in the same section. Section 7.3 deals with the tools for managing those inexactness in information. Some common approaches of dealing with inexactness in expert systems have been discussed in brief. In section 7.4, a suitability analysis has been provided in context to the present problem domain. In the last section, our conclusions and discussion have been provided.

7.2. Sources and Nature of Inexact Information

While developing an expert system which needs to reason with inexact information, one has to identify the sources and nature of inexactness for justifying the relative suitability of any method mechanising the process of reasoning with inexactness for the problem domain at hand.

We now discuss the typical physical and logical sources and nature of inexactness of information for the paediatric problem domain. In table 7.1, the possible physical sources and their nature and explanation have been presented. From the table 7.1, one can observe that there are not just one or two sources of inexactness, but seven major

---

[†] This is based on the publication [Proc. XXX Annual Conv. of CSI, 1995, Tata Mc Graw Hill Pub. Co. 258-267] of the author.

areas, which between them break down into almost, twenty sub-areas. It also indicates the varied nature of inexactness. Combining all these possible physical sources, one can identify the following possible logical sources of inexactness of information :

- Lack of adequate data,
- Inconsistency of data,
- Inherent human fuzzy concepts,
- Matching of similar rather that identical situations,
- Differing (expert) opinions,
- Ignorance,
- Imprecision in measurements,
- Lack of available theory to describe a particular situation.

In our system, five types [3] of inexactness have been classified as follows :

- **Uncertain information**

Fumbling answer from a child and / or from parents / guardians - an issue concerning lack of confidence.

Example :

Question : Did your baby have fever last night ?
Answer : Yes / No / May be or most probably.

The answer 'may be or most probably' does not offer absolute certainty about the piece of information. The degree of (un)certainty is usually represented by a numerical value. In the above example, if the answer is 'Yes' or 'No', the certainty factor (CF) is 1. But, if the answer is 'may be or most probably' the certainty factor may be in between 0.8 to 1.0 depending upon the stress of answer.

- **Fuzzy information**

Fuzziness (sharplessness) occurs when the boundary of a piece of information is not clear cut.

Example :

The fever of the child is quite high.
The condition of the child is critical.

The linguistic articulations like 'quite high', 'critical' are fuzzy terms and these terms have shades of meaning.

- **Simultaneous occurrence of uncertainty and fuzziness**

Sometimes, there are situations where uncertainty and fuzziness may occur simultaneously.

Example :

If the Rhythms (sleep and meals) of the baby is alright then the growth of the baby should be good (0.95).

Here, 0.95 is the certainty factor and 'alright', 'good' are fuzzy terms.

- **Uncertain-fuzzy**

There are situations where uncertainty can be fuzzy.

Example :

The fever of the child is very high (around 105° F). Here, 'around 105° F' is the fuzzy uncertainty and 'very high' is fuzzy term.

- **Non-monotonic nature**

Question : Does your baby take 4-5 meals/day ?
Answer : Doctor. She doesn't want to take meals.

With this reply (CF=0.9) the doctor proceeds to other investigations. After overall investigations, the doctor finds that the growth of the baby is normal. This finding, obviously, contradicts the reply from the mother whose CF may be 0.3 or 0.2. The associated facts and rules have also to be changed with this changing CF. This is how the nonmonotonic reasoning works to offer a safeguard to the subjective reply during a consultation session.

Table 7.1. Sources and nature of inexactness

SOURCES and NATURE	EXPLANATION
1. Problem domain a) Lack of precise numeric aspiration levels; b) Lack of appropriate or available well defined algorithms.	1. In a medical diagnostic system, it is very difficult to assign precise aspiration levels to some or all the controlling physiological parameters. Moreover, human body, does not follow a strict algorithmic approach for its running.
2. Child a) Lack of adequate data; b) Inconsistency of data; c) Subjective reply; d) Reply in fuzzy terms; e) Fumbling answer; f) No information.	2. Sometimes, a child may not be able to supply any verbal relevant information where language development is not adequate. The other natures of inexactness are also frequently received by a doctor during a consultation session.
3. Parents / Guardians a) Lack of adequate data b) Inconsistency of data; c) Subjective reply; d) Reply in fuzzy terms; e) Fumbling answer indicating lack of confidence; f) Ignorance.	3. Parents / guardians sometimes fail to supply complete history of the child, may be, due to their illiteracy. The other forms of reply are also frequently as received by a doctor during a consultation session.
4. Doctors Matching of similar rather than identical situations : model inequivalance.	4. Sometimes doctors are suppossed to apply heuristic knowledge gained in a number of years of practice. Any particular heuristic may not work with similar situation.
5. Laboratory tests / technicians Imprecision in measurement during clinical / pathological / radiological tests : instrumental / technician's error; impurity with the chemicals / glasswares used.	5. It is reported, sometimes,that the findings from the clinical view of a doctor may not match the findings from laboratory reports. Doctors,then advice to repeat the test for confirmation.
6. Symptoms a) Most of the symptoms are valid for more than one adverse situation; b) Information hiding.	6. A particular symptom may be reflected from two or more adverse situations. A secondary symptom may be more prominent than a primary symptom in a situation : information hiding.
7. Laboratory results not available.	7. Laboratory investigations may not be possible either due to financial constraints or there may not be any laboratory in the nearby area.

At the starting of a typical consultation session, a doctor has to interrogate the child, where possible, or the parents / guardians of the child on different issues. She / he may examine the child with his / her medical / clinical view and so on. She / he may face the above forms of inexactness of information which may be the result of the

combined conspiracy of the above discussed sources and other unidentified sources. Here, proper management of inexact information is necessary which plays a pivotal role in rational decision making.

7.3. **Tools for Managing Inexact Information**

A number of methods have been proposed to deal with different aspects of inexact information management with their varying degrees of success. In essence [4], they can take one of the seven forms such as non-numerical techniques, categorical techniques, probabilistic modelling, ad-hoc techniques, Bayesian inference, fuzzy logic, and Dempster-Shafer theory of evidence.

The common approaches in dealing with inexactness in expert systems are : Bayesian probability approach, DS-theory of evidence, Stanford CF-calculus, and Fuzzy set theoretic approach. In addition, inexact reasoning has itself non-monotonic aspect. It may be noted here that none of the methods except CF-calculus has been developed with a reference to AI and expert systems and neither has yet been universally adopted by theoreticians or practitioners.

It may be stated as a generally accepted fact that any theory designed to cover the concept of inexactness should essentially be able to provide a method for changing prior opinion in the light of new evidence : non-monotonic reasoning. For example, DS approach having a rigorous mathematical theory, handles the situation using Dempster's rule of combination.

7.3.1. **Bayesian probability theory** [5]

The Bayesian approach is based on formal probability theory and has shown up in several areas of AI research, including expert systems and pattern recognition problems. However, this particular approach can deal only with uncertainty.

One of the most important results of probability theory is Bayes' theorem. Bayes' theorem states :

$$Pr\{E_i \mid F\} = \frac{P_r\{F \mid E_j\}}{\sum_{i=1}^{n} P_r\{F \mid E_i\} P_r\{E_i\}} Pr\{E_j\} \quad \text{..... ..... 7.1}$$

where we assume that

$$Pr\{E_i\} > 0 \qquad \text{for all } i, \qquad \ldots\ldots\ldots\ldots\ldots\ldots 7.2$$

$$Pr\{E_i \cap E_j\} = 0 \qquad \text{for } i \# j \qquad \ldots\ldots\ldots\ldots\ldots\ldots 7.3$$

and

$$\sum_{i=1}^{n} P_r\{E_i\} = 1 \qquad \ldots\ldots\ldots\ldots\ldots\ldots 7.4$$

Usually, $Pr\{E_j\}$ and $Pr\{E_j | F\}$ are called Bayesian **'Prior'** and Bayesian **'Posterior'** respectively.

This theorem has been used in a number expert systems. One such notable expert system is PROSPECTOR developed [6] as a consultation system for mineral exploration. It is based on the above assumptions 7.2, 7.3 and 7.4. We may cite here at least one PROSPECTOR - like system which has been developed based on this theorem where the above assumptions are implied [7].

7.3.2. **Dempster / Shafer theory of evidence** [8,9]

The DS theory begins with the view that problem-solving activities can be described as a process of attempting to answer questions of interest [10]. Possible answers to a question are represented as a set of interrelated propositional sentences called a frame of discernment (FOD), denoted by Θ. For a given FOD, at any instant, one and only one propositional sentence is possibly true. One has to form a consensus about the truth of propositions based on opinions that bear upon the question of interest to find the correct evidences from two or more independent sources and then they are combined to get a consensus.

Within DS theory, each independent sources conveys its belief via a mass distribution m which is defined as :

$$m : 2^{\Theta} \longrightarrow [0,1] \qquad \ldots\ldots\ldots\ldots\ldots 7.5$$

where

$$m(\Phi) = 0 \qquad \ldots\ldots\ldots\ldots\ldots 7.6$$

and

$$\sum_{A \subseteq \Theta} m(A) = 1 \qquad \text{..... 7.7}$$

The quantity m(A) is called A's 'basic probability' or 'basic probability assignment' or simply 'basic assignment' and it is understood to be the measure of the belief that is committed exactly to A. To obtain the measure of total belief committed to A, one must use the following :

$$Bel(A) = \sum_{B \subseteq A} m(B) \qquad \text{..... 7.8}$$

Evidence obtained in the same context from two independent sources which are expressed by two basic assignments m_1 and m_2 on some power set $P(X)$ must be appropriately combined to obtain a joint basic assignment $m_{1,2}$. This can be achieved using Dempster's rule of combination as :

$$m_{1,2}(A) = \frac{\sum_{B \cap C = A} m_1(B).m_2(C)}{1 - k} \qquad \text{..... 7.9}$$

For A # Φ

$$K = \sum_{B \cap C = \Phi} m_1(B).m_2(C) < 1 \qquad \text{..... 7.10}$$

and

$$m_{1,2}(\Phi) = 0$$

Multiple independent opinions can be combined in any order affecting the final result since Dempster's rule is both commutative and associative. If the initial opinions are independent, the derivative opinions are independent as long as they share no common ancestors. If normalisation is performed after each pair of opinions have been pooled, then the total degree of conflict between all the knowledge sources will be lost. However, it is possible to show that deferring normalisation until the final orthogonal sum is computed will produce the same results while extracting the total degree of conflict [1].

The concept of support pair [S, U] is the basis of Dempster-Shafer theory of evidence. S, the necessary support, is the minimum support that is given to the truth value probability assignment. U, the possible support, is the maximum possible truth value probability assignment. In DS theory, a belief in hypothesis, B(H), is the sum of probabilities for all the subsets of H. Support pairs accommodate for the situations where the sum of the belief that a hypothesis is true and the belief that the negation of the hypothesis is also true, is less than one i.e.,

$$B(H) + B(\overline{H}) < 1$$

But for B(H) and $B(\overline{H})$ to be truth value probabilities, their sum must be equal to 1. Since this is not always the case, B(H) is called necessary support, and the complement of $B(\overline{H})$ is called the possible support. So,

$$S = B(H) \text{ and } U = 1 - B(\overline{H})$$

Therefore, [S, U] thus defines an evidential interval, where residual inexactness in H is given by U - S, which is sometimes referred to as the ignorance of H which is represented as Igr(H). This residual inexactness can be reduced upon learning new evidence supporting the hypothesis. Igr(H) will become zero having full evidence supporting the hypothesis; the evidential interval merges to become a point probability, the Bayesian method becomes a special case of the DS - representation. Dempster-Shafer theory has been considered as a prominent candidate to handle inexactness in expert systems.

7.3.3. **Stanford certainty factor model** [11]

If we have a hypothesis h based on an evidence e, then from some simple assumptions of certainty theory we may define "measure of belief (MB)" and "measure of disbelief (MD)". We write :

MB(h | e), the measure of belief of a hypothesis h based on an evidence e, and
MD(h | e), the measure of disbelief of a hypothesis h on an evidence e.

Now,

While MD(h | e) =0, $1 > MB(h \mid e) > 0$, or
While MB(h | e) =0, $1 > MD(h \mid e) > 0$.

We combine MB(h | e) and MD(h | e) to get the certainty factor as

$$CF(h \mid e) = MB(h \mid e) - MD(h \mid e).$$

As the certainty factor (CF) approaches 1, the evidence(e) is stronger for a hypothesis(h); as CF approaches -1, the confidence against the hypothesis gets stronger; and CF around 0 indicates that there is little evidence either for or against the hypothesis.

In a production system, different certainty factors are attached to every premises. These certainty factors are combined to get the overall certainty of the inference as follows. If p1 and p2 are two premises of an inference, then the combined certainty factors are :

$$CF(p1 \text{ and } p2) = \min(CF(p1),\ CF(p2)), \quad \text{and}$$
$$CF(p1 \text{ or } p2) = \max(CF(p1),\ CF(p2)).$$

One more measure is required, i.e., how to combine multiple CFs when two or more rules support the same result R. Here, CF - theory uses the analogy of the probability theory procedure of multiplying the probability measures to combine independent evidence. By using this rule repeatedly one can combine the results of any number of rules that are used for determining result R. Suppose CF(R1) is the present certainty factor associated with result R, and a previously unused rule produces result R (again) with CF(R2); then the new CF of R is calculated by :

$$CF(R1) + CF(R2) - CF(R1) * CF(R2) \quad \text{when } CF(R1) \text{ and } CF(R2) \text{ are +ve}$$

$$CF(R1) + CF(R2) + CF(R1) * CF(R2) \quad \text{when } CF(R1) \text{ and } CF(R2) \text{ are -ve}$$

$$\frac{CF(R1) + CF(R2)}{1 - MIN(|CF(R1)|,|CF(R2)|)} \quad \text{otherwise}$$

where |X| is the absolute value of X.

The combined certainty factor (CF) of the premises obtained from the above combining rules is multiplied by the originally assumed CF of the inference to get the new CF of the inference. This approach can deal with uncertainty only.

7.3.4. **Fuzzy set theory** [12]

This approach has been used in a commercially notable expert system REVEAL from ICL [13] which is essentially a decision support system. A number of commercial knowledge based shells have also incorporated fuzzy reasoning [14-18]. As a matter of fact, fuzzy logic has previously been used successfully in a number of knowledge based systems and the trend is good enough [19-21].

The concept of fuzzy set and fuzzy logic were introduced by Zadeh [12]. His intention of introducing this fuzzy set theory was to deal with problems involving knowledge expressed in vague, linguistic terms. Classically, a set is defined by its members. An object may be either a member or a non-member : the characteristic of traditional (crisp) set. The connected logical proposition may also be true or untrue. This concept of crisp set may be extended to fuzzy set with the introduction of the idea of partial truth. Any object may be a member of a set 'to some degree'; and a logical proposition may hold true 'to some degree'. Often, we communicate with other peoples by making qualitative statements, some of which are vague because we simply do not have the precise datum at our disposal e.g., a person is tall (we have no exact numerical value at that moment) or because the datum is not measurable in any scale e.g., a beautiful girl (for beautiful, no metric exists). Here, tall and beautiful are fuzzy sets. So, fuzzy concepts are one of the important channels by which we mediate and exchange information, ideas and understanding between ourselves. Fuzzy set theory offers a precise mathematical form to describe such fuzzy terms such as tall, small, rather tall, very tall, etc. in the form of fuzzy sets of a linguistic variable such as height. The details of the topic along with applications in the present domain have been provided in **chapter 9**.

7.3.5. **Non-monotonic reasoning**

Most of the available knowledge-based consultation systems / expert systems and different ES-shells use monotonic reasoning as their inference strategies which essentially assumes that axioms do not change and conclusions drawn from them remain true. In contrast to monotonic reasoning, nonmonotonic reasoning (NMR) proceeds with its reasoning as if the assumptions are true with their possible inexactness in the information. With its reasoning it reaches a conclusion. If one finds the conclusion absured, it is demanding at this stage to change an assumption and / or to change the associated (un)certainty values. NMR may be considered as an important feature of human problem solving and commonsense reasoning.

The information supplied by the parents / guardians or by the child himself / herself are subjective sometimes. To deal with this subjective reply, a kind of inexactness, NMR will be useful. NMR is also important and advantageous in connection with modifiability. We expect it useful to incorporate in our system as one of the measure of inexactness in information.

7.4. **Suitability Analysis**

Let us now investigate the suitability of the above method(s) of handling inexactness in information which seem(s) to be most sympathetic to the problem domain at our hand.

7.4.1. **Bayesian probability theory**

This approach works with two major assumptions : (1) All the statistical data on the relationships of the evidence with the various hypothesis are known in advance of processing starts; (2) All relationships between evidence and hypothesis are independent. Despite the commercial success of PROSPECTOR, the wide applicability of this approach is restricted and sometimes infeasible in some problem domain [2]. These asumptions are the bottlenecks of using this technique for a problem domain of diagnostic nature of child growth and development. In a medical diagnostic problem domain, it is very difficult for the domain experts to collect or estimate all prior conditional and joint probabilities. This seems to contradict the reasons of using an expert systems framework when and / or where the complete logic may not be known in advance.

For the medical domain, the assumption of independence of relationships between evidence and hypothesis cannot really be justified. The last problem arises in connection with modifiability, a desirable feature of an ES, of knowledge base. The knowledge base may have to be changed or updated due to different reasons. Particularly, when complete and sound knowledge may not be available in advance, a fact for the present domain, existing system should easily and quickly incorporate the required changes. In this situation, there is the need to rebuild all probability relationships which seems to be a cumbersome task. Considering all these factors, we find hardly any good reason to use this technique for the present problem domain at our hand.

7.4.2. **Dempster / Shafer theory of evidence**

This approach recognizes the distinction between uncertainty and ignorance by creating 'belief functions' - measurements of the degree of belief. The theory allows the decomposition of a set of evidence into separate, unrelated set of evidence. It allows us to use our knowledge to bound the assignment of probabilities to events without having to come up with exact probabilities when these may be unavailable; the situation where DS method may be a good candidate for applications like the integration of data from multiple radar sensors [22, 23]. It is concluded by O'Neill [1] that DS theory may be considered as a promising candidate for managing inexactness, as it includes PROSPECTOR's Bayesian belief functions and MYCIN's certainty factors as special cases. It also is based on a more mathematical foundation than either PROSPECTOR or MYCIN. However, we find to date, no notable expert system in the market using this model except some research applications [24]. The reason may be due to its involvement of so many numerical computations reducing the speed of inferencing and in the case of long inference chain the structure of the resulting belief functions would be very complex. One may expect its use where

the length of inference chain is of low or moderate size. Some studies are reported to reduce the computational complexity of the method using local computation technique for computing belief functions [25-26] and using some optimizing techniques [27]. However, the ways and means of using a simplification scheme seems to depend on case specific algorithms which deserves more scrutiny and thereby restricting its general use. We find no such commercially successful ES or ES-shell using this particular model. The above observations advice us, at present, not to use the technique for our present problem domain.

7.4.3. **Stanford certainty factor model**

This is a heuristic approach to the management of uncertainty. It is criticized as an ad-hoc technique. In particular, criticism from Adams may be considered worthwhile. Adams [28] concludes that the empirical success of MYCIN may be due to the fact that the chains of reasoning are short and the hypothesis involved are simple; this ideal situation may not be true for a complex system. Nevertheless, CF calculus finds its foundation among the expert system / expert system shell designers for its simplicity of use. The commercial success of MYCIN, EMYCIN, S.1, LEVEL5 etc. encourages people to use this technique for handling uncertainty. We do expect it useful for our problem domain to handle inexact information of uncertain nature.

7.4.4. **Fuzzy set theory**

In about 30 years of its existence, fuzzy set theory has been used in many areas including engineering, business, mathematics, psychology, management, semiology, medicine, image processing and pattern recognition. It may be fair to state that it has been used at length in control applications. In Japan alone, it has been reported, 2000 patents have been issued [29]. However, its applicability and usefulness are increasing interestingly in other fields as well [19-21]. In medical domain, fuzzy logic has previously been successfully used in a number of knowledge based systems [30-34]. For the paediatric problem domain, we find no such reported rigorous use. In connection with the management of inexact information in expert systems, the conventional approaches fail in four important respects [35] :

- They do not provide the means for dealing with the fuzziness of antecedents and consequents;
- They assume that the probabilities can be estimated as crisp numbers;
- They do not offer a mechanism for inference from rules in which the qualifying probabilities are fuzzy;
- The rules for composition of probabilities depend on unsupported assumptions about some conditional independence.

Fuzzy logic addresses some, but not all, of these problems. More specifically, fuzzy logic allows the antecedents and / or consequents and / or qualifying probabilities to be fuzzy. Furthermore, fuzzy logic makes it possible to estimate probabilities as fuzzy rather than crisp number.

Fuzzy set theory has done quite well as a formal mathematical system. Whether its theorems are interesting is a subjective opinion among mathematicians, but a large body of mathematical work exists. Where more work needs to be done is in establishing that fuzzy set theory actually captures something real in applicative fields and can make a pragmatic difference, for the right reasons [36].

It is tempting at this stage to use the technique as a measure of fuzzy concepts associated with the problem domain.

7.4.5. **Non-monotonic reasoning**

The information supplied by the parents / guardians or by the child himself / herself are subjective sometimes. To deal with this subjective reply, a kind of inexactness, NMR will be useful. NMR is also important and advantageous in connection with portability and modifiability. We expect it useful to incorporate in our system as one of the measures of inexactness in information.

7.5. **Conclusions** and **Discussion**

There are situations in real life where one has to reason with vague, insufficient, imprecise information to come to a rational decision. Any software / software tool developed for assisting peoples in their decision making needs to take into account the inexact nature of information. In this chapter, we have identified the potential sources and nature of inexactness in information in context to the present problem domain. We have also discussed different common approaches for managing inexactness in expert systems. There has been an attempt to analyse the relative suitability of those methods considering the problem domain of paediatrics. In this thesis, we have confined ourselves in considering certainty factor model and fuzzy set theoretic approach for managing inexactness. The future trend of expert system / expert system shell may probably emphasize more on the capability in handling both exact and inexact reasoning.

References

1. J. L. O'Neill. Plausible Reasoning. Australian Computer J; vol.19, no.1,1987, 2-15.

2. B. G. Buchanan and E. H. Shortliffe. Rule-Based expert systems : The Mycin experiments of the Stanford heuristic programming project. Addison-Wesley, Reading, Mass.; 1984.

3. A. K. Saha, M. G. Goswami, M. Chatterjee and R. K. Samanta. Child growth and development, expert systems and inexact information management. Proc. CSI-95, Hyderabad, Tata Mc-Graw Hill; 1995, 258-267.

4. P. L. Reynolds, P. W. Sanders and C. T. Stockel. Uncertainty in telecommunication network design. Expert Systems; vol. 12, no. 3, 1995, 219-228.

5. R. Duda, P. Hart and N. Nilsson. Subjective Bayesian methods for rule-based inference system. AFIPS National Computer Conference Proc.; vol.45, 1976, 1075-1082.

6. R. Duda, J. Gaschnig and P. E. Hart. Model design in the prospector consultant system for mineral exploration. D. Michie (ed.). Expert Systems in the Micro-Electronic Age. Edinburgh University Press; Edinburgh,1979, 153-167.

7. P. R. Cox and R. K. Broughton. Micro Expert Users Mannual. Version 2.1, ISIS Systems Ltd.; 1981.

8. A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. Ann. Math. Statist.; no. 38, 1967, 325-339.

9. G. Shafer. A Mathematical Theory of Evidence. Princeton University Press; Princeton, N. J., 1976.

10. L. P. Wesley. An entropy formulation of evidential measures and their application to real-world problem solving. B. Bouchon-Meunier, L. Valverde and R. R. Yager (eds). Uncertainty in Intelligent Systems. Elsevier Science Publishers B. V.; North-Holland, 1993, 145-154.

11. E. H. Shortliffe and B. G. Buchanan. A mathematical model inexact reasoning in medicine. Math. Biosci.; vol.23, 1975, 351-379.

12. L. A. Zadeh. Fuzzy Sets. Information and Control; vol. 8, 1965, 338-353.

13. M. Small. (ed). Knowledge engineering and decision support. International Computers Ltd.; Bracknell, UK 1984.

14. Leonardo. Creative logic Ltd.; Uxbridge, UK, 1987.

15. Fril. Fril Systems Ltd.; Bristol, UK, 1988.

16. Cubicalc. Hyperlogic. Escondido; California, USA, 1990.

17. Tilshell. Togai Infralogic. Irvine; California, USA, 1990.

18. K. S. Leung and W. Lam. A fuzzy expert system shell using both exact and inexact reasoning. J. Automated Reasoning; vol. 5,1989, 207-233.

19. D. Driankov, H. Hellendoorn and M. Reinfrank. An introduction to fuzzy control. Springer-Verlag; Berlin, 1993.

20. A. Kandel and G. Langholz. (eds.). Fuzzy Control Systems, CRC Press; Boca Raton, 1994.

21. M. Jamshidi, N. Vadiee and T. Ross. (eds.). Fuzzy Logic and Control. Prentice Hall, Englewood Cliffs; N. J., 1993.

22. J. D. Lowrance and T. D. Garvey. Tech. Note 307, SRI In 1983.

23. T. D. Garvey and J. D. Lowrance. J. Elec. Defense; July 1984, 1-41.

24. K. J. Cios, et al. An expert system for diagnosis of cornory artery stenosis based on TL-20L scintigrams using the Dempster-Shafer theory of evidence. Computer applications in the Bio-Sciences; Oxford University Press; vol.8, no.4,1990, 53-58.

25. P. P. Shenoy and G. Shafer. Propagating belief functions with local computations. IEEE Expert; vol.1, no.3,1986, 43-52.

26. G. Shafer, P. P. Shenoy and K. Mellouli. Propagating belief functions in qualitative Markov trees. Int. J. Approx. Reasoning; vol.1,1987, 349- 400.

27. H. Xu. An efficient tool for reasoning with belief functions, in B. Bouchon-Meunier, L. Valverde and R. R. Yager, (eds.). Uncertainty in Intelligent Systems. Elsevier Science Publishers B. V.; North-Holland, The Netherlands,1993, 215-224.

28. J. B. Adams. A probability model of medical reasoning and the MYCIN model. Math. Biosciences; vol.32,1976, 177-186.

29. T. Williams. Fuzzy logic simplifies complex control problems. Computer Design; March 1991, 90-102.

30. K. P. Adlassnig. A fuzzy logical model of computer-assisted medical diagnosis. Methods of Information in Medicine; vol.19,1980, 141-148.

31. K. P. Adlassnig. Fuzzy set theory in medical diagnosis. IEEE Trans. Systems, Man and Cybernatics; vol. 16,1986, 260-265.

32. K. P. Adlassnig and G. Kolarz. Representation and semiautomatic acquisition of medical knowledge in Cadiag-1 and Cadiag-2. Computers and Biomedical Research; vol. 19,1986, 63-79.

33. E. Binaghi. A fuzzy logic inference model for a rule-based system in medical diagnosis. Expert Systems; vol.7, no.3, 1990, 134-141.

34. K. J. Cios, I. Shin and L. S. Goodenday. Using fuzzy sets to diagnose coronary artery stenosis. IEEE Computer; March 1991, 57-63.

35. L. A. Zadeh. Why the success of fuzzy logic is not paradoxical. IEEE Expert; August 1994, 43-46.

36. B. Chandrasekaran. Broader issues at stake : a response to Elkan. IEEE Expert; August 1994, 10-13.

PROTOTYPE DEVELOPMENT : A SOFTWARE ENGINEERING ISSUE†

8.1. Introduction

Expert system development is the first and foremost Software engineering [1]. This is reflected in the importance of such issues as portability, integration, data base access, fielding, maintainability, robustness, reliability, concurrent access, performance, user interface, debugging support, and documentation. Two important contributions of ES-technology might be pointed out here (i) it has shown how to encode knowledge explicity and declaratively rather than implicity and procedurally, and thereby making programs more understandable and maintainable; and (ii) it is the pioneer of the new software development strategy of prototyping and refinement as opposed to the classical phase refinement approach.

Perhaps the most difficult part of constructing a large software system is deciding exactly what to construct. Requirements analysis, as software engineers call it, is the most crucial phase in the life of a project. An error in this phase can not only add to the time required for completing the project, but also lead to a delayed product that was not required in the first place [2].

The difficulty arises from two sources. Typically, the would-be users do not quite know what they want, at least not until they have tried out some version of the program. Compounding this difficulty is the fact that the planner of any software design activity does not have an exhaustive repertoire of questions that will yield him the necessary information. Neither can he be sure that the specifications he has derived are complete; chances are, they are not. Rapid prototyping has been touted as a way of this deadlock [3].

In the next section we have discussed the concepts of prototying and prototyping cycle. In section 8.3, a comparison has been made for phase refinement with prototyping approach. In section 8.4, the evolution stages of an expert system have been discussed. Section 8.5 presents our prototype 1.0. In section 8.6, our conclusions and discussion are summarized.

---

† This is based on the publications [Proc. Nat. Conf. on Software Engineering and its Application, Hyderabad, India, August 23-25, 1996, 127-134; CSI Communications, May 1997,15-21, ibid June 1997, 21-25] of the author.

8.2. **Prototyping and prototyping cycle**

Prototyping is the process of developing a scaled-down version of a system to use in building a full-scale system. The primary purpose of prototyping is to reduce time and expense in building quality systems. Prototyping mandates a philosophy of incremental system development that includes end users in the assessment of emerging system capabilities. Increased participation of end users leads to faster system development and ultimately to more useful systems [4]. Although recent innovations in prototyping [4-6], suggests the importance of end-user involvement, they do not go far enough to bring the end user into software development.

In contrast to traditional prototyping, knowledge engineering techniques commonly employ domain experts (experts in the problem area) as representative end users [7]. Prototype development in knowledge engineering sessions progresses through incremental refinement of a domain model, typically extending over two or more years.

Fig. 8.1 illustrates the iterative prototyping cycle [8]. The user and the designer work together to define the requirements and specifications for the critical parts of the envisioned system. The designer then constructs a model or prototype of the system in a prototype description language at the specification level. The resulting prototype is a partial representation of the system, including only those attributes necessary for meeting the requirements. It serves as an aid in analysis and design rather than as production software.

During demonstrations of the prototype, the user evaluates the prototype's actual behavior against its expected behavior. If the prototype fails to execute properly, the user identifies problems and works with the designer to redefine the requirements. This process continues until the user determines that the prototype successfully captures the critical aspects of the envisioned system.

The designer uses the validated requirements as basis for designing the production software. Additional work is often needed to construct a production version of the system. For example, the prototype :

a) might not include all aspects of the intended system,

b) might have been implemented using resources that will not be available in the actual operating environment,

c) might not be able to handle the full workload of the intended system, or

d) might meet its timing constraints only with respect to linearly scaled simulated time.

Experience with production use of a delivered system often leads to new customer goals, triggering further iterations of the prototyping cycle.

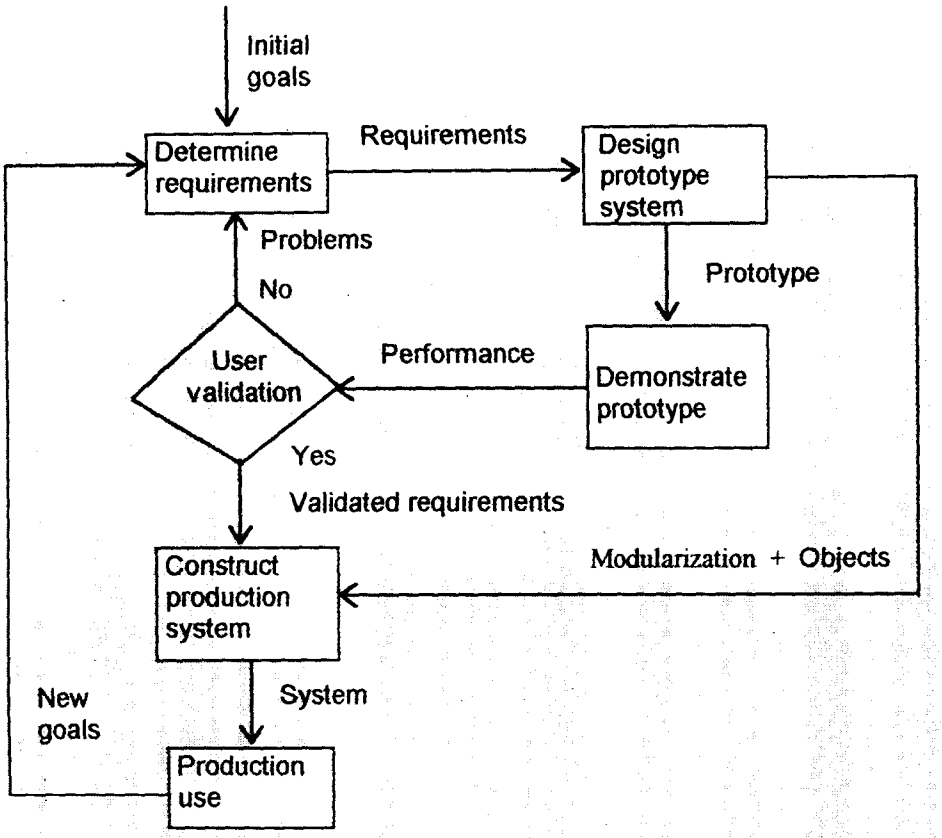


Fig. 8.1 The prototyping cycle

8.3. **Phase refinement vs. prototyping**

The classical phase refinement approach of system design is criticized for its high expense and long time for development of quality systems. Prototyping has been proposed for system development which reduce time and cost substantially. Prototyping is a paradigm which consists of some non-standard concepts and suggests increased participation of end users. Prototypes are subject to frequent and repeated changes to incorporate the suggestions from the human experts as well as from the end users. To find the benefits of prototyping than a conventional approach one has to apply : computer aided prototyping and object-oriented prototyping [8]. The first one provides mechanical assistance and the second one provides conceptual simplicity. To make the prototypes more flexible as well as achieve automation easier. O-O approach may be suitable.

While the above analysis might lead one to believe that the prototyping paradigm has all the good things in connection with the development of quality systems, the paradigm does have some bottlenecks. After evaluating the software request one has to determine whether the software to be developed is a good candidate for prototyping. Not all software is amenable to prototyping. Owing to poor project discipline, prototyping may increase cost and time. A number of prototyping candidacy factors [9] can be defined : application area and complexity, customer characteristics and project characteristics. In general, any application that creates dynamic visual displays, interacts heavily with a human or demands combinatorial processing that must be developed in an evolutionary fashion is a candidate for prototyping [10].

For a problem domain of large and varied knowledge base, an expert system should be developed in an evolutionary framework. The designers usually follow five stages as identified by Waterman [11] (section 8.4). Prototyping may have an adverse impact on modifiability and maintainability of knowledge bases since these may be patched and modified several times during the process of evolution of an expert system to get it in its commercial form. This may, however, be overcome by the use of O-O approach. As the system grows, the major changes will be with the addition of new and deletion of old objects rather than modifying the old objects. In this respect O-O approach is considered very useful for rapid prototyping.

8.4. **Stages of expert system evolution** [11]

Waterman identifies the following **five** stages in the evolution of an expert system :

Development Stage	Description
Demonstration prototype	The system solves a portion of the problem undertaken, suggesting that the approach is viable and system development is achievable.
Research prototype	The system displays credible performance on the entire problem but may be fragile due to incomplete testing and revision.
Field prototype	The system displays good performance with adequate reliability and has been revised based on extensive testing in the user environment.
Production model	The system exhibits high quality, reliable, fast and efficient performance in the user environment.
Commercial system	The system is a production model being used on a regular commercial basis.

8.5. Prototype 1.0

We now present our first demonstration prototype. During the total system development, we have taken care of the "five-stage" development suggestions of Buchanan et. al. [5]. Konwledge was acquired from the source shown in **Appendix A**.

8.5.1. General description

The architectural components of our system is shown in fig. 8.2. There are two key aspects involved in the process of design an expert system : Knowledge Base(s) and Inference Engine. The knowledge gathered from domain experts is stored in knowledge base (KB). The KB consists of two parts : static part and dynamic part. The static part is relatively fixed over time. The dynamic part is capable of adding new facts or facts can be removed from the KB as when required. After entering the relevant knowledge, one can save the KB(s) in the external storage for later use.

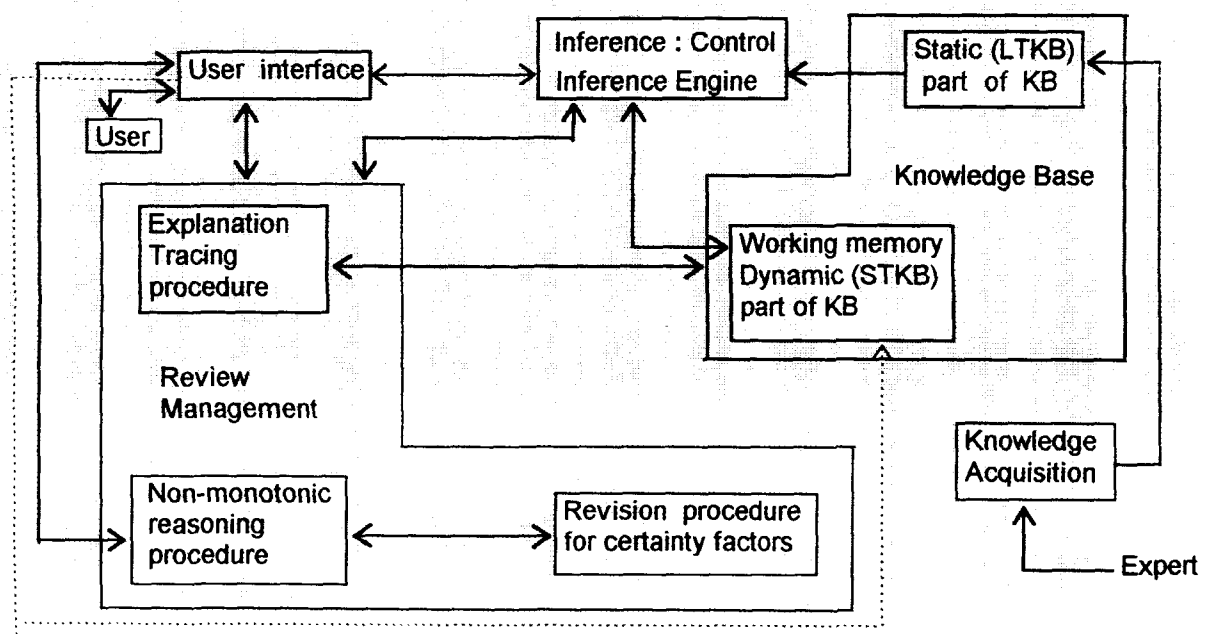


Fig. 8.2. Architectural components of the system

The inference engine uses LTKB and STKB to infer new facts. Backward reasoning process has been used here which favours the needs of the application domain. It has been shown below that the inference engine uses depth-first scanning but with an 'improved back tracking'.

A user interacts with the system with the user interface of the system. Through this module different queries are served by the system initiated by the inference engine.

The total review management is transparent to the user through this particular module. All accesses by a user to KB and review management module are through inference engine. However, logical access is presented through broken line of fig. 8.2. Knowledge acquisition module is responsible for enhancing the system knowledge by the knowledge engineer as when acquired from domain experts.

A more detail discussion on the organization of the knowledge base, the inference engine and review management is provided in the following sub-sections.

8.5.2. **Knowledge Base**

The domain experts provide the knowledge of the problem area. After extracting the necessary knowledge from the domain experts, one has to implement that knowledge in a correct and efficient knowledge base. It is the most important part of the expert system which is made separated from the inference engine with some well-known advantages. The knowledge base consists of two parts : static part and dynamic part. The static part of the knowledge base does not change over time on long term basis which may be termed as long-term knowledge base (LTKB). The dynamic part of the knowledge base is used from where facts can be added to or facts can be removed from the knowledge base as when required. This dynamic part is used on short-term basis which may be termed as short-term knowledge base (STKB). Facts obtained from the user during consultation session that apply only to the current consultation are stored in this dynamic part. This STKB may act as a part of the nonmonotonic reasoning process. This STKB also helps in achieving 'improved backtracking' which will be explained in sub-section 8.5.3. The dynamic knowledge base is stored in memory with the static knowledge base. One can save the dynamic as well as static knowledge base in a secondary storage device for later use.

8.5.3. **Inference Engine**

The inference engine has two functions : inference and control. Inference is the basic formal reasoning process which involves matching and unification. Such inference operates by modus-ponens. The control function determines the order in which the rules are tested and what happens when a rule succeeds or fails. The control function must also handle the well-known problem of conflict resolution. The inference engine scans the rules using backward chaining during the consultation process; i.e. starts at the goal and works backward. In this type of application of medical diagnosis, backward chaining is useful because the questioning is guranteed to follow the focused goal conclusion. The scanning used in our application is limited to depth-first nature. As a general principle, all rules relative to a particular goal are scanned as deeply as possible for a solution before it backtracks and tries an alternative goal. However, we may control that backtracking using STKB and using

some control rules. The STKB also helps in eliminating repetitive questioning during consultation session. This results in considerable speed up during inferencing. For the purpose, in the design we have added some control aspects using some control rules that are not a part of the knowledge base proper. The static knowledge base, then really contains two types of rules : knowledge base rules and control rules. Using some control structures we may eliminate certain search paths in problem space resulting in 'improved backtracking' provided that sufficient domain knowledge is available. The overhead of conventional chronological backtracking where no prior knowledge about the prospective backtrack point is available, can be much reduced using 'improved backtracking'.

8.5.4. **Review Management**

A review on the conclusions can be made during the consultation sessions. Under the general heading of review management, there are mainly three procedures involved, namely, Explanation tracing procedure, Nonmonotonic reasoning procedure and Certainty factor revision procedure. The Explanation tracing procedure provides explanations to mainly 'How' a conclusion has been achieved. The system may provide a sequence of 'fired rules' to achieve a particular goal. ' WHAT IF ' type of review is provided with a combination of nonmonotonic reasoning and certainty factor revision procedure. ' WHAT IF ' review can be used to find out WHAT conclusion will be deduced IF certain certainty factors are changed.

8.5.5. **Implementation**

For the present implementation we have taken the language Turbo Prolog. The basic reasons behind the choice are as follows: (1) It has its fast prototyping capability of developing expert systems [12], (2) In-built inference mechanism which expedites initial prototype development of the system, (3) Separation between knowledge base and inference mechanism, (4) Static and dynamic knowledge bases: the dynamic portion of the knowlwdge base remembers the results of the previous consulting session which may be considered as an intelligent activity of the system. In addition, facts can be added to or can be removed from the dynamic portion of the knowledge base which is one of the demanding characteristics of nonmonotonic reasoning, (5) Declarative as well as some procedural knowledge can be intermixed: this is a demanding characteristic of the object oriented approach and which also resolves declarative-procedural knowledge controversy in a restricted manner, (6) Supports modular and structured programming to achieve good modifiability and to satisfy the demand of the object oriented approach, (7) Table look-up scheme for fuzzification and the defuzzification processes can be easily implemented using TURBO-PROLOG's list-structure, and (8) The backward chaining favours the medical diagnosis problems.

8.5.6. **Analysing Process**

To initiate its analysis activity, an expert system requires some initial information about the application domain. The system applies rules on these facts and new facts are generated. Recursive application of this process leads to the goal, if satisfied. Dynamically generated new facts / results are stored in STKB for comparison, explained later on. In the process, different certainty factors of premises are supplied in actual runs. The system will ask for these factors and we should supply these from parents' informations about the child. It will calculate the certainty factors of the inferences of its own from the supplied factors of the premises. For a typical analysis, we have considered those activities as "highly likely abnormal" whose certainty factors are less than 0.6. To compare the performance of the child's current data with the previous consultation date, we preserve facts / results of one previous session in STKB. During the current session we compare the current CFs with the previous CFs. It calculates the percentage of increase or decrease or no change of performance of a child. This activity comparison is essential to comment on the overall performance profile of a child. For a typical conclusion, it calculates the overall performance and will display a message like "The growth has decreased in 45% cases". The variation in activity by 10% has been considered negligible. Then it draws a general conclusion like "The growth of the baby is expected to be abnormal". It is essential then to know the activites which are abnormal for the child. The system displays those abnormal activities which should get special attention for treatment. In addition, user may want to see the tracing of reasoning to goal for his / her satisfaction. The system may display the firing rules with their corresponding CFs. This tracing of reasoning to goal also helps a medical professional to get an idea about the sequence of firing rules with CFs for further actions. If one is not satisfied with the conclusion, he / she may want to change the CF of a rule at this stage. The supply of subjective informations by the parents / guardians of a child needs this revision of certainty factors. The system accommodates this revision and a fresh conclusion will be drawn. If the user is fully satisfied with the conclusion drawn, it will move to the next session, if desired by the user.

8.5.7. **Consulting the system**

We now present an excerpt from a typical consultation session with the system.

```
                    WELCOME TO
THE CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOPMENT

Reference Number    : r3 - 6 - 13
Name of the baby    : Jai
Father's Name       : Dhajen
Mother's Name       : Mira
Address             : Shibmandir
Age                 : One Month

                                Press any key to continue ...
```

Do you want to compare certainty factors (y/n) ? y

Certainty Factors			
Activity	Current value on 13/6/1994	Previous value on 12/5/1994	Comments
Axial muscle tone	0.3	0.8	Decreased by 50%
Muscle tone of limbs	0.6	0.9	Decreased by 30%
Spontaneous gestures	0.6	0.8	Decreased by 20%
Visual	0.6	0.9	Decreased by 30%
Gripping	-	-	No change
Relation	0.7	0.8	Decreased by 10%
Activity	0.79	0.8	Decreased by 1%
Language	-	-	No change
Rhythem	0.7	0.8	Decreased by 10%
EEG	0.99	0.9	Increased by 9%
			Press any key to continue ...

Do you want to know the general conclusion (y/n) ? y

A CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOPMENT

Growth of the baby decreasing in 45.45% cases.
The variation upto 10% has been neglected.

CONCLUSION

The growth of the baby is expected to be abnormal.
The activities which are most abnormal for the child:
Axial_Muscle_Tone

Press any key to continue ...

Do you want to trace of reasoning to goal (y/n)? y

A CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOMENT

Following sequence of rules leads to conclusion :
query(group_1, one)
activity(axial_ muscle_ tone,one)
by firing rule 'minm' :
Mininum value is : 0.3
i.e. CF of the above rule is : 0.3
activity(muscle_ tone_ of_ limbs, one)
by firing rule 'minm' :
Minimum value is : 0. 6
by firing rule 'minm1' :
Minimum value is : 0. 6
by firing rule 'minm2' :
Minimum value is : 0. 6
by firing rule 'minm3' :
Minimum value is : 0. 6
i.e. CF of the above rule is : 0. 6
activity(spontaneous_ gestures, one)
by firing rule 'minm' :
Minimum value is : 0. 8

Press any key to see next ...

:
:

A CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOPMENT
:
:

By firing 'comparing - factor' it concludes :
Growth of the baby decreasing in 45.45% cases

Press any key to continue ...

Hope you are satisfied (y/n) ? n
Do you want to change current value of any certainty factor
(y/n) ? y
To change the CF of an activity, give the activity no : 3
Enter new CF : 0.8
Do you want to change more (y/n) ? n

```
A CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOPMENT

The current data has been stored
Please wait for further information ...

        Growth decreasing in 36.36% cases
        The variation upto 10% has been neglected

                C O N C L U S I O N

The growth of the baby is expected to be abnormal
The activities which are abnormal for the child :
Axial_Muscle_Tone

                                        Press any key to continue ...
```

Do you want to trace of reasoning to goal (y/n) ? n
Hope you are satisfied (y/n) ? y
Do you want to consult further (y/n) ? n
Good bye friend ! See you !

The above excerpt clearly demonstrates how you can interact with the system. The reply from the system is displayed through screen-based frames for compactness of relevent informations.

8.6. Conclusions and Discussion

This chapter provides a prototype of our developed system. A small review of prototyping and prototyping cycles have been provided. A comparison has been made between phase refinement and prototyping. Five stages in the evolution of an expert system as identified by Waterman have been discussed. The architecture of the system have been presented and discussed. Typical consultation sessions have been presented.

Uncertainty in informations has been considered in terms of certainty factors in MYCIN style. Other forms of inexactness (e.g., fuzzy informations) will be presented in the later parts of the thesis. The present system contains static as well as dynamic databases which offers an additional intelligent activity of the system. The system is also portable and can be run on IBM-compatable systems. Last of all, we must mention that the system in its present form may not be used in a real medical diagnosis which needs more standarization. The standarization may be achieved through users' participation and through the knowledge acquisition of different experts in the

domain. The issue of system validation and testing has been provided at the later stages of our development.

References

1. J. Rothenberg. Expert system tool evaluation. Topics in expert system design, G. Guida and C. Tasso, eds., Elsevier Science Publishers B. V. ;(North-Holland), 1989, 228.

2. F. P. Brooks. No silver bullet. IEEE Computer; vol. 20, no. 4, April 1987, 10-19.

3. R. S. Pressman. Software Engineering. McGraw-Hill; New York, 1987.

4. Bernard H. Boar. Application prototyping : A requirements definition strategy for the 80's. John Wiley and Sons; New York, 1984.

5. B. G. Buchanan et. al. Constructing an Expert System. In Building Expert Systems, Frederick Hayes-Roth, Donald A. Waterman, and Douglas B. Lenat. (eds.), Addison-Wesley, Reading, Mass.; 1983, 127-167.

6. Barry W. Boehm. A spiral model of software development and enhancement. IEEE Computer; vol.21, no.5, May 1988, 61-72.

7. Robert R. Hoffman. The problem of extracting the knowledge of experts from the perspective of experimental Psychology. AI Magazine; vol.8, no.2, Summer 1987, 53-67.

8. Luqi. Software evolution through rapid prototyping. IEEE Computer; vol.22, no.5, 1989,13-25.

9. B. Boar. Applications Prototyping. Wiley-Inter Science; 1989.

10. Roger S. Pressman. Software Engineering - A Practitioner's Approach. (3rd ed.), McGraw-Hill, Inc.; 1992.

11. Donald A. Waterman. A guide to expert systems. Addison-Wesley, Reading, Mass.;1986, 136-141.

12. Ivan Bratko. Fast prototyping of expert systems using prolog. Topics in expert system design, G. Guida and C. Tasso, eds., Elsevier Science Publishers B. V.; (North-Holland), 1989.

FUZZY CONCEPTS IN PAEDIATRICS [1]

9.1. Introduction

In medical domain in general, and in paediatric domain in particular, doctors frequently have to take decisions based on vague and imprecise knowledge which can be called inexact knowledge which comes from our linguistic articulations. Fuzzy logic and fuzzy set theory has been proposed as a convenient framework for dealing with such vague, linguistic articulated knowledge. This chapter presents ideas on a research direction incorporating an outline of some fuzzy concepts in paediatrics in order to design a powerful expert system which needs to take into account such fuzzy concepts. It is also intended to present fuzzy knowledge based prototype systems as applications of the theory.

To understand the meaning of the term "**fuzzy**", we may take the weight of a baby as an example. The weight of a baby can be expressed by different vague linguistic articulations such as **good**, **bad**, **satisfactory** etc., instead of exact numeric measurements. Such linguistic articulations like **good**, **bad** or **satisfactory** are fuzzy terms (values) for the variable weight. These terms have shades of meaning. For example, the weight of many babies may be described as **satisfactory** but their exact measurements may differ markedly. Terms like **good**, **satisfactory** etc., do not assume a simple unique value but a range of values. The boundaries of such linguistic articulations are not sharp but fuzzy which essentially means that the decision path from **satisfactory** to not **satisfactory** is a gradual progression which has no sharp boundary. Moreover, if the weight of a baby is decreased by some grammes only, he / she can retain the value **satisfactory**.

Our real-life reasoning systems often use some inexact knowledge of fuzzy form leading to some rational decisions. However, as the knowledge itself is inexact, the derived decisions can not be exact but approximate, having a good rationality.

In about 30 years of its existence, fuzzy set theory has been used in many areas including engineering, business, mathematics, psychology, management, semiology, medicine, image processing and pattern recognition. Its applicability and usefulness

---

[1] This is based on the publications [Proc. xxx Annual Conv. of CSI; 9 -12 Nov 1995, 258-267, Hyderabad; Proc.3rd. Int. Conf. on Cognitive Systems (ICCS'97) 13-15th Dec 1997, vol.2, 627- 636, Delhi; and VIVEK: a Quarterly in Artificial Intelligence, vol.11, no.1, 1998, 3-15] of the author.

are increasing interestingly in diverse fields. In chapter 7, we had a discussion in this connection. Concentrating on medical domain, fuzzy logic has previously been used in a number of knowledge based systems [1-7]. For paediatric problem domain Ong and Qiu-He [8] report on interesting application. Here, the investigators have applied fuzzy logic for the diagnosis of convulsions in children. Using only symptoms (without CSF test) of twenty five patients to make diagnosis, their system achieved an accuracy of 92% compared to 67.7% on the average as observed from the doctors' diagnosis in similar conditions [9]. However, in our opinion, the potential of fuzzy logic has not been exploited at length to cover different aspects of the paediatric domain. This motivated us to explore such possibility of use of this logic in paediatric problem domain.

In the next section, we discuss the basics of fuzzy logic and fuzzy set theory. Section 9.3 presents fuzzy concepts in neonatal problem domain. In section 9.4, a fuzzy knowledge based consultation system (Prototype 2.0) has been presented using Appendix A. In section 9.5, we present a fuzzy knowledge based neonatal resuscitation management system(Prototype 3.0). In the last section, we draw our conclusion.

9.2. Basics of Fuzzy Logic and Fuzzy Set Theory

The concept of fuzzy set and fuzzy logic were introduced by Zadeh [10]. Zadeh was working in the field of control engineering. His intention in introducing this fuzzy set theory was to deal with problems involving knowledge expressed in vague, linguistic terms. Classically, a set is defined by its members. An object may be either a member or a non-member : the characteristic of traditional (**crisp**) set. The connected logical proposition may also be true or false. This concept of crisp set may be extended to fuzzy set with the introduction of the idea of partial truth. Any object may be a member of a set 'to some degree'; and a logical proposition may hold true 'to some degree'. Often, we communicate with other people by making qualitative statements, some of which are vague because we simply do not have the precise datum at our disposal e.g. a person is **tall** (we have no exact numerical value at that moment) or because the datum is not measurable in any scale e.g. a **beautiful** girl (for **beautiful**, no metric exists). Here, **tall** and **beautiful** are fuzzy sets. So, fuzzy concepts are one of the important channels by which we mediate and exchange information, ideas and understanding between ourselves. Fuzzy set theory offers a precise mathematical form to describe such fuzzy terms such as **tall**, **small**, **rather tall**, **very tall**, etc. in the form of fuzzy sets of a linguistic variable. To represent the shades of meaning of such linguistic terms, the concept of grades of membership (μ) or the concept of possibility values of membership has been introduced. We write $\mu(x)$ to represent the membership of some object to the set X. Membership of an object will vary from full membership to non-membership :

- $\mu = 0$ for no membership;
- $\mu = 1$ for full membership;
- $0 < \mu < 1$ for partial membership.

Any fuzzy term may be described by a continuous mathematical function or discretely by a set of pairs of values {numeric values of linguistic variable, corresponding grade of membership}.

For example, '**tall**' may be described by a sigmoid as shown in fig. 9.1.

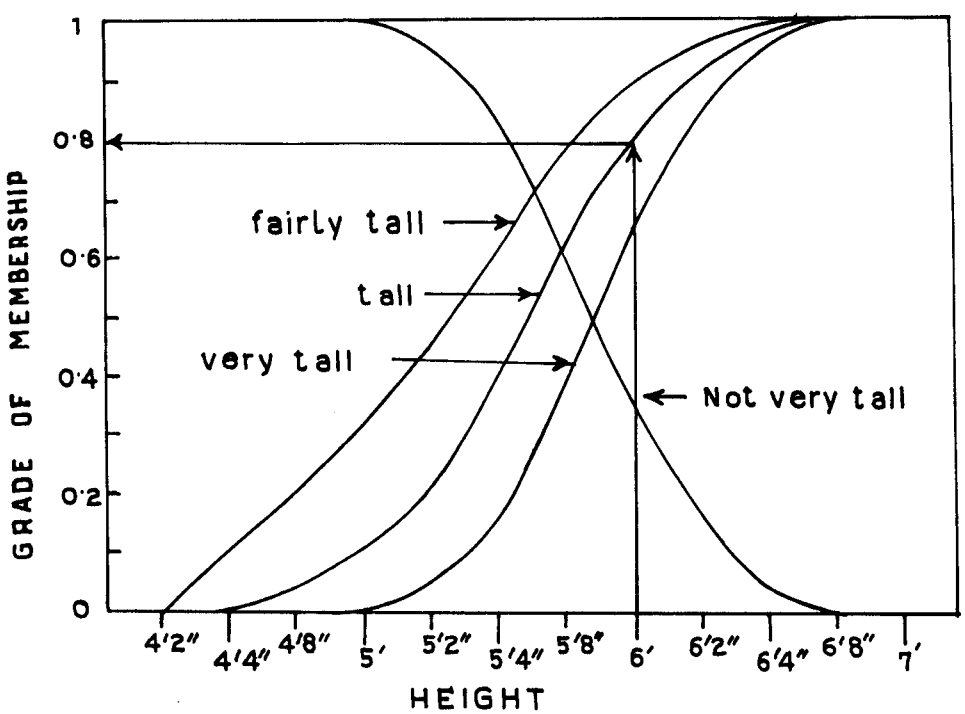


Fig. 9.1 Fuzzy term tall with modifiers

The fuzzy term '**tall**' can also be described by the following fuzzy set :

Height	Grade of membership
4'0"	0.00
4'2"	0.00
4'4"	0.01
4'8"	0.04
5'0"	0.10
5'2"	0.20
5'4"	0.38
5'8"	0.62
6'0"	0.80
6'2"	0.92
6'4"	0.98
6'8"	1.00
7'0"	1.00

Every element of the fuzzy set will have its corresponding membership value in this range (fig.9.1). Having the numerical representation of these linguistic terms, one has to define the set theoretic operations of union, intersection and complementation along with their logical counterparts of conjunction, disjunction and complementation which are as follows :

- Union (logical OR) - the membership of an element in the union of two fuzzy sets is the larger of the memberships in these sets.

  μ(A OR B) = max(μ(A), μ(B)) e.g.,
  μ(tall OR small) = max{μ(tall), μ(small)}

- Intersection (logical AND) - the membership of an element in the intersection of two fuzzy sets is the smaller of the memberships in these sets.

  μ(A AND B) = min(μ(A), μ(B)) e.g.,
  μ(tall AND small) = min{μ(tall), μ(small)}

- Complement (logical NOT) - the degree of truth of the membership to the complement of the set is defined as (1 - membership).

  μ(NOT A) = 1 - μ(A) e.g.,
  μ(NOT tall) = {1 - μ(tall)}

Example :

In fig. 9.1, we may consider two fuzzy sets : **fairly tall** and **not very tall**. The height 5'4" has a grade of membership of 0.62 in the first set, and 0.86 in the second. Thus, the grade of membership in the combined set **fairly tall** AND **not very tall** is min (0.62, 0.86) = 0.62.

The grade of membership in the combined set **fairly tall** OR **not very tall** is max (0.62, 0.86) = 0.86. And, the grade of membership in the set **NOT very tall** = 1 - μ (**very tall**) = 1 - 0.14 = 0.86.

Fuzzy numbers, like ordinary numbers, can be used in different arithmetic operations like addition, multiplication etc. that give another fuzzy number as the result as shown in fig. 9.2.

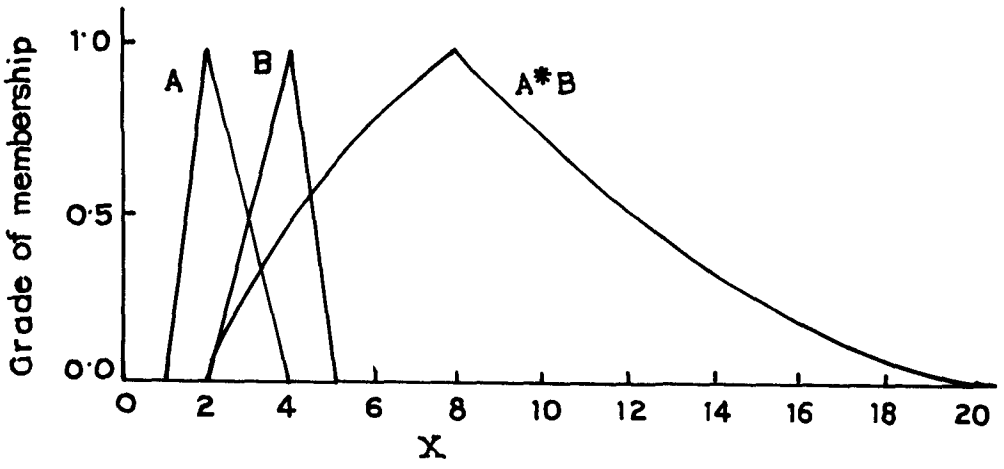


Fig. 9.2 Multiplication of two fuzzy numbers

Moreover, some fuzzy modifiers or 'hedges' such as **very**, **around**, **rather**, **quite**, **fairly**, **extremely** are common in our real-life knowledge-transfer. One can obtain the possibility distribution of a fuzzy concept like **very tall** or **fairly tall** by applying arithmetic operations on the fuzzy set of the basic fuzzy term **tall**. Power factors are a simple and convenient way for the required arithmetic operations since the grade of membership of a fuzzy set falls in the interval [0,1]. For example, we can calculate the possibility values of each height in the fuzzy set representing the fuzzy concept **very tall** by taking the square of the corresponding possibility values in the fuzzy set of **tall** (fig.9.1). Similarly, we can tackle the situation for **fairly tall** by using the square root operation on fuzzy set **tall** (fig.9.1). If one wants to generate fairly tall but not very tall, the following procedure may be followed. First we generate the set **fairly tall** by taking the square root of **tall**; then the set **very tall** by squaring **tall**; then the set **not very tall** by negating the set **very tall**. Finally, we take the intersection of these two sets (representing **but** by the **and** operation) producing the final set.

Possibly much human reasoning is based on the concepts of implications which states that :

IF antecedent THEN consequent.

For example,

IF X is A THEN Y is B.

If **antecedent** is true, **consequent** will also be true. This is **modus ponens** inference. However, there are other ways of reasoning from implications such as **modus tollens** and **hypothetical syllogism**. **Modus tollens** is based on the reasoning from data about Y to a conclusion about X. In **hypothetical syllogism**, an implication relating X to Y is combined with an implication relating Y to Z to yield an implication relating X to Z. Researchers in fuzzy logic have explored fuzzy versions of all of these, but only modus ponens has seen applications in expert systems. The present author, obviously, will continue discussion using **modus ponens** approach.

Often human knowledge is expressed in such a way that antecedent and / or consequent may contain fuzzy and / or crisp values. The following table may be considered valid :

Antecedent	Consequent
crisp	crisp
crisp	fuzzy
fuzzy	fuzzy

Note here that for 'fuzzy A' degree of truthness of B should not be greater than that of A. Fuzzy logic allows both the antecedent and consequent to be fuzzy propositions. These fuzzy propositions comprise statements involving linguistic variables, which will have shades of meaning or varying degrees of truth. An antecedent of any rule may be a simple clause (or atomic propositions) or may be a combination of number of clauses connected via the fuzzy logical operators AND, OR, NOT AND and NOT OR. For examples :

Example 1.

IF

Rhythms (sleep and meals) of a baby is satisfactory

THEN

Growth of the baby should be good.

Example 2.

IF

Production volume is high

AND

Flexibility is high

THEN

Variety is high [11].

Recently, however, some researchers [8] have suggested an extension to the basic conjunctions of AND and OR by incorporating the ADD and REL conjunctions for the linkage of premises in conventional rules.

Now, we have a fact 'Rhythms of a baby is highly satisfactory' which matches with the rule as stated in above example 1. We have to find out the corresponding conclusion which will be reflected from the consequent section of the rule. The fuzzy concepts satisfactory and good can be modeled by a fuzzy relation R, represented by a matrix. Let F_1 and F_2 be the fuzzy sets representing the concepts satisfactory and good, respectively. One can obtain the fuzzy relation R [12] by performing some fuzzy operations on F_1 and F_2 , expressed as vectors. Different approaches have been proposed to compute the fuzzy relation R. One may use the Cartesian product $F_1 \times F_2$ to get the mapping or relation $R(F_1 \rightarrow F_2)$.

The fuzzy concept highly satisfactory can be represented by a fuzzy set F, which we may obtain by applying an arithmatic operation (a square operation in this case) on F_1 . The fuzzy set C representing the effect or conclusion after the application of the said fact can be deduced by applying a fuzzy operator called composition operator (denoted by $\mathbf{O}$) on F and R i.e.,

$$C = F \; \mathbf{O} \; R$$

A number of forms for composition operator has been suggested to compute C. Most of the developers of expert systems prefer the **max-min** composition rule of Zadeh [13]. Obviously, the vector C will indicate very good and the conclusion growth of the baby should be very good will be drawn.

9.3. Fuzzy Concepts in ' NEONATES' Problem Domain

The knowledge base(s) used by a typical computer-based expert system often comprises vague, linguistic rules as articulated by domain experts, i.e., Paediatricians in the field of paediatrics. Moreover, the basic facts on which the reasoning process starts may also often comprise such vague, linguistic articulations. In designing an expert system with fuzzy uncertainty / inexactness for a problem domain, one has to identity, first, the linguistic variables for the domain. Next, one has to define the term set of fuzzy sets which adequately covers the spaces of the domain. The members of a term set are linguistic terms characterising the corresponding linguistic variable. Once such fuzzy variables and term sets are defined, the knowledge representation using rules will become easy. For example, for the present problem domain (for a neonate), some main linguistic variables and the corresponding term sets may be identified as :

- General status → {Healthy, sick};
- Birth weight → {SGA, AGA, HGA};
- Muscle tone → {flaccid, some-flexion, actively moving the extremities};
- Heart rate → {none, normal(100-140), low};
- Respiratory effort → {none, slow/irregular, good/crying};
- Reflex stimulation → {noresponse; grimace; cries, coughs or sneezes};
- Colour → {blue/pale, periphery blue and body pink, pink}.

Every member of a term set will be attached to a set of numerical values between 0 and 1 (inclusive) called possibility values or grades of membership in the term set. This fixation of numerical values, obviously, will be done by domain experts. It is, now, important to examine the 'adequacy' of a term set for the problem domain. The question of granularity of representation comes into picture in this context. If we have too few members in a term set, a system may be inadequately descriptive. If we have too many members in a term set, this may lead to unmanageable situation in two important respects. First, large amount of storage space will be required for storing fuzzy tables. Or if one desires to represent such terms using mathematical functions, the number of such functions may be unmanageably high which may lead to reduction of speed of a typical expert system. Second, the associated rules will become cumbersome.

One feasible solution of the above problem may be achieved using the concept of '**hedges**' and fuzzy logical operators AND, OR, NOT. For example, if we are talking

about '**general status**' of a baby, the term set {**healthy, sick**,} may be considered as our primitive term set for the linguistic variable '**general status**'. Other fuzzy terms like **fairly healthy**, **very healthy**, **not very healthy** etc. can be derived from the members of primitive term set. Now, we are in a position to generate a fuzzy set to represent a complete phrase 'fairly healthy but not very healthy'. Fig.9.3 illustrates the required intermediate fuzzy sets to generate fairly healthy but not very healthy. This kind of analysis may be useful in assessment of a patient by a doctor. For example, a baby with respiratory distress or convulsion may be analysed in this manner for further management i.e. whether the baby needs hospitalization with or without intensive care, may be kept for observation or may be advised for domiciliary treatment.

The above concepts and ideas have been used in developing a prototype system for neonatal resuscitation management which has been considered in section 9.5. In the next section, we now present a fuzzy consultation system (Prototype 2.0) for the present problem domain based on Appendix A.

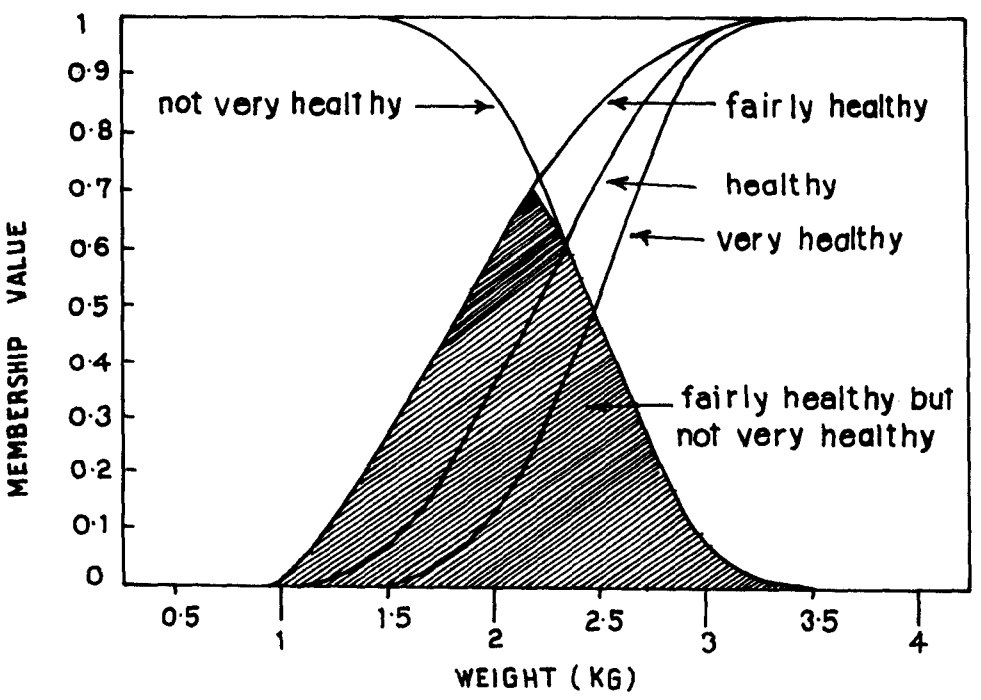


Fig. 9.3 Generating fairly healthy but not very healthy.

9.4. **Prototype 2.0**

This version deals with some fuzzy concepts in terms of linguistic articulations. The architectural components of our modified system is shown in fig. 9.4. The knowledge base (KB) consists of two parts : static part and dynamic part. The static part is relatively fixed over time. The dynamic part is capable of adding new facts or facts can be removed from the KB as when required. The inference engine uses LTKB and STKB to infer new facts. It has two well-known functions : inference and control. Backward reasoning process has been used here which favours the needs of the application domain. The inference engine uses depth-first scanning but with an 'improved back tracking' using some control rules provided that sufficient domain knowledge is available. A user interacts with the system with the user interface of the system. Through this module different queries are served by the system initiated by the inference engine. The fuzzification and defuzzification required for user supplied linguistic terms (fuzzy) and that required for fuzzy knowledge rules extracted from domain experts are governed by control section of the inference engine. The total review management is transparent to the user through this particular module. All accesses by a user to KB and review management module are through inference engine. However, logical access is presented through broken line of fig. 9.4.

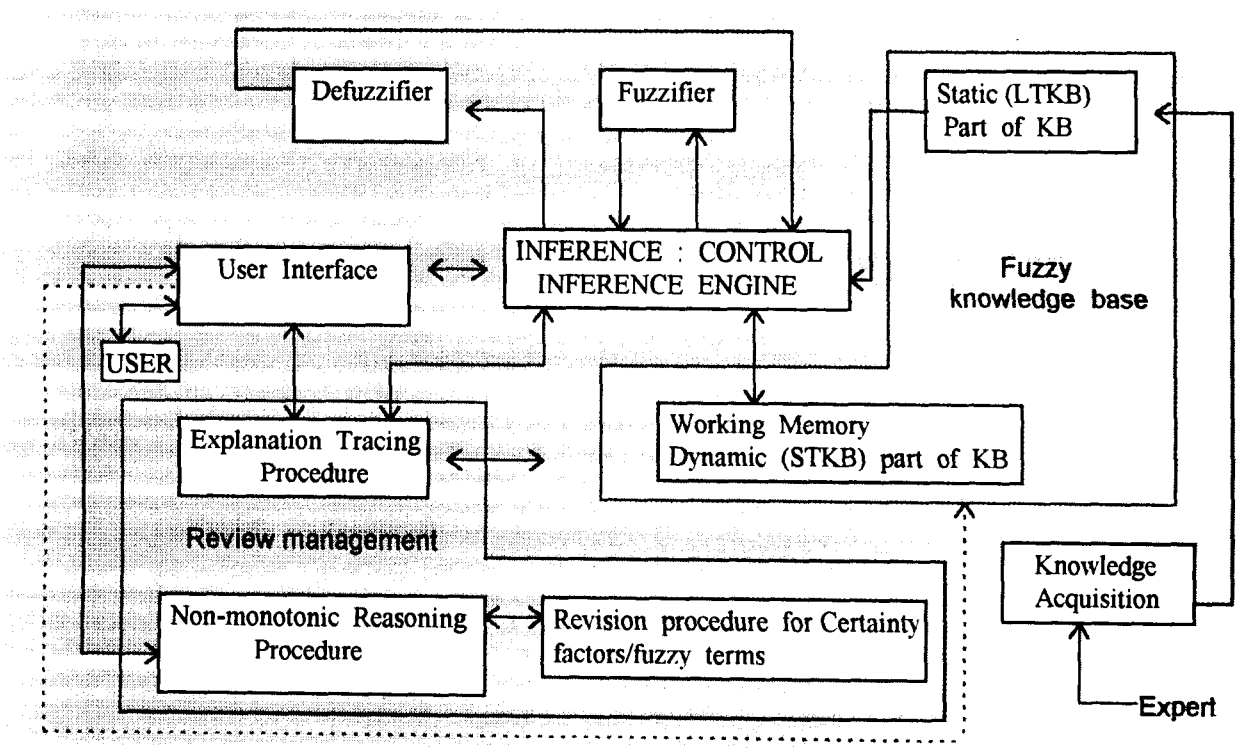


Fig. 9.4 System architecture

9.4.1. A typical consultation session

The following presents an excerpt from a typical consultation session with the system. The objective of this excerpt is to highlight some of the important features of the system. It is a menu-driven system.

A CONSULTATION SYSTEM FOR CHILD GROWTH AND DEVELOPMENT

MAIN MENU

Addition of records and consultation
Previous record with reference number
Printing record with reference number
Read records
Updation
Delete records
Dos
Exit Enter Command ☐

CAUTION : PRESENT PROTOTYPE NOT FOR MEDICAL USE

Considering the first menu, the particulars like name, age, parents' name etc about the child have to be supplied along with the values of different activities to be examined in terms of CF / fuzzy values. We get the following analysis :

Reference No.: c22/8/95basam43	Age : up to One Year
Activity	CF/Fuzzy Values On 2/8/95
Axial Muscle Tone	Good
Muscle Tone of Limbs	Very good
Spontaneous Gestures	Very good
Gripping	Good
Relation	Good
Emotional & Social Development	Almost poor
Language	Almost poor
Rhythms	Very good
EEG	Good

ABNORMAL ACTIVITIES
Emo_Social_Development
Language

Press a Key

SUMMARY

Total No. of Activities : 9

VERY GOOD : 3 GOOD : 4 ALRIGHT : 0 UNKNOWN : 0
ALMOST POOR : 2 POOR : 0 POOREST : 0

EXPERT CONCLUSION : GROWTH IS ALMOST NORMAL

Press a Key

If then compares with the previous values, if required, and then offers an advice as follows :

COMPARISON WITH PREVIOUS RECORD

Growth increasing in 80% cases.
The variation upto 10% has been neglected.

CONCLUSION :
The growth of the baby is expected to be normal

Press a Key

ADVICE

Please take care of the following activities of Basudev

Axial Muscle Tone
Emotional & Social Development
Language

Press a key

9.5. Neonatal Resuscitation Management : An Application (Prototype 3.0)

9.5.1. System analysis

The goals of neonatal resuscitation are to prevent the morbidity and mortality associated with hypoxic-ischemic tissue (brain, heart, kidney) injury and to re-establish adequate spontaneous respiration and cardiac output [14].

Different levels of resuscitation are required depending on the signs and symptoms of a newborn as observed by a medical practitioner. In general, **APGAR**-score [14] is used for resuscitation management. This scoring system uses five main components such as **muscle tone of limbs**, **heart rate**, **respiratory effort**, **reflex stimulation** and **colour**. Although doctors are used to dealing with precise numeric data in respect of some factors, for example, heart rate, there is neverthless considerable uncertainty with these factors. Much of the knowledge as gathered by doctors may have shades of meaning(fuzzy). For example, heart rate, say 95/min may belong to two regions but with different possibility values as depicted in fig. 9.5.

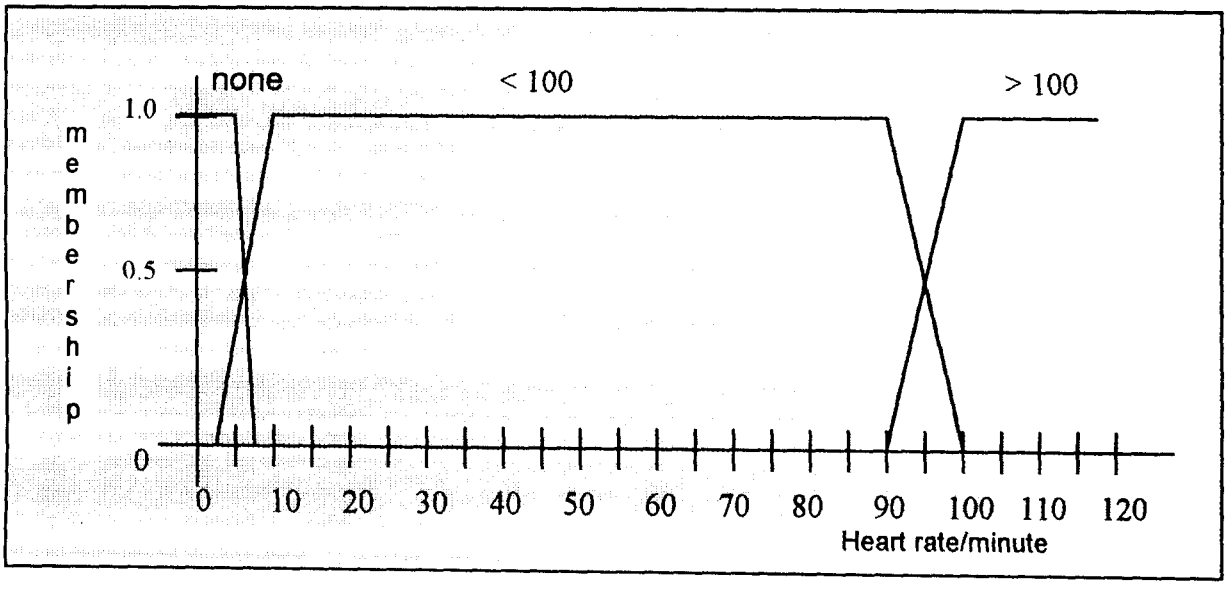


Fig. 9.5 Heart rate

This heart rate 95 per minute might not sharply be defined in strictly one region. For another example, let us consider muscle tone of limbs. In general, flexion is observed at wrist, elbow, shoulder, ankle, knee and hip. It is sometimes difficult to define the value as flaccid or some-flexion or active as depicted in fig. 9.6.

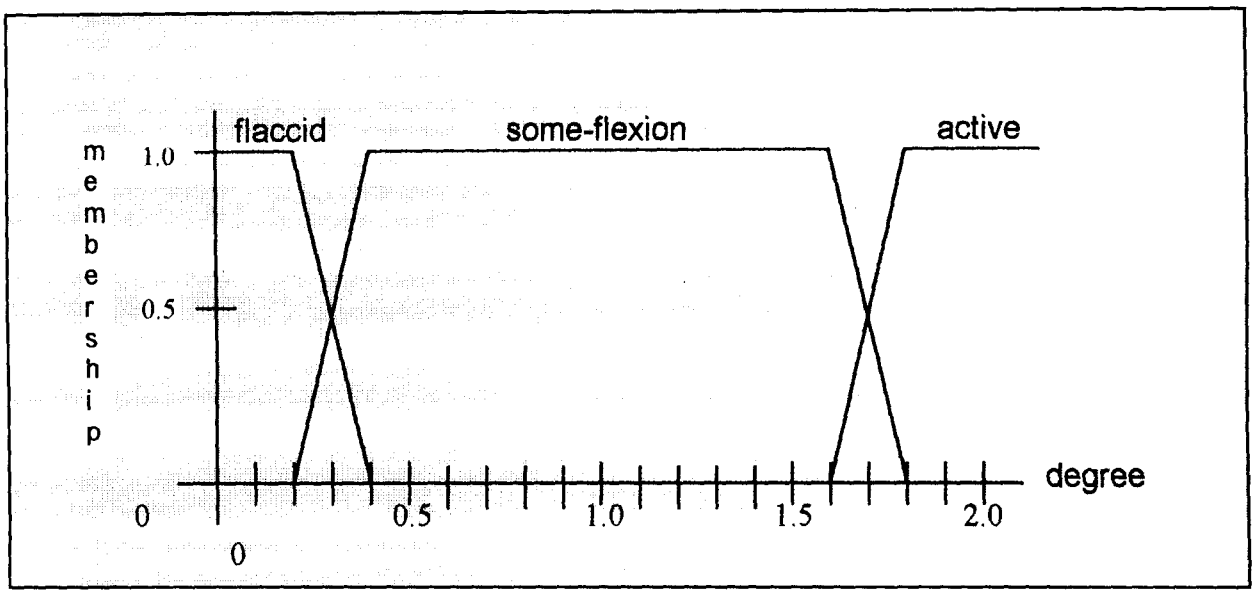


Fig. 9.6 Muscle tone of limbs

Once such fuzzy variables and term sets are defined, the knowledge representation using rules will become easy. For the present problem domain (for a neonate), main linguistic variables and the corresponding term sets may be identified as :

- Muscle tone → {flaccid, some-flexion, actively moving the extremities};
- Heart rate → {none, normal(100-140), low};
- Respiratory effort → {none, slow/irregular, good/crying};
- Reflex stimulation → {noresponse; grimace; cries, coughs or sneezes};
- Colour → {blue/pale, periphery blue and body pink, pink}.

Every member of a term set will be attached to a set of numerical values between 0 and 1(inclusive) called possibility values or grades of membership in the term set. This fixation of numerical values, obviously,was done by a domain expert.

9.5.2. **Fuzzification of system state input variables**

In this model, **muscle tone of limbs**, **heart rate**, **respiratory effort**, **reflex stimulation** and **colour** are treated as the state fuzzy variables. Fuzzification of variables lies under the trade-off between precision in resuscitation decision and computation time. Each of the system state fuzzy variables is decomposed into a reasonable number of fuzzy regions following the rules of thumb [15]; that is, an odd number of labels associated

with a variable had been chosen. Each label should overlap somewhat between 10% and 50% with its neighbours. The five system input state variables were fuzzified e.g. fig. 9.5 and 9.6 where fuzzification of heart rate and muscle tone of limbs are shown respectively. Fig. 9.7, 9.8 and 9.9 show the fuzzification of respiratory effort, reflex stimulation and colour respectively.

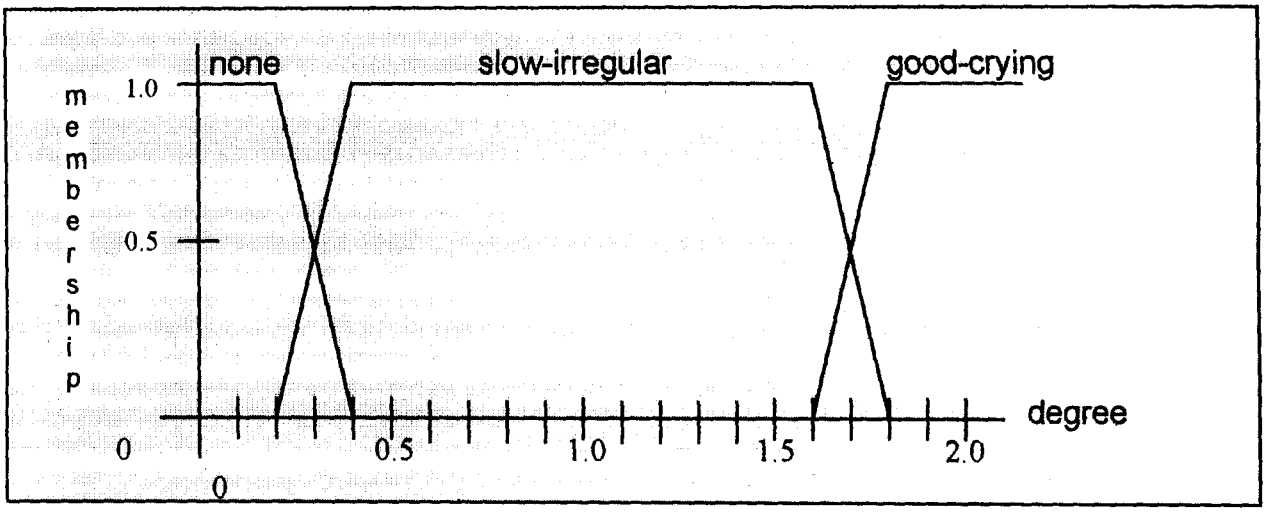


Fig.9.7. Respiratory effort

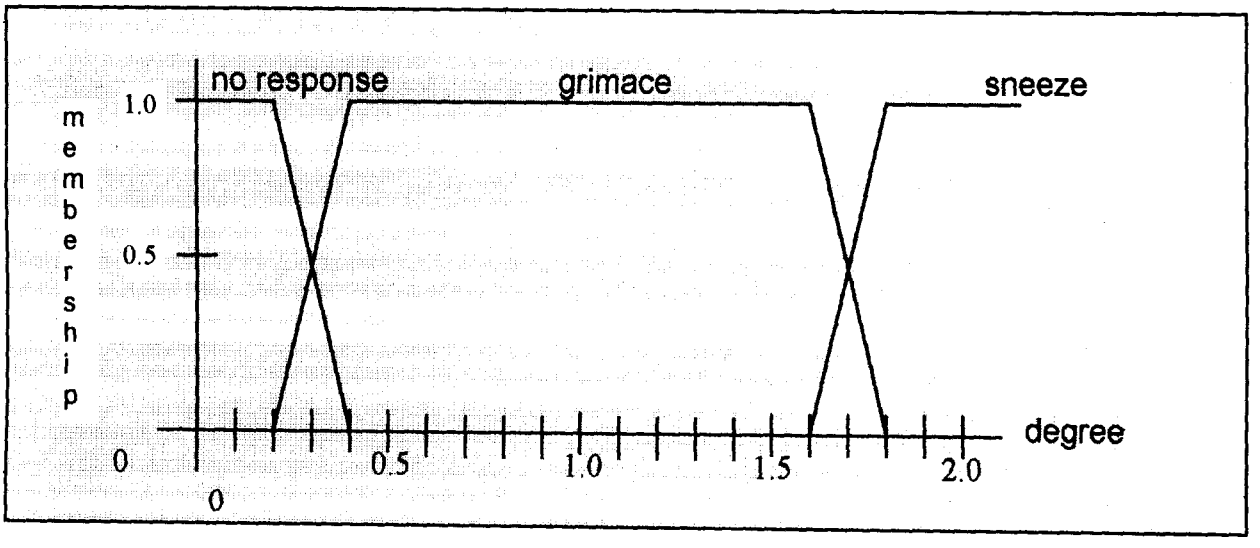


Fig.9.8. Reflex stimulation

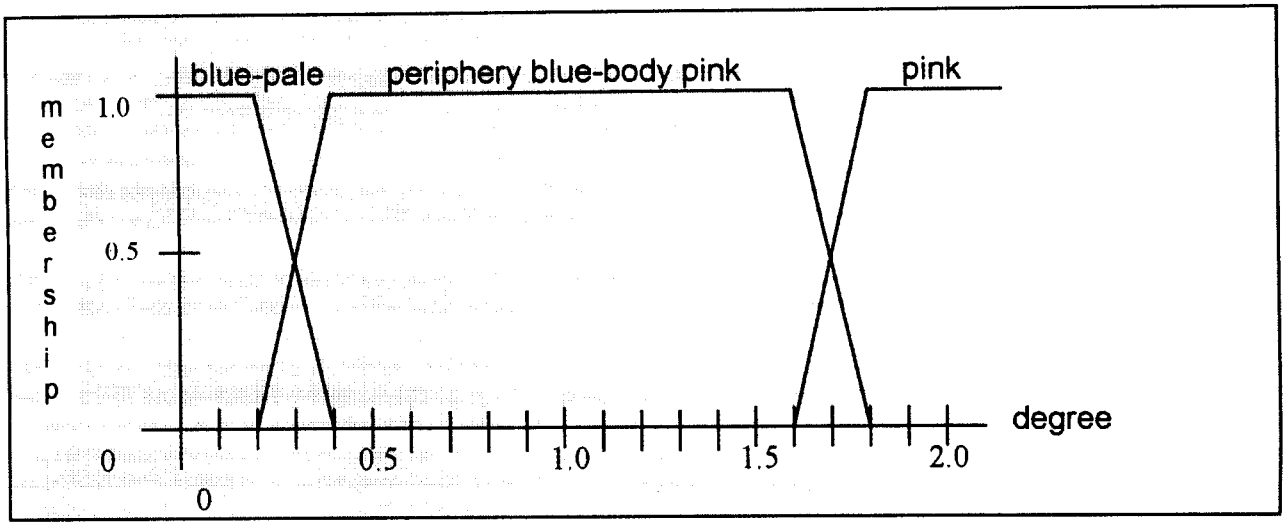


Fig. 9.9. Colour

9.5.3. **Fuzzification of system state output variable**

The output of the designed fuzzy model is the decision like type I or type II or type III, a neonate has to be undergone. The type I baby should be undergone an immediate resuscitation; Intra-tracheal incubation and Sodium-bi-Carbonate with I V Dextrose solution. For type II baby some resuscitation is needed; e. g. gentle patting at the back; throat suction; Sodium-bi-Carbonate and I V Dextrose solution. And Type III baby should need no resuscitation; only tactile stimulation is sufficient. The system output variable had been fuzzified as shown in fig. 9.10.

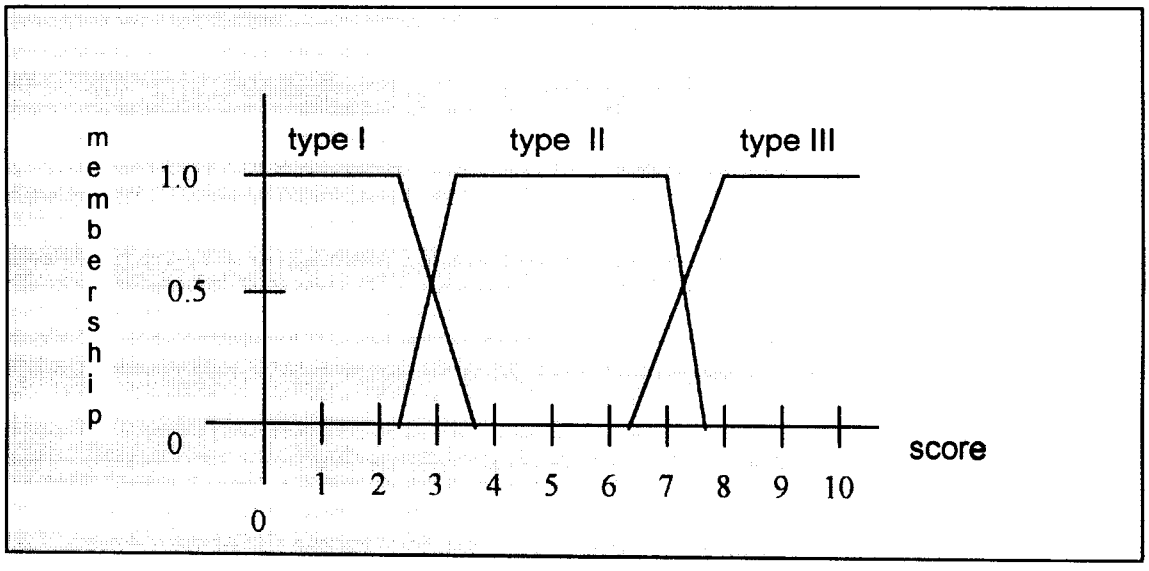


Fig. 9.10. Fuzzification of system output variable : resuscitation

9.5.4. Inferencing process

Since fuzzy sets are defined for each system state variable, a fact supplied by the user will have a membership for each fuzzy set. As the fuzzy sets in the system are defined by arbitrary {value, membership} pairs, a particular fact may require interpolation between defined pairs. Some of the membership grades so determined might not be appropriate due to their small numeric value. In order to eliminate undesired effects, a fuzzification threshold is introduced, in this case 0.2, in the line of [11,16]. Only calculated membership ≥ 0.2 are used in the process of inferencing.

We concentrate on fuzzy rule based approach :

If < fuzzy proposition >, then < fuzzy proposition >,

where the fuzzy propositions are of the form, "p is Q" or "p is not Q", p being a scalar variable and Q being a fuzzy set associated with that variable. Generally, the number of rules is related to the number of system state input variables. For our case, we have five such variables each of which is divided into three fuzzy regions leading to a total of 243 fuzzy rules comprising the fuzzy knowledge base. Knowledge acquisition was done from a domain expert employing interview techniques. Architectural components of the system are shown in fig. 9.4.

A total of 32 combinations as input are used for knowledge base scanning. Different rules may be activated at the same time and combination of their outputs is then defuzzified to compute the resuscitation status.

Fig. 9.11 shows the inferencing and defuzzification process due to multiple rules. The 'maximum' method of defuzzification has been applied in the model [17]. This simply means that from among the fuzzy rules for the input set, the one with the largest

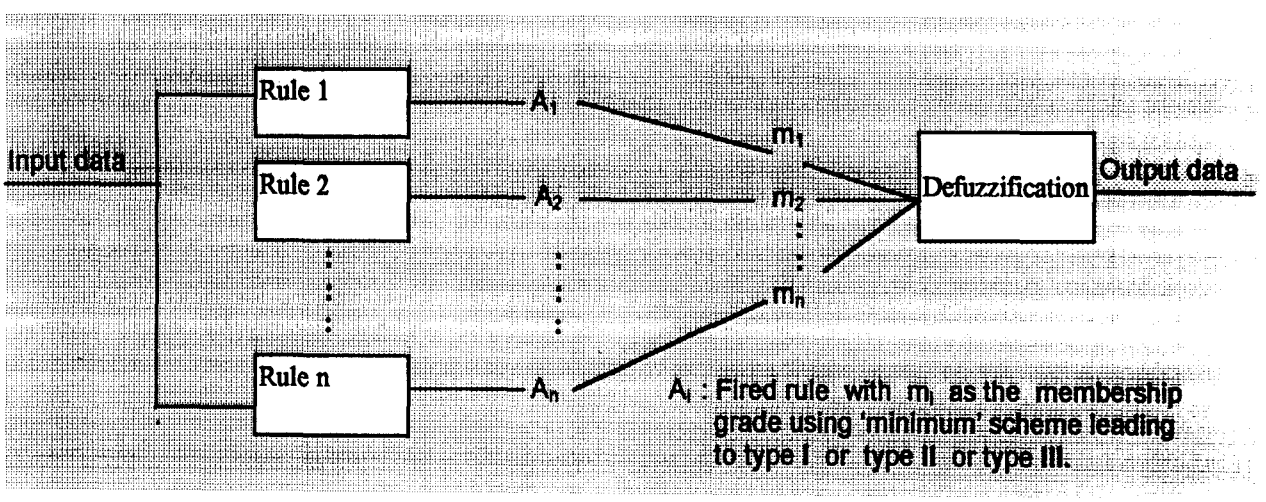


Fig. 9.11. Inferencing and defuzzification

membership grade is used to interpret the conclusion. There can be a number of conclusions according to the knowledge base, input values and defuzzification procedure and so, for presentation to the users, they are ordered by the 'degree of truth'. Thus the results of the defuzzification procedure is a list of suggestions sensible for the resuscitation procedures. The system has been developed using Turbo-prolog running under MS-DOS.The static and dynamic knowledge base features of Turbo-prolog facilitates the implementation. It is a compiled language providing a better runtime response. Prolog is amenable to problems that treats objects and their relationships. For this reason some people prefer Prolog as one object-oriented language [18]. One can easily identify the rules(objects) fired. In addition, the o-o nature of Prolog enables it to be applied within the context of the prototyping paradigm for software engineering.

The power of a fuzzy system lies in the fact that a fuzzy rule can replace many conventional rules. The power of this system lies in the ability to deal with crisp as well as non-crisp values (fuzzy) of input. Moreover, the fuzzy inference process allows input to be inaccurate to some degree or missing but still produces sensible results in contradiction to traditional knowledge based systems where output heavily depends on absolutely contradiction free knowledge.

9.5.5. **Performance Evaluation**

Lastly, performance evaluation should be produced. As a matter of fact, it is to some extent difficult to establish the degree of performance owing to the non-availability of experimental results in the classical sense. As a preliminary test of the system's performance, we put here the following example data as input using the fuzzy knowledge base :

(A) Crisp values (input)

Muscle tone of limbs = 0.2;
Heart rate per minute = 80;
Respiratory effort = 0.7;
Reflex stimulation = 1.7;
Colour = 1.7

Decision (output) : The patient probably has to be undergone with type II resuscitation with the membership 0.5.

$$[(\min(m_{MTL}(0.2), m_{HRT}(80), m_{RES}(0.7), m_{RST}(1.7), m_{COL}(1.7)$$
$$= \min(1.0, 1.0, 1.0, 0.5, 0.5)$$
$$= 0.5]$$

(B) Combinations of fuzzy and crisp values (input)

Muscle tone of limbs = 0.2;
Heart rate per minute = low;
Respiratory effort = 0.7;
Reflex stimulation = 1.7;
Colour = periphery-blue-body-pink.

Decision (output) : The patient probably has to be undergone with type II resuscitation with the membership 0.5.

(C) Fuzzy values (input)

Muscle tone of limbs = flaccid;
Heart rate per minute = low;
Respiratory effort = slow_irregular;
Reflex stimulation = grimace;
Colour = periphery-blue-body-pink.

Decision (output) : The patient probably has to be undergone with type II resuscitation with the membership 1.0.

The expert confirmed that the decision suggested by the system were reasonable for the given input data and knowledge base. However, more realistic case studies should be produced to validate the system more accurately.

In order to validate the system more accurately, the results of **twenty-one** case studies are now produced. The results suggested by a domain expert were compared with those suggested by our system. The system has an accuracy of **95%**.

9.6. **Conclusions**

In this chapter, we have presented an outline of fuzzy concepts in paediatric problem domain to show the usefulness and importance of fuzzy logic and fuzzy set theory keeping in mind the importance of experience and judgement that an expert uses when examining a patient. In designing computer-based expert systems, one of the key problems is to manage inexact knowledge of different kinds. One important kind is fuzziness arising from human linguistic articulations. It is argued that fuzzy concepts have to be dealt with properly to offer a rational decision. After a brief introduction to fuzzy logic and fuzzy set theory, we have identified different primary linguistic variables and corresponding term sets which were required during the design of a fuzzy, knowledge based neonatal resuscitation management system. Some

potential problems, e.g., the problem of 'adequacy' of a term set, have also been addressed in this context. The fuzzy system produces sensible results for both the cases of the nature of input values (fuzzy or crisp and / or a combination of fuzzy and crisp values). The performance was further evaluated using some practical case studies available from two domain experts engaged in Hospital and / or Nursing homes. We are re-affirmed with the belief that the decision making in this particular domain is suitable for modelling with fuzzy logic. More case studies are planned in order to validate the system further.

Finally, fuzzy logic offers a natural and convenient way of managing inexactness expressed in the form of vagueness. The naturalness of fuzzy logic may certainly assist us in both knowledge acquisition and inferencing procedures. Due to the naturalness of input - such as low, normal, slow-irregular, the programs for inferencing are generally much smaller and faster than conventional programs using binary logic.

References

1. K. P. Adlassnig. A fuzzy logical model of computer-assisted medical diagnosis. Methods of Information in Medicine; vol.19, 1980, 141-148.

2. K. P. Adlassnig. Fuzzy set theory in medical diagnosis. IEEE Trans. Systems, Man and Cybernatics; vol. 16,1986, 260-265.

3. K. P. Adlassnig and G. Kolarz. Representation and semiautomatic acquisition of medical knowledge in Cadiag-1 and Cadiag-2. Computers and Bio-medical Research; vol. 19,1986, 63-79.

4. K. P. Adlassnig. Uniform representation of vagueness and imprecision in patient's medical findings using fuzzy sets. Proc. Cybernatics and systems '88; Kluwer Academic, Dordecht, 1988, 685-692.

5. E. Binaghi. A fuzzy logic inference model for a rule-based system in medical diagnosis. Expert Systems; vol. 7, no. 3,1990, 134-141.

6. K. J. Cios, I. Shin and L. S. Goodenday. Using fuzzy sets to diagnose coronary artery stenosis, IEEE Computer; March, 1991, 57-63.

7. G. J. Klir and T. A. Folger. Fuzzy sets, uncertainty and information. Prentice-Hall, Inc.; Englewood Cliffs, N. J., 1988, 246-254.

8. K. Ong and Qiu-He Wang. A generalised fuzzy reasoning algorithm for an object-oriented expert systems tool. Expert Systems; vol.12, no.3, 1995,199-206.

9. Q. Wang. Computer aided differential diagnosis for common convulsions in children. Chinese Physics Medicine; vol.5, no.4, 1983, 219-223.

10. L. A. Zadeh. Fuzzy sets. Information and Control; vol.8, 1965, 338-353.

11. C. P. Schründer, J. E. Galletly and J. R. Bicheno. A fuzzy Knowledge-based decision support tool for production operations management. Expert Systems; vol.11, no. 1, 1994, 3-11.

12. M. Mizumoto, S. Fukami and K. Tanaka. Some methods of fuzzy reasoning. Advances in fuzzy set theory and Applications. M. M. Gupta, R. K. Ragade and R. R. Yager (eds.); North-Holland, Amsterdam, 1979, 117-136.

13. L. A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans. Systems, Man and Cybernatics; SMC-3, 1973, 28-44.

14. W. E. Nelson., Text Book of Pediatrics, 15th edition, Book1, W. B. Saunders Company, Bangalore, 1996.

15. E. Cox. Fuzzy Fundamentals. IEEE Spectrum; October, 1992, 58-61.

16. E. H. Shortliffe and B.G. Buchanan. A Mathematical model inexact reasoning in medicine. Math. Biosci; Vol. 23, pp. 351-379, 1975.

17. I. Graham. and P. L .Jones. Expert Systems : Knowledge, Uncertainty and decision. Chapman and Hall, 1988.

18. R. S. Pressman. Software Engineering - A practitioner's approach. 3rd edn., McGraw Hill; New York, 1992.

KID : AN INTEGRATED SYSTEM (PROTOTYPE 4.0)[†]

10.1. Introduction

As a matter of fact, the domain knowledge is vast and varied. For the unfolding of intimate areas of the knowledge, it was required to develop demonstration prototypes incorporating different issues using Turbo Prolog as a fast prototyping tool. These prototypes also helped understanding the requirements of domain users soliciting the development of an integrated, menu-driven user-friendly system. This chapter is meant for presenting such a system devloped named 'KID'. The system KID is totally object-oriented starting from the analysis to implementation.

In the next section, we shall present the analysis and design of the system. Section 10.3 contains the graphical user interface (GUI) issue needed for the domain. Section 10.4 deals with the implementation issue. In section 10.5, we have presented an excerpt of a consultation session with the system. In the section 10.6, we have considered the validation issue. At last, our conclusions and discussions are summarised.

10.2. Analysis and Design

The total paediatric range is divided into the following stages : foetal (conception - birth), neonates (birth - 4 weeks), infants (4 weeks - 1 year), toddler (1 year - 3 years), pre-school going (3 years - 5 years), school going (5 years - 10 years) and adolescence (10 years - 18 years); and hence it was decided to have seven modules of the system. These seven sub-domains were again managed with pull-down menus. The following typical steps are followed by doctors investigating a baby except 'foetal' module:

General identification and preliminary complaints (Phase I)

↓

General / Physical examination (Phase II)

↓

Complaints oriented examinations (Phase III)

↓

Differential diagnosis and advice

↓

Related investigations (Phase IV)

↓

Final diagnosis and treatment

[†] This is based on the publication [Proc. of the 3rd. Int. conf. on Modelling and Simulation. Victoria University of Technology, Melbourne, Australia, 29-31 Oct' 97, 151-156] of the author.

In Phase I, a doctor considers the general informations like name, age, sex, weight at birth etc along with the preliminary complaints from parents/guardians. Some of these informations may pilot the actions to be taken in later phases of investigations. In Phase II, usually, the examinations are done using the following four steps in order :

i) Inspection;
ii) Palpation;
iii) Percussion; and
iv) Auscultation.

The examination steps should be followed in every case so that it should be methodological. But, however, examination steps may have to be altered in some situations. For example, very often Auscultation is done before Palpation and Percussion when the baby is quite and co-operative before he turns hostile owing to various examination procedures. Phase III considers the complaints oriented examinations as well as it considers the knowledge relevant with complaints and symptoms. The successful end of this phase leads to differential diagnosis and advice. For final diagnosis and treatment, we consider Phase IV where related investigations are done. After having the reports of laboratory investigations, the system suggests the final diagnosis and treatment.

We consider each phase as a distinct class. A class represents general properties and structure of a group of objects. Each object has its associated attributes. For example, the class 'Physical Examinations' (**Appendix B**) has the typical objects like heart, abdomen, ear, eye, liver etc. A typical object, say ear, has the attributes like 'malformed pinna' or 'preauricular sinus'. Objects manifest themselves in one of the ways represented in fig.10.1[1].

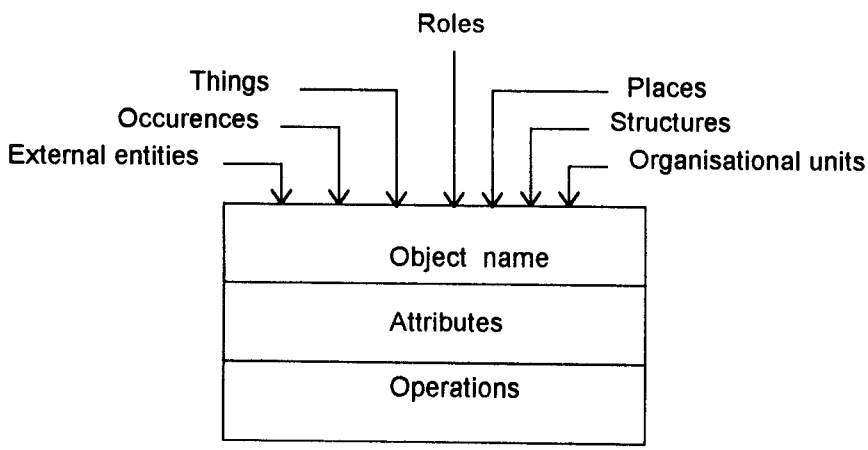


Fig.10.1 Objects

We have identified potential objects which are nothing but extraction of nouns. Coad and Yourdon [2] suggest following six selection characteristics for actual inclusion of objects in a project which are Retained information, Needed services, Multiple attributes, Common attributes, Common operations, and Essential requirements. They suggest that a potential object should satisfy all (or almost all) of the above characteristics for the legitimate inclusion of an object. In addition, we have identified different 'Rule Groups' for the identification of the cause(s) of a disease. For example, the cause of Jaundice in neonate may be Rh-Incompatibility or Septicemia or Neonatal Hepatitis or others. The Rule Group : Septicemia Diagnosis is as follows :

Rule Group : Septicemia Diagnosis(In a neonate presents with jaundice)

Rule 1

IF Appearence Day IS Third AND Lethergy AND Refused To Suck AND
Reflex IS Poor AND DIC & Shock AND Necrotising Enterocolitis
THEN Septicemia Diagnosis (CF = 85)

Rule 2

IF Appearence Day IS Fifth AND Lethergy AND Refused To Suck AND
Cry IS Poor AND Reflex IS Poor AND DIC & Shock
THEN Septicemia Diagnosis (CF = 70)

.
.
.

Rule n

IF Appearence Day IS Tenth AND Lethergy AND Refused To Suck AND
Cry IS Poor AND Reflex IS Poor AND Sclerema
THEN Septicemia Diagnosis (CF = 65)

Moreover, the links between different phases have been identified. For example, the attribute, 'Weight At Birth' of class 'Identification' (Phase I) will be useful in Phase III and Phase IV for suggesting differential diagnosis and advice; and final diagnosis and treatment. Currently, the system consists of 452 rules.

The structures and the number of items of each pull-down menus should certainly depend on the complexity of each module. For example, the foetal module may have the options for:

1) Prediction of gestational age from foetal parameters (sonograhically measured);
2) Prediction of foetal body weight from foetal parameters (sonograhically measured);
3) Determination of IUGR(Intra Uterine Growth Retardation).

Moreover, the issue of zonal variation has been considered in the above options. The neonatal module, as per present prototype, can fulfill the following objectives :

1) To find the general status of a neonate;
2) To find out the gestational age of neonates by date and/or examination;
3) To find out any problem related to its growth and development;
4) Complications due to prematurity and/or Low birth weight (LBW) and management;
5) Guideline of routine examination of neonates at delivery room;
6) Guidelines of management of new born at delivery room;
7) Common neonatal diseases affecting growth and development;
8) Guidelines of management of neonates requiring Level I and Level II care;
9) Guideline for directing the neonates for Level III care;
10) Finding the cause(s) of Jaundice and management.

The other modules are also planned as per the requirements of the sub-domains. For all these modules from foetal to adolescence, 'Phase I - Phase IV' scheme has been incorporated.

10.3. **Graphical User Interface (GUI)**

The better acceptance and popularly of a software depend not only on the internal technological content of the software but on the user interface to a large extent. A "user friendly" system is welcome. An interface may not satisfy all classes of users; this depends on the user's level of education, experience and expertise. It may be a formidable task to design a system with various levels of interface to satisfy all classes of users. But, there may have some common approaches of interface design along with some specific strategies for that particular domain and intended users. So, user interface design has as much to do with the study of people as it does with technology issues [1]. So, one has to fix up primarily the following : (i) Who will be the users ? (ii) How will they use it ? (iii) How will they learn to interact with the new computer-based system ? (iv) What will the user expect of the system ? (v) What sort of difficulties user might face and how to overcome those difficulties ? (vi) How the question-answer sequence is tailored ? Answering all these questions basically depends on the analysis of the problem domain and the requirements of the intended users (e. g. doctors or para-medical staffs etc. for medical domain). However, if a designer specifies a human-computer interface (HCI) that makes undue demands on STM(short-term memory) and / or LTM (long-term memory) of users, the performance of the human element of the system should degrade.

Four different models come into play when an HCI has to be designed : (i) design model, (ii) user model, (iii) system perception, and (iv) system image. A design model of the entire system incorporates data, architectural and procedural representations of the

software. The user model depicts the profile of the end users of the system. To build an effective user interface, "all design should begin with an understanding of the intended users, including profiles of their age, sex, physical abilities, education, cultural or ethnic background, motivation, goals and personality" [3]. The system perception is the image of the system that an end user carries in his or her head. The system image combines the outward manifestation of the computer-based system with all the supporting information that describes system syntax and semantics. Each of these models may differ significantly. Reconciliation of these differences should be taken into consideration during the interface design. Users generally feel comfortable with the use of a system when the system image and system perception are coincident. For the purpose, the design model must have been developed to accommodate the information contained in the user model and the system image must accurately reflect syntactic and semantic information about the interface. In connection with the good coincidence of the system image and system perception, graphical user interface (GUI) methodologies might be a better choice. Moreover, GUIs are widely used for interaction with computers because of their transparencies, ease of use, user friendliness and robustness as they provide the user to interact with the system by manipulating the graphical objects [4,5]. For a diagnostic system where sign and symptom based input is necessary, graphical objects are highly suitable. In such domains, command driven, menu driven or simple textual user interface (TUI) might not be suitable simply because of the non-availability of displaying graphic objects / images. Medical diagnostics is such an application domain where GUIs are highly suitable for better decision making. For example, the Dubowitz [6] method of determining menstrual / gestational age of a new born uses different postures of the baby as depicted in fig. 10.2.

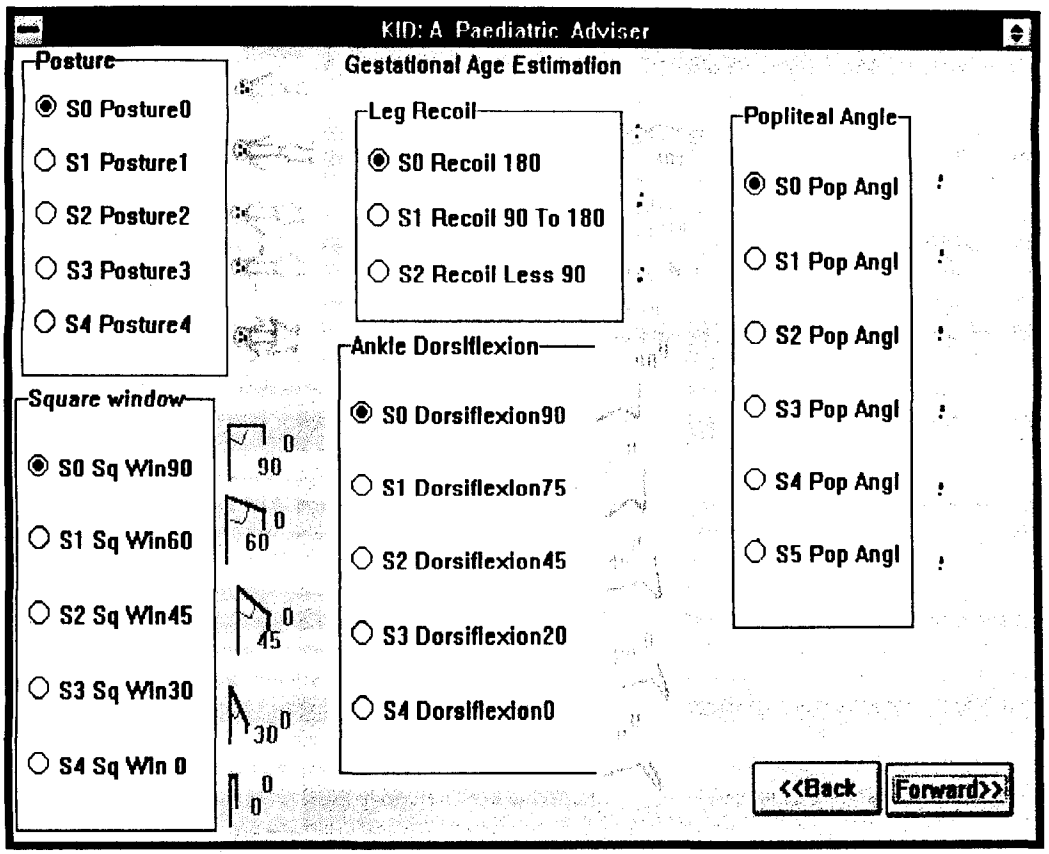


Fig. 10.2. Postures of neonates.

Our system provides GUI with the following characteristics :

(i) Having facility to select/deselect multiple options from a specified item-screen (e.g.,skin, fig.10.3).
(ii) Ability to take inputs at per user perceptions.
(iii) Ability to display pictures/images and texts as well as a combination of these.
(iv) Real-world situations are simulated through the system. For example, the 'postures' of a neonate (fig. 10.2) can be fed to the system as input as a doctor visualises a neonate.
(v) Lastly, user friendliness i.e. a menu-driven, GUI- based system needs no such rigorous training for its use.

10.4. **Implementation**

As a developmental tool one can use one of three : AI-languages, tool kits and shells. The selection of an appropriate tool should depend on the characteristics of the problem domain as well as on the architectural approaches with varying functionality and performance of candidate tools.

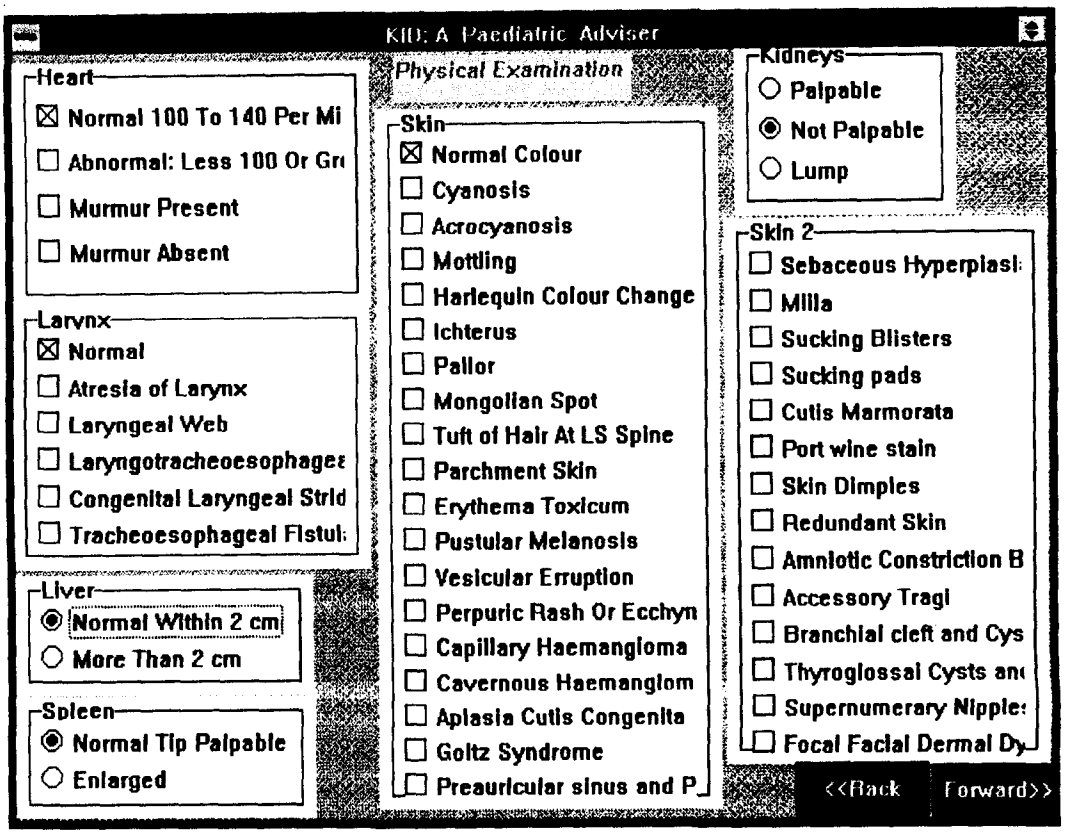


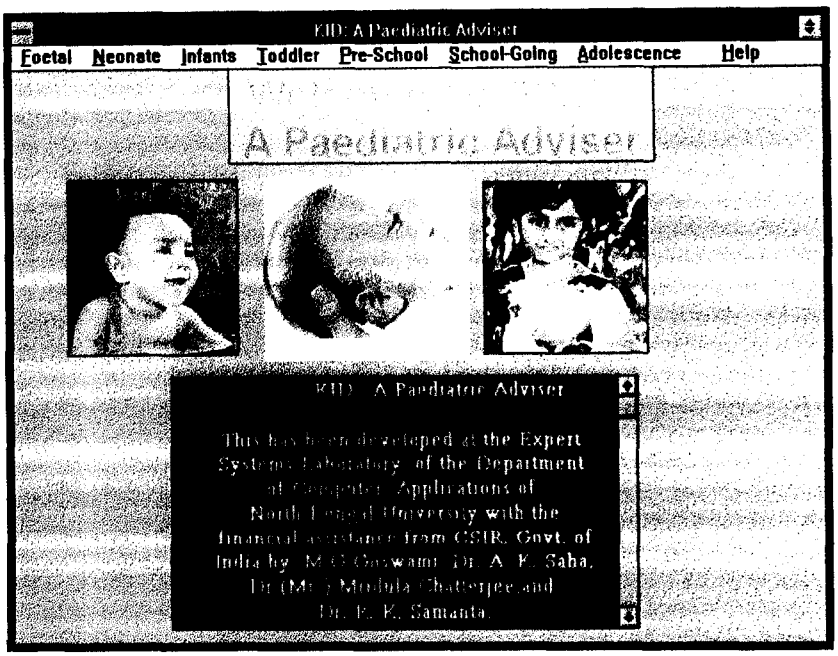
Fig. 10.3. Physical examination input screen.

This selection process should also be governed by whether the goal of the system development is to develop a system for actual use or to make major advances in the state-of-the-art of ES-technology. The goal of this system KID is to use it for practical situations. For the purpose, a large, hybrid object-oriented toolkit should be more appropriate as the problem is now well understood and formulated.

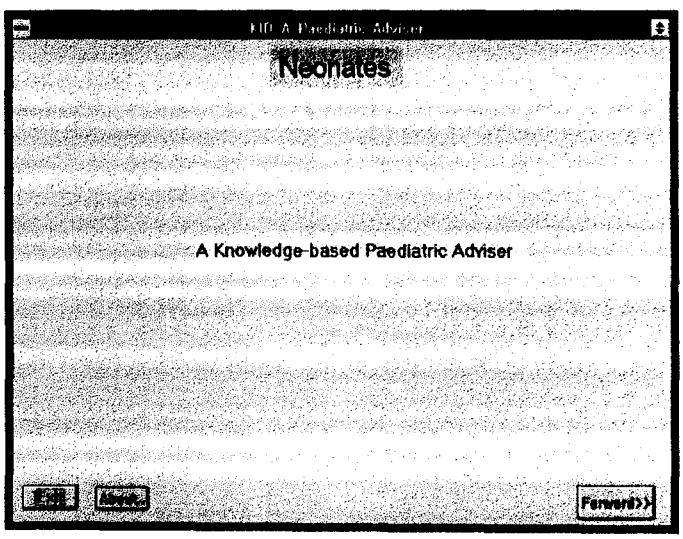
As an implementation tool, we have used Level5 Object running on PC and under Windows environment. Level5 is totally object-oriented starting with the design of the software itself. The editor, visual windowing system, displays, database interfaces, inference engines, knowledge bases, devices, files and timers are all built-in objects [7]. Level5 packages these objects as system classes that contain a rich array of built-in-logic and object tools, which give the developer ultimate control over application design and function. But, however, it can manage inexactness of uncertainty type only.

10.5. A Consultation Session : An Excerpt

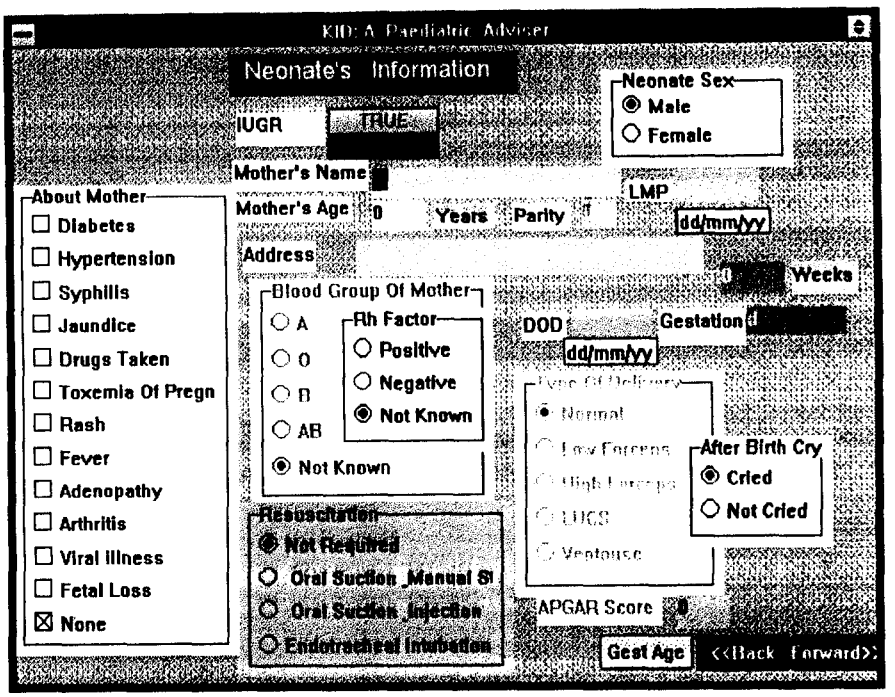
Let us now present an excerpt of a typical consultation session with the KID - expert. The following is the opening screen of the system :



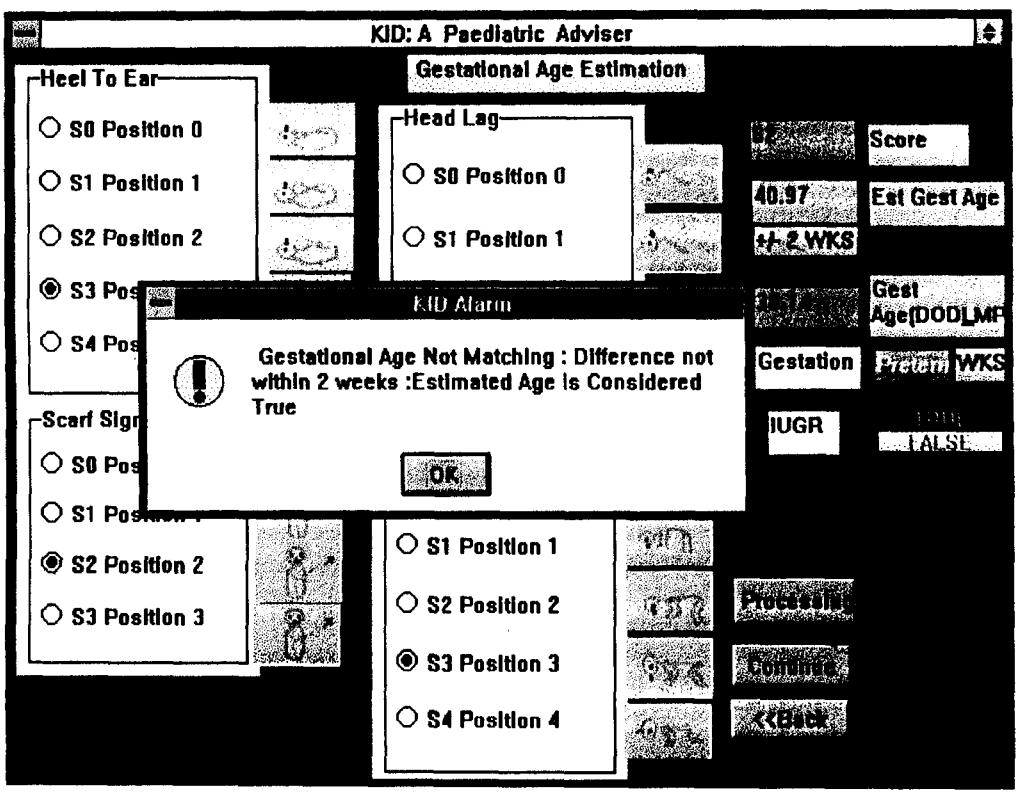
Using different menus and pull-down menus one can branch to different units of diagnosis and management. Let us select **Neonate ➪ new case** as the following screen appears :



Click 'forward' you will get next :



Concentrating on the determination of menstrual / gestational age of a neonate using Dubowitz [6] method and comparing it with the LMP - DOB method we get the following output screen.



10.7. **Conclusions** and **Discussion**

In this chapter, an attempt has been made to present an integrated expert system (KID) within the confines of the state-of-the-art of ES-technology. Different developmental issues have been discussed. The findings of this system are compared with the findings of a domain expert. At per our initial observations, the agreement is satisfactory. But, for a system handling human life, more scrutiny is required before using it as a production/commercial system. More issues need to be added into the system to make it more useful. Our future efforts will obviously be towards such achievements.

References

1. R. S. Pressman. Software Engineering - A Practitioner's Approach. 3rd ed., McGraw-Hill, Inc; 1992.

2. P. Coad and E. Yourdon. Object-Oriented Analysis. Prentice-Hall; 1990.

3. B. Shneiderman. Designing the user Interface. Addison-Wesley; 1987.

4. G. Lee. GUI application development. Prentice-Hall; 1993.

5. D. Collins. Designing object-oriented user interfaces. Benjamin Cummings Pub. Co.; 1995.

6. L. Dubowitz and V. Dubowitz. Gestational Age of the Newborn. Addison-Wesley; 1977.

7. Level5 Object for Microsoft Windows Reference Guide (Rel 3.0). Information Builders; New York, USA; 1993.

A CASE BASED APPROACH (PROTOTYPE 5.0)[†]

11.1. Introduction

Although, for more than 30 years the rule-based reasoning (RBR) has been the predominant technology for KBS / ES projects in different domains, but, however, this approach is critisized for: (i) It is sometimes difficult to translate the knowledge of a domain into rules; (ii) Managing thousands of rules for a practical system design seems very difficult since (a) the system would be very complex and time consuming to develop, (b) they are difficult to maintain once developed [1], (c) it is very difficult to add/subtract rules as well as to pinpoint the effects of a rule within such a complex system as complex debugging procedures are required; (iii) The time required, with this volume of rules, to reach a conclusion seems untolerably high; (iv) It is very hard to find out the contradictory rules for the system. Although, however, some improved techniques, e.g., object-oriented technology, have been proposed for knowledge representation and management to ease the debugging process of knowledge but knowledge acquisition, the primary task of KBS / ES development, still remains to be a complicated task.

Case based reasoning (CBR), a graceful alternative of rule based approach, is gaining swift momentum as a relatively new technology. CBR is a problem-solving paradigm using past experiences to guide problem-solving [2]. In CBR, cases similar to the current problem are retrieved from the case memory, the best match is selected from those retrieved, and if an adaptation is necessary then the best match will be adapted to fit the current problem based on the differences between the previous and current cases. The issue of knowledge acquisition needs to be revisited here. Where do we start knowledge acquisition? As argued by di Piazza and Helsabeck [3], the extraction of cases seems to be the most effective way to begin knowledge acquisition for any KBS / ES project because the case description is the easiest way for the experts to express him(her) self. Obviously, the previous cases must be available to the experts. It is usual in medical domain that the doctors observe cases, diagnose and advice what is their daily fare. So, case-based reasoning might be a good choice. Moreover, CBR does not require an explicit domain model and so knowledge elicitation would be performed by gathering case histories [4]. From case histories rule generation might be possible describing the system.

[†] This is based on the publication [Proc. of the Nat. conf. on Medical Informatics, Institute of Public Enterprise, Hyderabad, India, 6-7 April, 1999, 99 - 107] of the author.

Two problems can be encountered when one relying exclusively on case-based reasoning [5] : (i) it is difficult to start from the very begining with an empty case library; and (ii) after some time the case library can become huge and contains much redundancy. Therefore, it is better to combine the case-based reasoning with some other paradigm to compensate for these marginal insufficiencies. As a potential compensating candidate, the rule-based reasoning component can play useful role with the help of system knowledge and heuristics. This essentially means that the rule-based part can offer a solution as and when requested. The following highlights the circumstances when one has to request for rule-based approach:

- No old solution can be found for a current situation in the case library. For this event, the rule-based reasoning part can be activated. But, however, such RBR generated solution should be checked more carefully.

- There might be some situations which are almost same but not identical. Such cases cause a high level of redundancy of the case library. Replacement of such a class of very similar cases by a set of rules can partially solve this problem.

- The rule-based reasoning part can also assist in solving some specific situations.

11.2. **Neonatal Resuscitation Management : A case study**

CBR has already been applied in different domains such as Engineering, Medicine, Design, Law, Business planning [6-11]. But, however, to our knowlegde, CBR has not been applied to the problem domain of neonatal resuscitation management. This work suggests a case-based approach with rule-based augmentation for developing a knowledge based system for neonatal resuscitation management. From academic point of view, CBR's application to this domain might be a novel concept, but before applying it to this domain one has to answer: is it a workable and valid solution?

The goals of neonatal resuscitation are to prevent the morbidity and mortality associated with hypoxic-ischemic tissue (brain, heart, kidney) injury and to re-establish adequate spontaneous respiration and cardiac output [12].

In general, APGAR- scheme [12] is used for resuscitation management which is shown in table 11.1.

Table 11.1. **APGAR scheme**[12]

SIGN	0	1	2
Heart rate/minute	Absent	Below 100	Over 100
Respiratory effort	Absent	Slow, irregular	Good, crying
Muscle tone of limbs	Flaccid	Some flexion	Active motion
Reflex stimulation(catheter in the nose)	No response	Grimace	Cries,coughs or sneezes
Colour	Blue or pale	Body pink,extremities blue	Pink

11.3. **CBR's Validity**

Before applying case based reasoning to this problem domain, we must have to clarify CBR's validity in this research. In the line of Farngou et al. [11], we would like to explore this issue of validity from two perspectives (i) AI prespective, and (ii) Domain perspective.

11.3.1. **AI perspective**

One of the important application areas of AI is expert system (ES) / knowledge based system (KBS). To solve problems KBS / ES use knowledge in the form of human experience. Obviously, the success/failure of the development and use of KBS / ES should certainly depend on the understanding of the mechanisms human experience use in solving problems. This essentially demands the understanding of the human cognitive processes. About 37 years back, Newell and Simon [13] proposed rule based approach as a model of human cognition. The rules take the form: IF < condition > THEN < action >. Using this 'condition-action' pair scheme, a good number of attempts has been made to develop KBS/ES in different domains starting from MYCIN. There are reports of relative success/failure of these attempts. Concentrating on medical domain one might observe that a very few attempts find their applications in actual medical floor. Why ? One of the reasons might be the cognitive approach in the form of rule-based reasoning. Do experts think in the form of rules and solve problems from memory ? [14].

An altenative cognitive approach presented by Riesbeck and Schank [15] is 'case based' thinking. They view that human experiences are stored in the human brain in the form of previous cases, rather than a set of rules. According to them, the basic justification for the use of CBR is that human thinking does not use logic or reasoning from first principle. In addition, CBR is not constrained to a model and so, it allows the

addition/subtraction of new cases or experience, medical practitioners acquire during case handling, without the need of complex debugging what is required in rule-based reasoning.

11.3.2. **Domain Perspective**

In medical domain doctors are supposed to observe, diagnose and advise depending on signs and symptoms. The correctness of diagnosis should certainly depend on the proper manifestation of signs and symptoms. In this regard there are a good number of uncertainties involved in the domain[16]. Any software tool, in particular, a computer based consultation systems / ES-developed to aid doctors in their decision making needs to take into account the inexact nature of information expecting to lead to rational decisions.

A number of methods have been proposed to deal with different aspects of inexact information management with their varying degrees of success. In essence [17], they can take one of the seven forms such as non-numerical techniques, categorical techniques, probabilistic modelling, ad-hoc techniques, Bayesian inference, fuzzy logic and Dempster-Shafer theory of evidence.

In expert systems, the common approaches in dealing with inexactness are : Bayesian probability approach, DS theory of evidence, Stanford CF calculus and Fuzzy set theoretic approach. In addition, inexact reasoning has itself non-monotonic aspect. It may be noted here that none of the models has yet been universally adopted by theoreticians or by practitioners. Most of the attempts have been made to manage different forms of inexactness using rule-based approach.

From the perspective explored above, it may be stated that there is no unique model for managing all types of knowledge - exact and inexact nature. Successful doctors are said to have better 'clinical eye'. The sharpness of the clinical eye should certainly depend on the experience and as experience enters into a discussion suggests the applicability of CBR approach. It is unlikely that the doctors use rule book for diagnosis of a new case. Rather, he or she are more likely to use their experiences and judgement. Moreover, case descriptions can incorporate the uncertainties of different forms.

We have tried to explore both the AI and domain perspective of applying the CBR approach to the problem domain. Convincingly, CBR approach might be a viable approach to the problem domain. This is highlighted as follows:

- No unique model is available for managing all types of knowledge of exact and inexact nature for the problem domain. CBR does not demand an explicit model, so knowledge elicitation can be achieved by acquiring cases [4].

- Doctors use heuristics and personal judgement during a consultation session in the presence of so many uncertainties, and so it is unlikely that knowledge can be as structured as in the form of production rules. This suggests that problem domain might be suited to CBR techniques.

- Updation of knowledge base is easier in CBR approach as doctors gain from their experiences. But for the rule/model-basaed approach the knowledge updating process would require complex debugging process.

- Less experienced users of a KBS/ES who lack in-depth domain knowledge, may find CBR more user-friendly since they would have the ability to retrieve cases, whether or not the user has input all the necessary problem situation data [18].

- Developing rule/model-based system is really more complex and time consuming compared to CBR approach. 800% cost saving is reported using CBR approach [19].

With the above justifications of applying CBR to this domain, we should not underestimate the role of RBR (rule-based reasoning). Because CBR may be augmented by RBR compensating the marginal insufficiencies in the former as discussed in the introduction. So, in this work an attempt has been made to develop a case-based system with rule-based augmentation.

11.4. **The System**

A case-based screening system requires a retrieval mechanism that can retrieve similar cases, a selection mechanism that can select the best match and an adaptation mechanism to fit the new situation. The present system consists of : an input interface, a case memory, and a decision maker. The decision maker has itself three components: the matching/selection module, adaptation module, and the rule based reasoner module. The schematic view of the system is shown in fig. 11.1. The system is implemented using Level5(Object), a tool kit from Information Builders Inc, USA under PC-based window environment.

For the consultation system, consultation starts with the input interface accepting the new case(s). Using the input interface it provides multi-selection-based input screen (see figs.11.2 & 11.3) for the description of the new case. The new case 'feature-values' may consists of crisp values as well as non-crisp values in terms of linguistic articulations. The input interface then passes the case to the matching/selection module. If the exact matching is not found then the control is passed to the adaptation module. After adaptation if exact matching is not found, the control is passed to the rule-based reasoner part of the system.

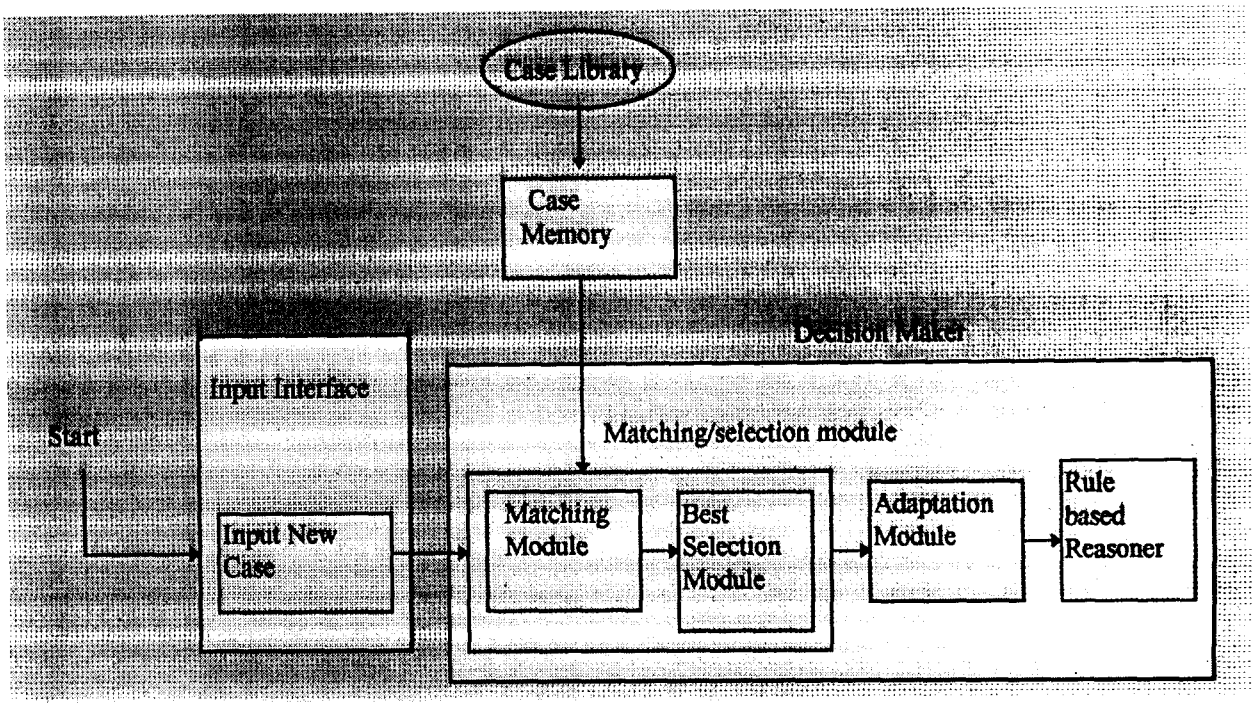


Fig.11.1. Schematic view of the system

From the domain experts, case descriptions have been collected. Although most screening cases are useful in CBR, storage of all cases (in thousands) can lead to overly large case memory [10]. Moreover, the time for screening would be undesirably high. The case memory is therefore intended to contain a representative set rather than a complete set of cases which keeps the memory size relatively small. The case memory is currently populated with 25 cases. As representative examples, we site here two cases of interest.

Case 1 : Level II resuscitation

- Respiration : None
- Heart rate : Less than 100/minute
- Colour : Whole body pink feet blue
- Muscle tone of limbs : Flexion at three joins rest flaccid
- Reflex : First touch sneeze

Case 2 : Level III resuscitation

- Respiration : Rate normal character rythmic
- Heart rate : 100/minute
- Colour : Whole body pink feet blue
- Muscle tone of limbs : Flexion at four joins rest flaccid
- Reflex : First grimace then sneeze

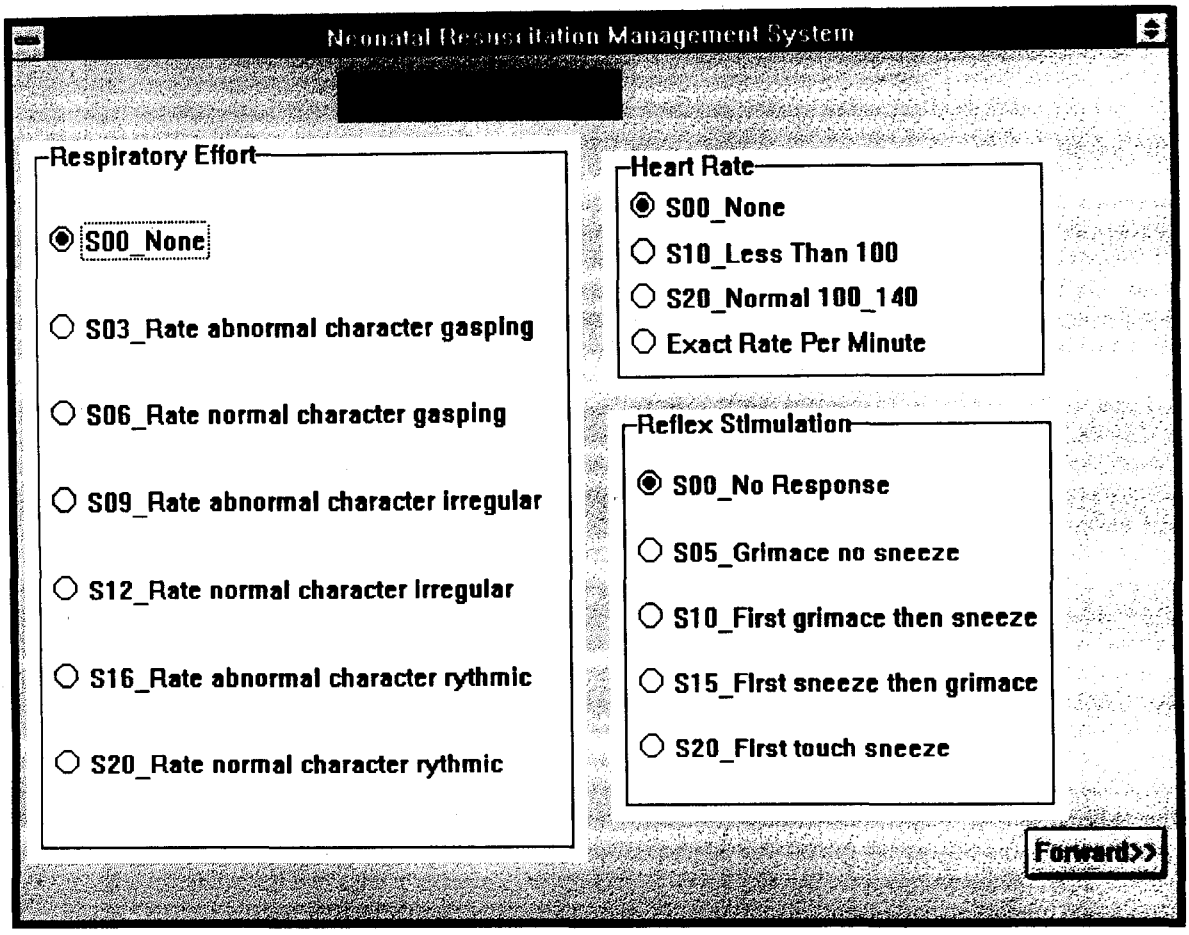


Fig.11.2. New case input

A case is a 'conceptualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner' [20]. It is important that cases are represented in a form that effectively imparts knowledge, to the user to reason with, so that a particular task can be performed [11]. In this application, the user needs to access a case (or cases) that suggests the resuscitation level of a neonate for taking further actions. There are a few issues that need addressing before a proper representation can be devised. Who will be the proposed user, Neonatalogists, general paediatricians, general medial practitioners having M.B.B.S degree or medical students undergoing training ? For its common acceptance the system should provide advice to all of its intended users (experts or novices). Therefore, the lessons presented within each case should be understandable and integrated regardless of the levels of intended users. And so system vocabulary has been selected according to the levels of intended users. The cases are grouped according to the goals, i.e., level I, II, or III. This grouping is useful for the debugging process. Every case is weighted at per the 'feature-values'. We assume that the weight of all features are equal.

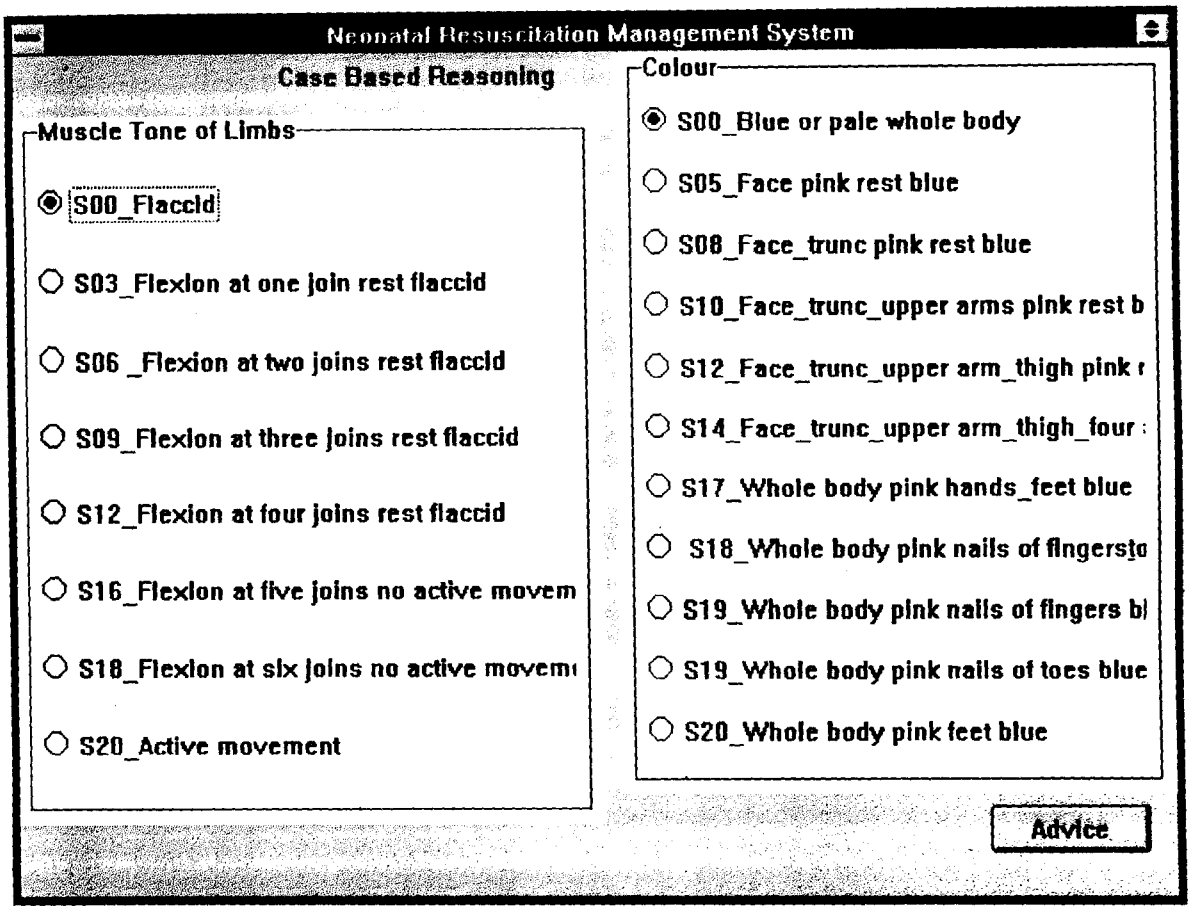


Fig. 11.3. New case input

We consider each goal as an object. Every object has its attributes - Respiration, Heart rate, Colour, Muscle tone of limbs,and Reflex. Level I object has 7 instances. Similarly, Level II and Level III have 9 instances each. We 'FIND' that instances 'WHERE' the attributes of the new case match with the old cases. The complexity of this retrieval technique is essentially linear. As the number of cases is small of the presented system, this retrieval technique might be accepted.

The overall goal of the decision maker is to advice the level of resuscitation. It consists of two parts: case retriever and rule-base reasoner. The goal of case retrieval is to find the best match between the current problem (new case) and past problem (from memory) that has a solution [15]. Upon accepting a new case, our system acts in order as follows :

Step 1

It first pursues the old cases to match the new case as 100% matching basis without any reference to the weight factors assigned to each 'feature-values'. It scans the cases at per the heighest level of weight factors assigned to the cases and the searching is stopped at the point where it gets 100% match or the number of cases are exhausted. If the scan is successful, it displays its decision; and the processing is stopped. As a

matter of fact, matching on 100% basis is a rare situation; one has to take help of an adaptation scheme as described in step 2 follows.

Step 2

In this application, we adapt the input 'feature-values'. Fuzzy reasoning techniques have been emphasized by some work on intelligent systems [21,22]. We deployed here fuzzy set theoretic approach for input adaptation. The crisp 'feature-values' are fuzzified. For example, heart rate 90/minute is fuzzified as shown in fig. 11.4. It then repeats the step 1. If exact matching is not found, it proceeds to the step 3.

Step 3

In this step, the system takes the total score calculated from the weight factors assigned to each 'feature values' of the new case under consideration into account. The output decision is fuzzified as shown in fig. 1.5. Depending on the score obtained by the new case the decision is taken by the system at per fig.11.5 with the corresponding possibility value.

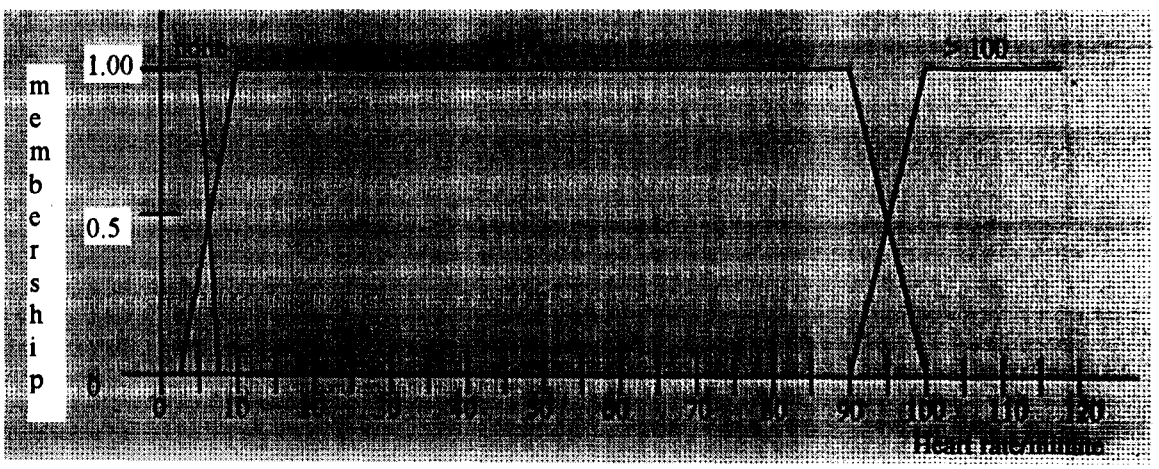


Fig.11.4. Heart rate

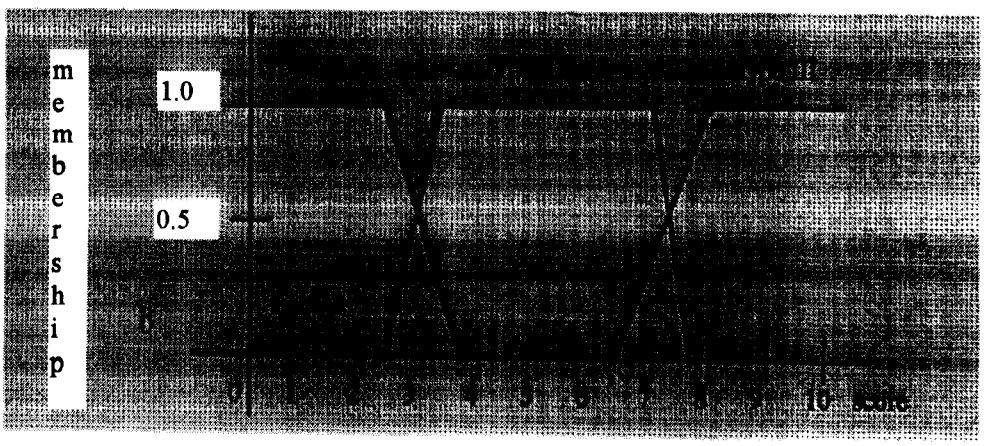


Fig. 11.5. Fuzzification of system decision

11.5. Performance Evaluation

Lastly, performance evaluation should be produced. As a matter of fact, it is to some extent difficult to establish the degree of performance owing to the non-availability of experimental results in the classical sense. Validating a case-based system different methods are there in the literature. Methods such as leave-one-out [18], using part of the case-library as test cases [23], and applying actual field cases from the domain. In order to validate the system more accurately, twenty-one case studies as available from a domain expert, who is not connected to this research, were analysed. The results suggested by the expert were compared with those suggested by our system. The system showed an accuracy of 85%.

11.6. Discussions

Starting from MYCIN, the rule based approach has long been the principal approach to use for medical diagnosis systems. The rule base paradigm suffers since it is driven neither by exemplars nor by a model (partial in most cases). For constructing intelligent systems CBR is viable alternative to rule-based systems. Current research indicates that case based approach may be better suited to medical diagnosis systems. But, however, the appeal of RBR should not be overruled. Case-based reasoning is augmented by rule-based reasoning compensating the marginal insufficiencies in the former. This chapter explores the applicability of CBR to neonatal resuscitation management with rule-based augmentation. Although, the accuracy of the current system is less than that we achieved [24] using fuzzy rule based approach (95%), nevertheless the results are encouraging. The discrepency of two results is not surprising. The accuracy of the current system would increase with the increase of cases in the memory.

References

1. F. Coenen and T. J. M. Bench-Capon. Building knowledge based systems for maintainability. Proc. of DEXA '92; Valencia, 1992.

2. J. E. Vargas and S. Raj. Developing maintainable expert systems using case-based reasoning. Expert Systems; vol. 10, no. 4, 1993, 219-225.

3. J. S. di Piazza and F. A. Helsabeck. Laps : Cases to Models to Complete Expert Systems. AI Magazine; Fall 1990, 80-107.

4. I. Watson. An Introduction to Case-Based Reasoning. Proc. of the 1st UK CBR Workshop; 12th Jan. 1995. The BCS Specialists Group on ES, University of Salford.

5. O. Babka and Jiri Hlasivce. Case-based reasoning : reusing a successful solution. Proc. of Int. Conf. on advances in SSCC; 1998, Durban, South Africa, vol. II, 186-189.

6. W. M. Roddis and J. Bocox. Case-Based approach for steel bridge fabrication errors. J. Comp. in Civil Engg; ASCE, 1997, 84-91.

7. L. Hunter. Knowledge acquisition planning : Results and prospects. Proc. of the Sixth International Machine Learning Workshop; San Mateo, CA : Morgan Kaufmann, 1989.

8. A. K. Goel, J. L. Kolodoner, M. Pearce, R. Billington and C. Zimring. Towards a case-based tool for aiding conceptual design problem solving. In E. R. Bareiss (ed.) Proc. of the DARPA Case-Based Reasoning Workshop; San Francisco : Morgan Kaufmann, 1991.

9. L. K. Branting and B. W. porter. Rules and precedents as complementary warrants. Proc. of AAAI-91; Cambridge, MA : AAAI Press / MIT Press.

10. Li D. Xu. Developing a case-based knowledge system for AIDS prevention. Expert Systems; vol. 11, no. 4, 1994, 237-244.

11. Andreas J. Frangou, Yan Wan, J. Antony and M. Kaye. ESAS : A case-based approach to bussiness planning and monitoring. Expert Systems; vol. 15, no. 3, 1998, 182-196.

12. W. E. Nelson. Text Book of Pediatrics. 15th edition, 1996, Book 1, W. B. Saunders Company; Bangalore.

13. A. Newell and H. A. Simon. Human Problem Solving. Englewoods Cliffs, NJ : Prentice-Hall; 1972.

14. I. Watson and S. Abdullah. Developing case-based reasoning systems : A case study in diagnosing building defects. Colloquium on 'Case Based Reasoning : Prospects for Applications'; London, 3rd. March, 1994, IEE.

15. C. K. Riesbeck and R. C. Schank. Inside Case-Based Reasoning. Hillsdale : Lawrence Erlbaum Associates;1989.

16. A. K. Saha, M. G. Goswami, R. K. Samanta and M. Chatterjee. Child Growth and Development, Expert Systems and Inexact Information Management. Proc. Computer Society of India, 9-12 Nov, 1995, Hyderabad, Tata McGraw Hill Pub. Co.; 258-267.

17. P. L. Reynolds, P. W. Sanders and C. T. Stockel. Uncertainty in telecommunication network design. Expert Systems; vol. 12, no. 3, 1995, 219-228.

18. Yan. Wan. Cost estimation of sewage treatment systems using artificial intelligence. Ph.D. Thesis, University of Portsmouth; UK, 1996.

19. E. Simoudis. Knowledge acquisition in validated retrieval. Int. J. of Expert Systems; vol. 4, no. 3, 1992, 299-315.

20. J. Kolodoner. Case-Based Reasoning. Morgan Kaufmann; USA, 1993.

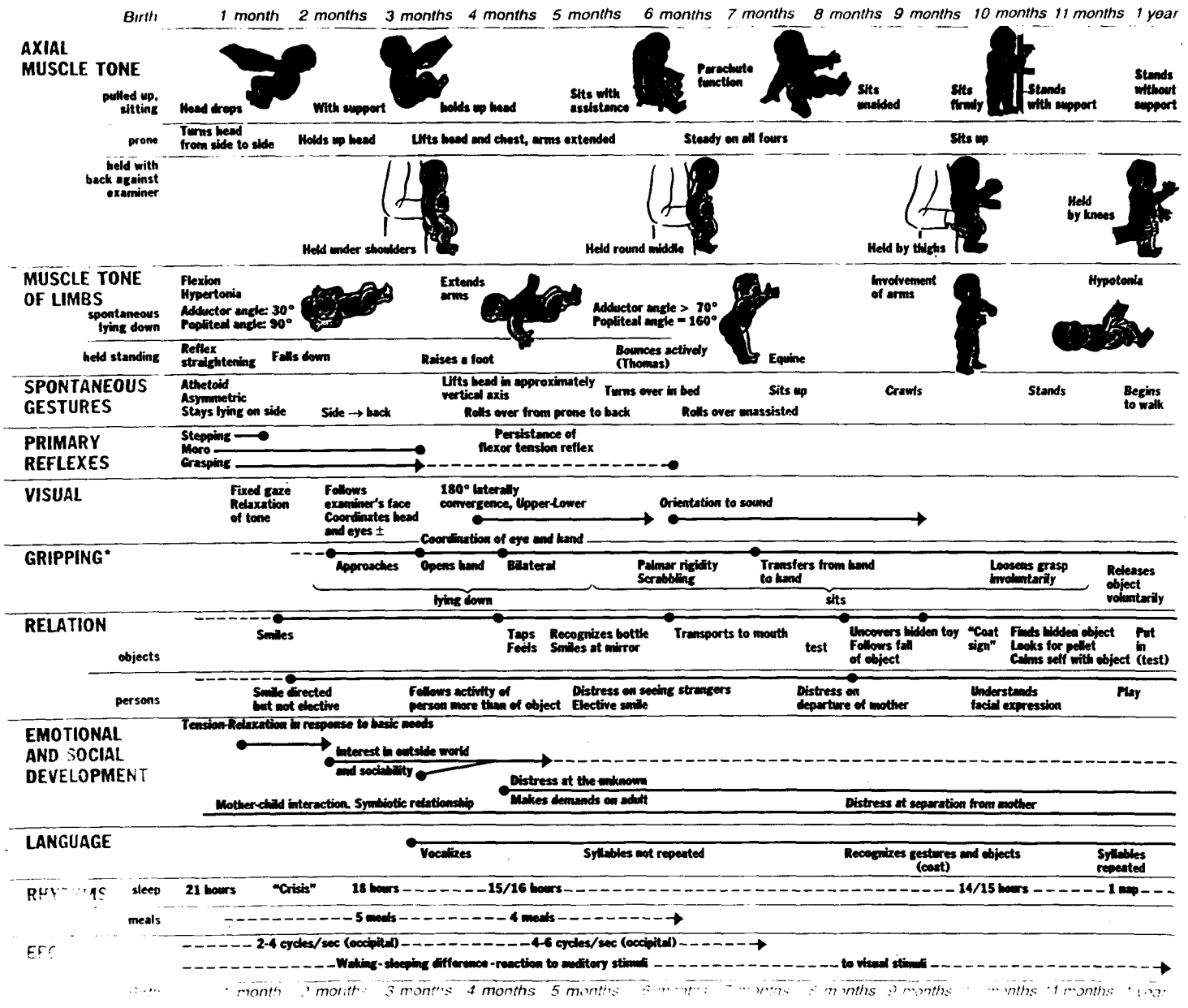
21. P. Hajek et al. Uncertain Information Processing in Expert Systems. CRC Press, Boca Raton; 1992.

22. J. Kolodoner and W. Mark. Case-Based reasoning. IEEE Expert; vol.7, no. 5,1992, 5-6.

23. M. Goodman. Prism : A case-based telex classifier. In Innovative Applications of Artificial Intelligence, vol. 2, A. Rappaport and R. Smith (ed), Cambridge, MA : MIT Press; 1990.

24. R. K. Samanta, A. K. Saha, M. G. Goswami and M. Chatterjee. On Fuzzy Concepts in Paediatrics within Expert Systems Framework. Vivek; vol. 11, no. 1, 1998, 3-15.

Child Activities



* Very previous performances may be observed in the approach to an object at a size of between 15 days and 1 month is very short and at the age of the child is related by the examiner (referred to as "fixed" motor development)

PULLED UP, SITTING: A test in which the child, supported behind the shoulders, is raised [illegible] the head with the sitting position

MUSCLE TONE OF LIMBS: the first two drawings on the [illegible] are concerned with the position spontaneously assumed by the child when asleep or [illegible] back [illegible] and their strength

Grasping reflex: the whole hand closes [illegible] against an object placed in the palm
Flexor tension: flexion of the fingers persists when an object is grasped
GRIPPING: the first stages indicated are to be seen only when the child is hung on its back

	1 year	15 months	18 months	21 months	24 months
GROSS MOTOR DEVELOPMENT	Walks holding one hand, then alone	Goes up, one step with assistance Crawls on all fours alone Crouches when playing	Goes upstairs alone (holding on) Sits down alone	Climbs on (toboggan) ++	Tricycle (pedals) Jumps with both feet Goes up and down stairs
USE OF OBJECTS known	Plate Coat Shoes	Appropriate gestures for dressing Drinks alone from glass Scribbles when told to Points to what has been done	Takes part in dressing Turns over pages of book Eats unaided Seeks self behind mirror	Opens door Turns on light Approaches before climbing	Kicks (ball) Takes off shoe
tests	Puts cube in cup Pincer movement of thumb and forefinger	Stacks 2 cubes. Fills cup Pellet in bottle Fits in shapes ●	Stacks 3 cubes Dumps pellet from bottle	Lines up cubes (train) ● ■	Imitates train Folds paper ● ■ ▲
SOCIAL (behaviour) parents	Play ++	Recognizes after absence (regular, but not daily visitor)			
strangers	Suspicion				
children		Parallel game	Explores: self / others	(sometimes aggressive: bites...)	
EMOTIONAL AND SOCIAL DEVELOPMENT**		Sensorimotor phase			Symbolic phase
		Process of separation	Individualization		Appearance of "I"
		• Frustration-Permission			
		• Autonomy-Dependence			
		• Gives-Takes back			
			Anticipation of return "transitional" object		
LANGUAGE understanding	Take-give Defence Goodbye Bravo	Brothers and sisters Names of various objects	Orders (gather up, go and look, shut, sit, stand etc.) Many objects		
tests				Clothing 5 parts of body 3 different places	Listens to story 4 pictures shown
	rapid enrichment				
expression	5 words	Jargon	10-12 words Names pictures	100 words Begins to put phrases together (2 words)	
				rapid enrichment	
SPHINCTERS	Not clean	Passively clean		Clean during day (with accidents) Sometimes clean at night	Goes on own
SPONTANEOUS PLAY	Motor functions-tactile exploration (accidents)		Transports-stacks hides-finds		Animistic or imitative play (dolls, playing houses)
EEG	Recording of sleep interesting:	hypersynchronism of falling to sleep light sleep spikes	Deep sleep spindles		
	1 year	15 months	18 months	21 months	24 months

** These reference points are, of course, highly schematic and are intended merely as essential guides that are continuously varied by the various situations in which the child is placed.

LANGUAGE: 5 parts of the body: indicates them when asked
3 different places: puts in chair in 3 different places

30 Months | 3 Years | 4 Years | 5 Years | 6 Years | 7 Years

GROSS MOTOR DEVELOPMENT

Runs stiffly
Runs on uneven ground
Carries bulky objects

Goes up stairs one foot after the other.
Stands on one leg. Hops
Pedals well. Bicycle without side wheels
Opens door,
enters and closes door unaided
Gets in and out of car

Jumps with
feet together

Puts ball
in goal

VISUAL AUTONOMY

Helps actively
with dressing

Undresses with assistance
Begins to dress
Undoes buttons large buttons
Unlaces shoes begins to lace up
Washes hands Washes face
Dresses doll

Goes to W.C. alone
all buttons
ties bow
does hair
able to wash alone

Simple shopping
Opens lock
with key
Telephones

Fork

Eats properly unaided

Pours

Drinks with straw

Helps self
unassisted

Cuts meat

FINE MOTOR-ADAPTIVE

Stacking
Fitting together
Puzzles (simple)

Tests:
- 4 beads in 2 min.
- Bridge of blocks
- Sewing
- Lacing (first appearance begins to lace...)

Puts away in box
Cuts with scissors

Laces properly (turning over)

10 matches in box in 20 seconds
Rolls up string
Sews with large needle
Does not turn over

Disappearance of synkinesis
(hand movements)

Realization of personal
left and right

Screwing:
understands the movement — performs it

EMOTIONAL AND SOCIAL DEVELOPMENT

Opposition phase

Identification-Rivalry

Locating
persons and places
Socialization

Idea of sex - Oedipal stage

LANGUAGE

Everyday objects
Simple orders
Parts of body
Clothing, etc.

Colours
Where is it?
Why?
High, low
Large, small

Yesterday, tomorrow
Like, unlike
When? (duration)
How?

Tests:
- 2nd degree comprehension
- Missed train
- Fire in house...
- Logical order:
colours, thicknesses, dimensions

Words with articles
Substantive verbs

Incorrect tenses
Unconnected words

States own sex

Tenses of verbs
Accumulation of words

States name and age

Mistakes in logic
Phrases put together

Knows address and telephone number
Extension of vocabulary

Correct language
Improved syntax

DRAWING

Attempt to copy
vertical and horizontal lines
Scribbles
Centripetal lines ± deflection

Completing half a circle

full circle

Rudimentary human figure

Various drawings
(trees, houses etc)

Whole body
turned towards objects

Start of schooling
Reception class

Writes

Positions, transparencies

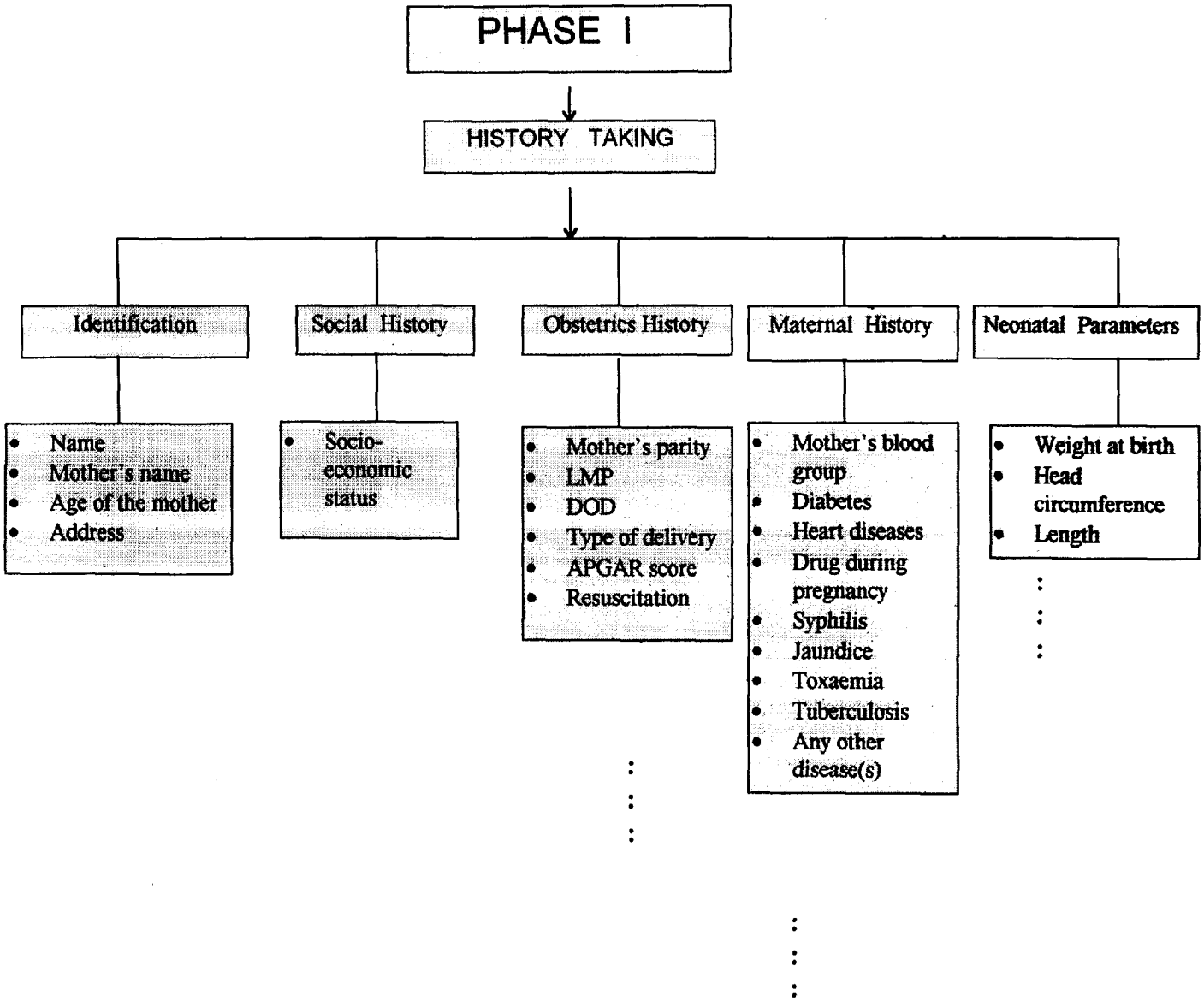
Elementary class 1

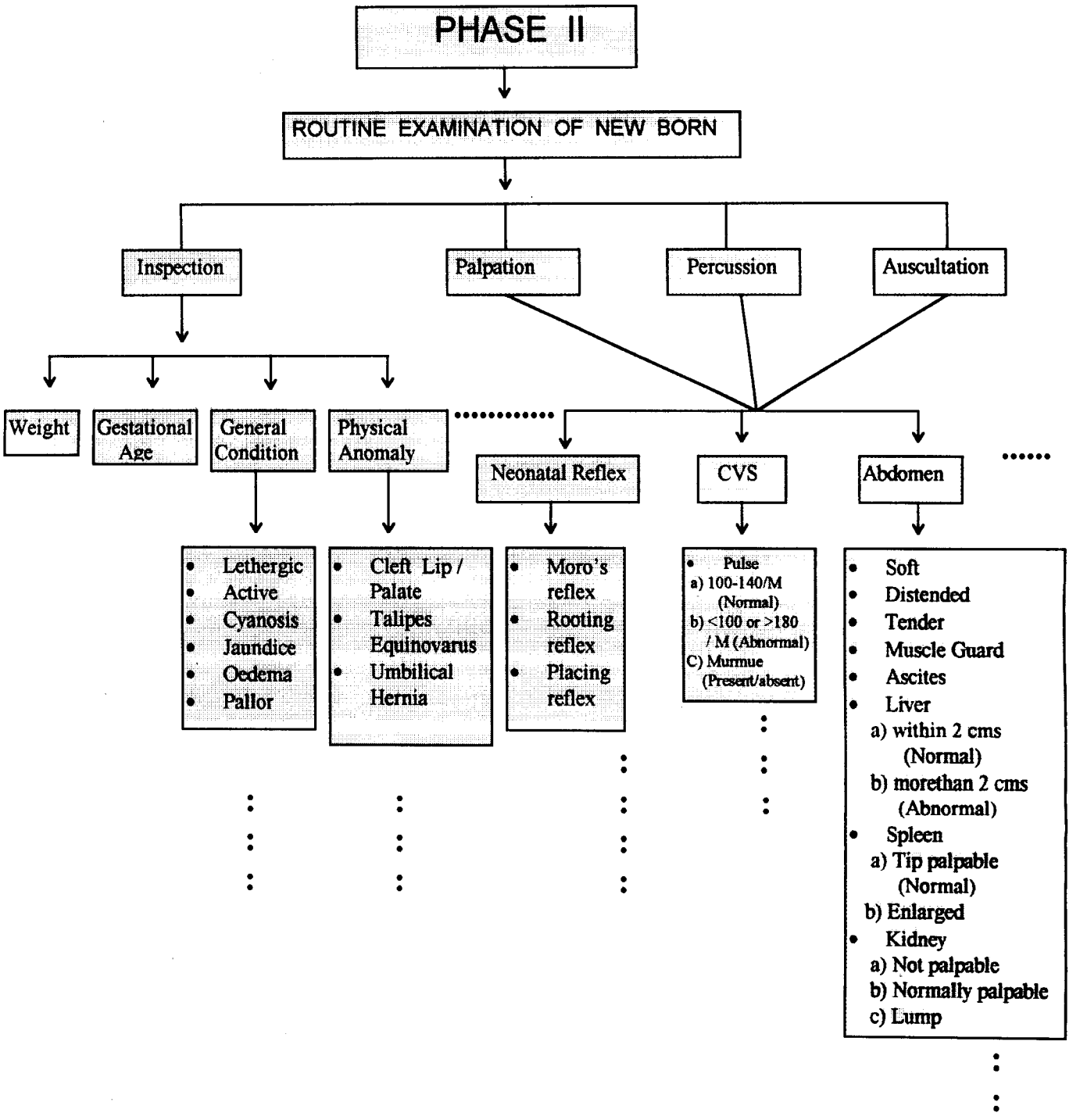
AUTONOMY: the successive stages corresponding to the unknown and to the need for the child of large and then of smaller objects

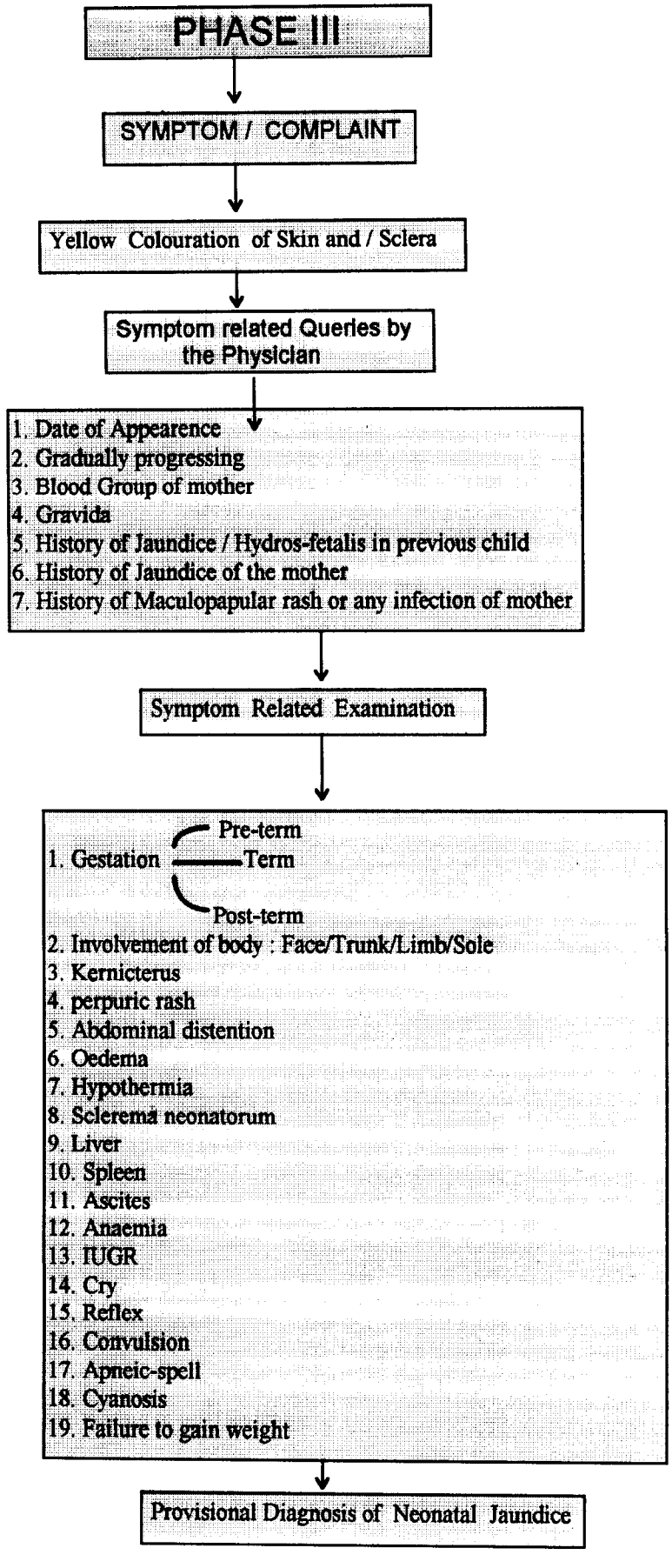
Personal spatial orientation: [illegible] but not on the other side

2nd DEGREE COMPREHENSION: the relationship of the questions of the child to Sensation from itself, what it can see, can miss, the loss of the house, car, bus, etc.

Typical Classes and Objects


PHASE I
HISTORY TAKING
Identification
Social History
Obstetrics History
Maternal History
Neonatal Parameters
Name
Mother's name
Age of the mother
Address
Socio-economic status
Mother's parity
LMP
DOD
Type of delivery
APGAR score
Resuscitation
Mother's blood group
Diabetes
Heart diseases
Drug during pregnancy
Syphilis
Jaundice
Toxaemia
Tuberculosis
Any other disease(s)
Weight at birth
Head circumference
Length


PHASE II
ROUTINE EXAMINATION OF NEW BORN
Inspection
Palpation
Percussion
Auscultation
Weight
Gestational Age
General Condition
Physical Anomaly
Neonatal Reflex
CVS
Abdomen
• Lethergic
• Active
• Cyanosis
• Jaundice
• Oedema
• Pallor
• Cleft Lip / Palate
• Talipes Equinovarus
• Umbilical Hernia
• Moro's reflex
• Rooting reflex
• Placing reflex
• Pulse
a) 100-140/M (Normal)
b) <100 or >180 / M (Abnormal)
C) Murmue (Present/absent)
• Soft
• Distended
• Tender
• Muscle Guard
• Ascites
• Liver
a) within 2 cms (Normal)
b) morethan 2 cms (Abnormal)
• Spleen
a) Tip palpable (Normal)
b) Enlarged
• Kidney
a) Not palpable
b) Normally palpable
c) Lump


PHASE III
SYMPTOM / COMPLAINT
Yellow Colouration of Skin and / Sclera
Symptom related Queries by the Physician
1. Date of Appearence
2. Gradually progressing
3. Blood Group of mother
4. Gravida
5. History of Jaundice / Hydros-fetalis in previous child
6. History of Jaundice of the mother
7. History of Maculopapular rash or any infection of mother
Symptom Related Examination
1. Gestation
Pre-term
Term
Post-term
2. Involvement of body : Face/Trunk/Limb/Sole
3. Kernicterus
4. perpuric rash
5. Abdominal distention
6. Oedema
7. Hypothermia
8. Sclerema neonatorum
9. Liver
10. Spleen
11. Ascites
12. Anaemia
13. IUGR
14. Cry
15. Reflex
16. Convulsion
17. Apneic-spell
18. Cyanosis
19. Failure to gain weight
Provisional Diagnosis of Neonatal Jaundice

PHASE IV

↓

Differential Diagnosis

↓

1. Physiological Haemolytic Disease
2. Septicemia
3. Intra-uterine infection
4. Cytomegalo inclusion bodies
5. Toxoplasmosis
6. Familial non-haemolytic Jaundice
7. Biliary Atresia
8. Neonatal Hepatitis
9. Hereditary Spherocytosis
10. Drug induced Haemolytic Aneamia
11. Galactosemia
12. Inspissated bile syndrome bile
13. Cretinism

⋮

↓

MANAGEMENT

↓

Investigation → **FINAL DIAGNOSIS** → **TREATMENT**

Preliminary Treament

FINAL DIAGNOSIS & TREATMENT

An Example

Specific Signs and Symptoms

1. Day - 3rd - Onwards
2. Lethergy
3. Refused to Suck
4. Poor Cry
5. Poor Reflex
6. Circumoral Cyanosis
7. Vomiting
8. Irretability
9. Apneic-spell
10. Loose Motion
11. Abdominal Distention
12. Hyperthermia & Hypothermia
13. Failure to gain weight
14. Ecchymosis / Perpuric rash
15. Staphylococcal Skin Infection
16. Umbilical Sepsis
17. Pallor
18. Convulsion
19. Sclerema
20. Dic & Shock
21. Necrotizing Enterocolitis

⋮

→ Jaundice due to Septicemia →

↓ Preliminary Treatment of Jaundice with Broad Spectrum Antibiotics, Feeding Advice, & Temperature Maintenance

Specific Investigations

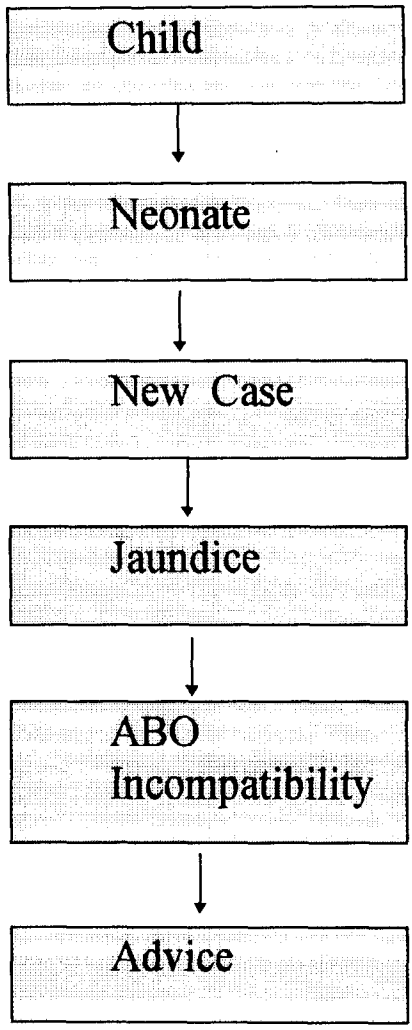
a. Blood Culture
b. Swab Culture
c. TC, DC, Hb% Toxic grannules & Band cells
d. 'C' Reactive Protein

⋮

→ **Treatment According to drug Sensitivity and other Investigations**

KID

CASE A


Child
Neonate
New Case
Jaundice
ABO Incompatibility
Advice

KID: A Paediatric Adviser

Foetal | Neonate | Infants | Toddler | Pre-School | School-Going | Adolescence | Help

✓New Case

✓Resuscitation

✓Exit

Welcome To 'KID': A Paediatric Adviser

KID : A Paediatric Adviser

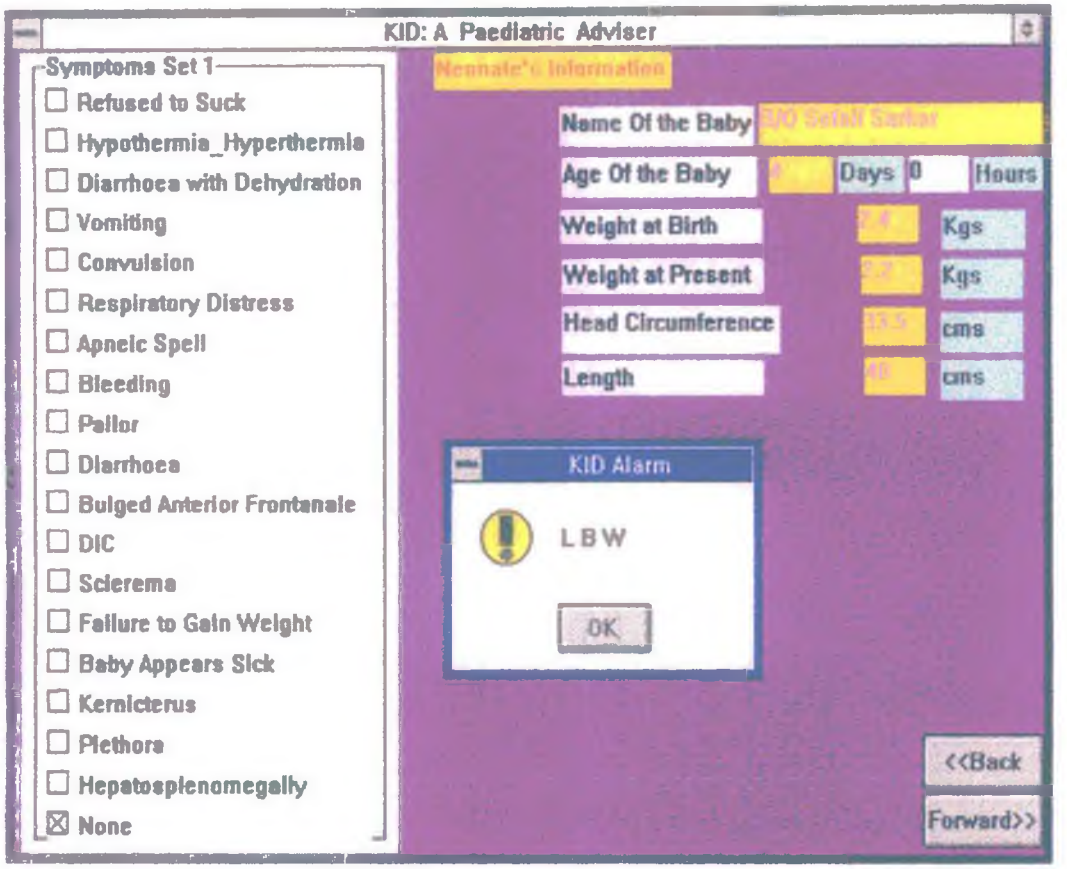
This has been developed at the Expert Systems Laboratory of the Department of Computer Applications of North Bengal University with the financial assistance from CSIR, Govt. of India by M.G.Goswami, Dr. A. K. Saha, Dr.(Mrs) Mridula Chatterjee, and Dr. R. K. Samanta.

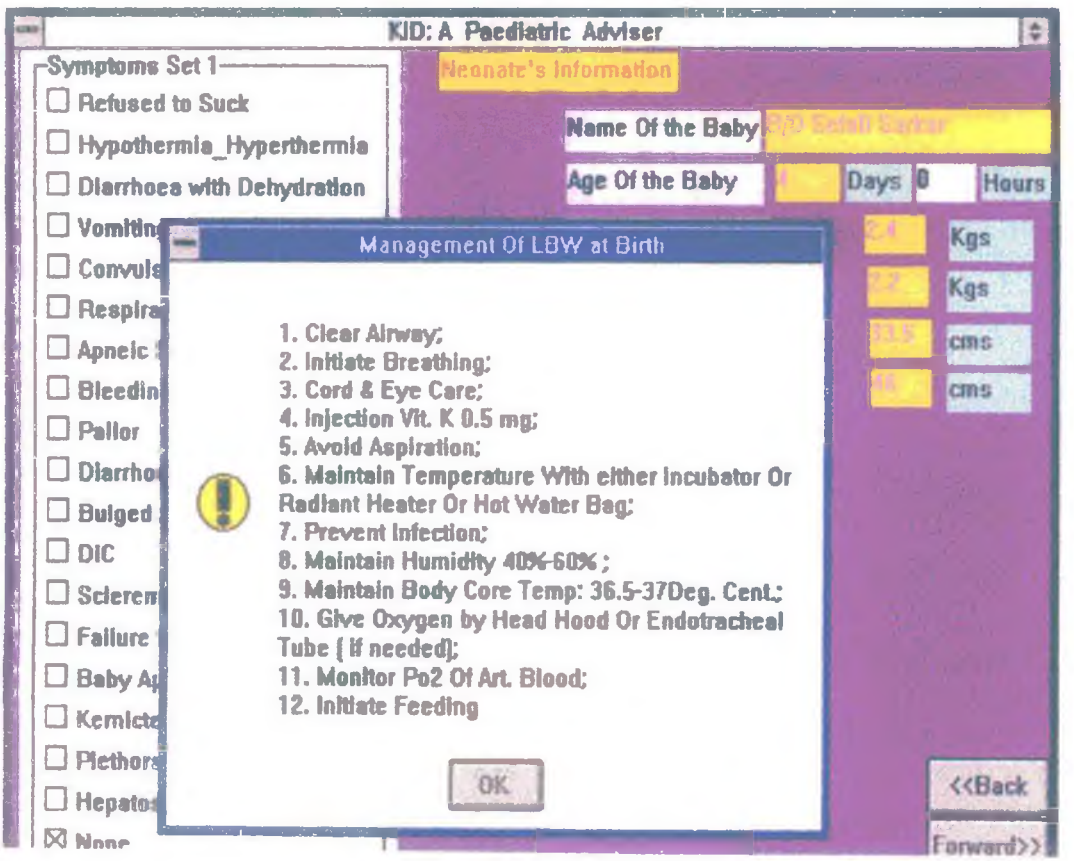
KID: A Paediatric Adviser

Neonates

A Knowledge-based Paediatric Adviser

Exit | About... | Forward>>


KID: A Paediatric Adviser
Symptoms Set 1
Refused to Suck
Hypothermia_Hyperthermia
Diarrhoea with Dehydration
Vomiting
Convulsion
Respiratory Distress
Apneic Spell
Bleeding
Pallor
Diarrhoea
Bulged Anterior Frontanale
DIC
Sclerema
Failure to Gain Weight
Baby Appears Sick
Kernicterus
Plethora
Hepatosplenomegally
None
Neonate's Information
Name Of the Baby
Age Of the Baby
Days
0
Hours
Weight at Birth
Kgs
Weight at Present
Kgs
Head Circumference
cms
Length
cms
KID Alarm
LBW
OK
<<Back
Forward>>


KID: A Paediatric Adviser
Symptoms Set 1
Refused to Suck
Hypothermia_Hyperthermia
Diarrhoea with Dehydration
Neonate's Information
Name Of the Baby
Age Of the Baby
Days
0
Hours
Kgs
Kgs
cms
cms
Management Of LBW at Birth
1. Clear Airway;
2. Initiate Breathing;
3. Cord & Eye Care;
4. Injection Vit. K 0.5 mg;
5. Avoid Aspiration;
6. Maintain Temperature With either Incubator Or Radiant Heater Or Hot Water Bag;
7. Prevent Infection;
8. Maintain Humidity 40%-60% ;
9. Maintain Body Core Temp: 36.5-37Deg. Cent.;
10. Give Oxygen by Head Hood Or Endotracheal Tube (If needed);
11. Monitor Po2 Of Art. Blood;
12. Initiate Feeding
OK
<<Back
Forward>>

KID: A Paediatric Adviser

Neonate's Information

IUGR TRUE

Neonate Sex
- Male
- Female

Mother's Name Sefali Sarkar

LMP 26/11/98 dd/mm/yy

Mother's Age 28 Years Parity 0

Address W/O Pintu Sarkar

Weeks

About Mother
- Diabetes
- Hypertension
- Syphilis
- Jaundice
- Drugs Taken
- Toxemia Of Pregn
- Rash
- Fever
- Adenopathy
- Arthritis
- Viral Illness
- Fetal Loss
- None

Blood Group Of Mother
- A
- O
- B
- AB
- Not Known

Rh Factor
- Positive
- Negative
- Not Known

DOD 31/08/99 dd/mm/yy

Gestation Term

Type Of Delivery
- Normal
- Low Forceps
- High Forceps
- LUCS
- Ventouse

After Birth Cry
- Cried
- Not Cried

Resuscitation
- Not Required
- Oral Suction _Manual S
- Oral Suction _Injection
- Endotracheal Intubation

APGAR Score

Gest Age <<Back Forward>>

KID: A Paediatric Adviser

Neonate's Information

IUGR TRUE

Neonate Sex
- Male
- Female

Mother's Name Sefali Sarkar

LMP 26/11/98 dd/mm/yy

Mother's Age Years Parity 0

About Mother
- Diabetes
- Hypertension
- Syphilis
- Jaundice
- Drugs Taken
- Toxemia Of Pregn
- Rash
- Fever
- Adenopathy
- Arthritis
- Viral Illness
- Fetal Loss
- None

KID Alert: Term Baby

Present Weight is Proper

OK

Weeks

Gestation Term

Not Known

After Birth Cry
- Cried
- Not Cried

Resuscitation
- Not Required
- Oral Suction _Manual S
- Oral Suction _Injection
- Endotracheal Intubation

APGAR Score

Gest Age <<Back Forward>>

KID: A Paediatric Adviser

Physical Examination

Mouth
- [x] Normal
- [ ] Cleft Lip
- [ ] Cleft Lip _Cleft Palate
- [ ] Cleft Palate
- [ ] Microgonathia
- [ ] Glossoptosis
- [ ] Facial Asymetry
- [ ] Excessive Frothing
- [ ] Bifid Uvula
- [ ] Retention Cyst Of The Gum
- [ ] Yellow Follicle in Anterior Tonsillar F
- [ ] Sucking Pad
- [ ] Transitory Teeth
- [ ] Ebstein Pearl
- [ ] Bohn Nodule
- [ ] Mandibulofacial Dysostosis
- [ ] Unilateral Hypoplasia of Mandible

Chest
- (●) Normal Shaped
- ( ) Abnormal Shape
- ( ) Breast Hypertrop
- ( ) Supernumerary I
- ( ) Neuromuscular c
- ( ) Central Hypoven

Limbs
- [x] Normal Limbs
- [ ] Constriction Deformity
- [ ] Paresis
- [ ] Earbs Palsy
- [ ] Tenderness _Swelling

Neck
- [x] Normal Neck
- [ ] Sternomastoid Tumour
- [ ] Branchial Cyst
- [ ] Thyroglossal Cyst
- [ ] Cystic Hygroma
- [ ] Webbing
- [ ] Congenital Torticollis

Feet
- [x] Normal Feet
- [ ] Talipes Equinovarus
- [ ] Polyductyly
- [ ] Hypertropy
- [ ] Absent Nail
- [ ] Synductyly
- [ ] Metatarsus Adductus
- [ ] Calcaneovalgus Feet
- [ ] Vertical Talus or Rock
- [ ] Toe Deformity

Hip
- (●) Complete Abduction
- ( ) Not Complete Abduct

<< Back Forward >>

KID: A Paediatric Adviser

Skull Bones
- [x] Normal
- [x] Frontanalae Normal
- [ ] Frontanalae Craniostosis
- [ ] Large Anterior Frontanalae
- [ ] Depression of the Skull
- [ ] Craniotabes
- [ ] Moulded
- [ ] Caput Succedenium and Ce
- [ ] Craniosynostosis
- [ ] Scaphocephaly
- [ ] Frontal plagiocephaly
- [ ] Occipital Plagiocephaly
- [ ] Trigonocephaly
- [ ] Turricephaly
- [ ] Kleeblattschadel deformity
- [ ] Crouzon syndrome
- [ ] Apert syndrome
- [ ] Carpenter syndrome
- [ ] Chotzen syndrome
- [ ] Pfeiffer syndrome

Physical Exam

Rectum
- (●) Normal
- ( ) Imperforate

Cry
- (●) Shrill Cry
- ( ) Poor

Miscellaneous
- [ ] Tremulous Ankle M
- [ ] Mild Puffiness Ove
- [ ] Icthyasis Congenit
- [ ] Neurological Abnor
- [ ] Septicaemia
- [x] Hyperbilirubinemia
- [ ] Diarrhoea with Deh
- [ ] Seizure

Oedema
- ( ) Present
- (●) Absent

Reflex
- (●) Good
- ( ) Poor

Umbilical Cord
- ( ) Unhealthy Purulant dischar
- ( ) Unhealthy Umbilical Granul
- ( ) Single Umbilical Artery
- (●) Normal
- ( ) Urachal cyst or Persistant U
- ( ) Congenital Omphalocele
- ( ) Tumours
- ( ) Haemorrhage
- ( ) Umbilical Polyp
- ( ) Umbilical Hernia
- ( ) None

<< Back Forward >>

KID: A Paediatric Adviser

Physical Examination

Ears
- [x] Normal Ear
- [ ] Preauricular Pit
- [ ] Cartilage Not Well Formed
- [ ] Preauricular Skin Tag
- [ ] Prominent or Loop Ear
- [ ] Microtia or Anotia
- [ ] Malformation of External Ear
- [ ] Congenital Stenosis of Exter
- [ ] Low Set Ears

Nose
- [x] Normal nose
- [ ] Choanal Atresia
- [ ] Depressed Bridge
- [ ] Perforation of nasal Septur
- [ ] Septal Deviation
- [ ] Midline Nasal Mass

Eyes
- [ ] Normal
- [ ] Conjunctival Haemo
- [ ] Cornea More Than
- [ ] Bilateral Red Reflex
- [ ] Cataract Visual Pro
- [ ] Microphthalmia
- [ ] Long Philtrum
- [ ] Retinal Haemorrhag
- [ ] Heterocromia

Eyes 2
- [ ] Aniridia
- [ ] Gen Exam of eyes
- [ ] Coloboma
- [ ] Microcaria
- [ ] Congenital Mydriasis
- [ ] Dyscoria and Corectopia
- [ ] Anisocoria
- [ ] Persistent Pupillary Mer
- [ ] Heterochromia
- [ ] Leukocoria
- [ ] Ptosis
- [ ] Epicanthal Folds
- [ ] Hypertelorism
- [ ] Hypotelorism
- [ ] Microspherophakia

<<Back Forward>>

KID: A Paediatric Adviser

Heart
- [x] Normal 100 To 140 Per Mi
- [ ] Abnormal: Less 100 Or Gr
- [ ] Murmur Present
- [ ] Murmur Absent

Larynx
- [x] Normal
- [ ] Atresia of Larynx
- [ ] Laryngeal Web
- [ ] Laryngotrach
- [ ] Congenital La
- [ ] Tracheoesopl

Liver
- (•) Normal Withi
- ( ) More Than 2

Spleen
- (•) Normal Tip Palpable
- ( ) Enlarged

Physical Examination

Skin
- [x] Normal Colour
- [ ] Cyanosis
- [ ] Acrocyanosis
- [ ] Mottling
- [ ] Harlequin Colour Change
- [x] Ichterus
- [ ] Pallor
- [x] Mongolian Spot
- [ ] Tuft of Hair At LS Spine
- [ ] Aplasia Cutis Congenita
- [ ] Goltz Syndrome
- [ ] Preauricular sinus and P

KID Alert: Take care of Skin

Slate gray or bluish macular lesion normally over back. Usually fades.

OK

Kidneys
- ( ) Palpable
- (•) Not Palpable
- ( ) Lump

Skin 2
- [ ] Sebaceous Hyperplasi
- [ ] Milia
- [ ] Sucking Blisters
- [ ] Sucking pads
- [ ] Cutis Marmorata
- [ ] Port-wine stain
- [ ] Pimples
- [ ] dant Skin
- [ ] tic Constriction B
- [ ] sory Tragi
- [ ] hial cleft and Cys
- [ ] glossal Cysts an
- [ ] Supernumerary Nipple
- [ ] Focal Facial Dermal Dy

<<Back Forward>>

KID: A Paediatric Adviser

Physical Examination

Abdomen
- [x] Soft
- [ ] Distended
- [ ] Tender
- [ ] Muscle Guard
- [ ] Ascites
- [ ] Diastasis Recti
- [ ] Umbilical Hernia

Breathing
- [x] Normal 30 To 40 Per Minute
- [ ] Greater Than 60 Per Minute
- [ ] Gasping
- [ ] Diaphargmatic or paradoxdcal Breathing
- [ ] Apnea
- [ ] Respiratory Distress
- [ ] Moderate and Severe Asphyxda

Genitilia
- [x] Normal Genitilia
- [ ] Imperforate Hymen
- [ ] Crypto Orchism
- [ ] Retractile Testicle
- [ ] Ambiguous Genitilia
- [ ] Colour and Rugae and L
- [ ] Bleeding PV on 2nd or 3
- [ ] Nonpurulant Discharge I
- [ ] Transitory Hydrocele
- [ ] Hypospadias
- [ ] Epispadias

Feeding
- [ ] Gavage
- [ ] IV
- [x] Normal

Breast
- (●) Breast Hypertrophy
- ( ) Supernumerary Nipple

<<Back | Management | Analysis

KID: A Pediatric Adviser CASE: Jaundice

Appearence Day
- (●) First
- ( ) After Second
- ( ) Second To Fifth
- ( ) Four To Seven
- ( ) Fifth To Tenth
- ( ) After Seven
- ( ) After Tenth
- ( ) Not Known
- ( ) Within Three Days
- ( ) Within Seven Days
- ( ) Second To Seventh
- ( ) Second To Six Weeks

Disappearance Day
- ( ) Within Seven Days
- ( ) Within Fifteen Days
- ( ) Within One Month
- ( ) Persists After One Month
- (●) Not Known

Urine
- ( ) Dark Yellow
- (●) Yellow
- ( ) Normal
- ( ) Not Known

Symptomatic Jaundice

Mother's issue
- ( ) First
- (●) Second
- ( ) Third

Stool
- (●) Light
- ( ) Normal
- ( ) Acolic
- ( ) Loose Motion
- ( ) Pale Yellow
- ( ) Constipation
- ( ) Meconeum
- ( ) White_Pale

Jaundice Type
- [ ] Mild
- [ ] Moderate
- [x] Severe
- [ ] Persistant
- [ ] Progressing

Forward>>

KID: A Pediatric Adviser CASE: Jaundice

Symptoms Set 1
- [ ] Refused to Suck
- [ ] Hypothermia_Hyperthermia
- [ ] Diarrhoea with Dehydration
- [ ] Vomiting
- [ ] Convulsion
- [ ] Respiratory Distress
- [ ] Apneic Spell
- [ ] Bleeding
- [ ] Pallor
- [ ] Diarrhoea
- [ ] Bulged Anterior Frontanale
- [ ] DIC
- [ ] Sclerema
- [ ] Failure to Gain Weight
- [ ] Baby Appears Sick
- [x] Kernicterus
- [ ] Plethora
- [ ] Hepatosplenomegally

Symptom Set 2
- [ ] Hepatomagally
- [ ] Spleenomegally
- [ ] Poor Activity
- [ ] Shock and Coma
- [ ] Psychomotor Retardation
- [ ] Neuromotor disorder
- [ ] Fever
- [ ] Purpura
- [ ] Jaundice more than 14 days

Heart Rate
- ○ Normal
- ○ Abnormal

Cry
- ◉ Shrill Cry
- ○ Poor
- ○ Failure to Cry

Reflex
- ○ Good
- ◉ Poor

Gestation
- ○ Preterm
- ◉ Term
- ○ Postterm

Signs Jaundice

Abdomen
- [x] Soft
- [ ] Distended
- [ ] Tender
- [ ] Muscle Guard
- [ ] Ascites
- [ ] Diastasis Recti
- [ ] Umbelical Hernia

Body Involvement
- ○ Face
- ○ Face_Trunc
- ○ Face_Trunc_Limb
- ○ Face_Trunc_Limb_Sole
- ○ None

<<Back Forward>>

KID: A Pediatric Adviser CASE: Jaundice

Mother's blood Group
- ○ A
- ◉ O
- ○ B
- ○ AB
- ○ Not Known

Mother's Rh Facto
- ◉ Positive
- ○ Negative
- ○ Not Known

Baby's Blood Group
- ◉ A
- ○ B
- ○ O
- ○ AB
- ○ Unknown

Baby_Rh Facto
- ○ Positive
- ○ Negative
- ◉ Not Known

Skin
- [x] Normal Colour
- [ ] Cyanosis
- [ ] Acrocyanosis
- [ ] Mottling
- [ ] Harlequin Colour Change
- [ ] Ichterus
- [ ] Pallor
- [ ] Capillary Haemangioma
- [ ] Mongolian Spot
- [ ] Tuft Of Hair At LS Spine
- [ ] Parchment Skin
- [ ] Erythema Toxicum
- [ ] Pustular Melanosis
- [ ] Vesicular Erruption
- [ ] Perpuric Rash Or Ecchymosis
- [ ] Cavernous Haemangioma
- [ ] Apliasia Cutis Congenita
- [ ] Goltz Syndrome
- [ ] Preauricular sinus and Pit
- [ ] Rash
- [ ] Abnormal Diasthesis

Anaemia
- ○ Mild
- ○ Moderate
- ○ Severe
- ○ None

Serum Bilirubin
- ○ Half mg per hour
- ○ Eighteen mg or m
- ◉ Not Known

Age of the Baby
- ◉ Within One Day
- ○ Within Two Days
- ○ Within Three Days
- ○ Within Three to Five Days
- ○ Within Three to Seven Day
- ○ Within Seven Days
- ○ Within Ten Days
- ○ Within Fifteen Days

<<Back Forward>>

KID: A Pediatric Adviser CASE: Jaundice

About Mother

- ☐ Diabetes
- ☐ Hypertension
- ☐ Syphilis
- ☐ Jaundice
- ☐ Drugs Taken
- ☐ Toxemia Of Pregnancy
- ☐ Rash
- ☐ Fever
- ☐ Adenopathy
- ☐ Arthritis
- ☐ Viral Illness
- ☐ Fetal Loss
- ☒ None

Coombs Test

- ○ Positive
- ○ Negative
- ◉ Not Known

<<Back | Forward

KID: A Pediatric Adviser CASE: Jaundice

Differential Diagnosis		
1. Rh Incompatibility	CF=	0
2. ABO Incompatibility	CF=	70
3. Septicaemia	CF=	0
4. Intra-Uterine Infection	CF=	0
5. Cytomegalo	CF=	0
6. Toxoplasmosis	CF=	40
7. Familial Non-Haemo	CF=	0
8. Dubin Johnson	CF=	0
9. Biliary Atresia	CF=	0
10. Neonatal Hepatitis	CF=	0
11. Heriditory Spherocysis	CF=	0
12. Drug Indu Haemo	CF=	0
13. Galactocaemia	CF=	0
14. Inspissated Bile Syndrome	CF=	0
15. Cretinism	CF=	0

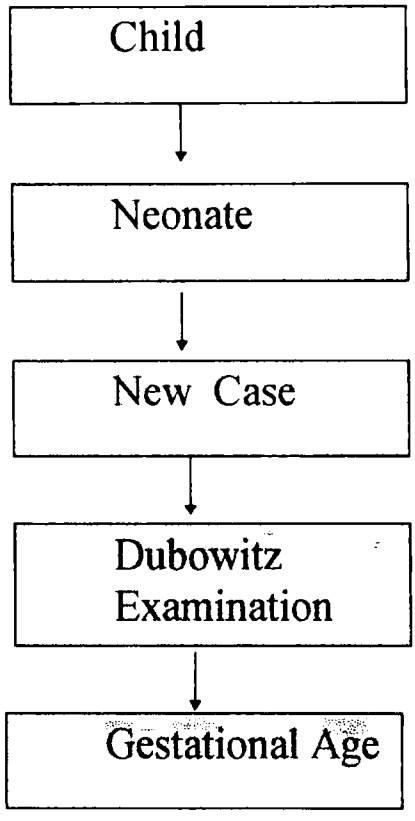
ADVICE

Phototherapy and/or Exchange Transfusion may be required depending on the following factors:

1. Age of the baby in hours;
2. Cord Bilirubin level;
3. Rise of Serum Bilirubin ;
4. Presence of Kernicterus;
5. Serum Bilirubin 20 mg% in full term;
6. Blood Haemoglobin < 10 gm%
7. Reticulocyte Count; and
8. Normoblast Count.

<<Back | Preset

CASE B


Child
Neonate
New Case
Dubowitz Examination
Gestational Age

KID: A Paediatric Adviser

Neonate's Information

IUGR TRUE

Neonate Sex
- Male
- Female

Mother's Name Sefali Sarkar

LMP 26/11/98 dd/mm/yy

Mother's Age 28 Years Parity

Address W/O Pintu Sarkar

Weeks

About Mother
- Diabetes
- Hypertension
- Syphilis
- Jaundice
- Drugs Taken
- Toxemia Of Pregn
- Rash
- Fever
- Adenopathy
- Arthritis
- Viral Illness
- Fetal Loss
- None

Blood Group Of Mother
- A
- O
- B
- AB
- Not Known

Rh Factor
- Positive
- Negative
- Not Known

DOD 31/08/99 dd/mm/yy Gestation Term

Type Of Delivery
- Normal
- Low Forceps
- High Forceps
- LUCS
- Ventouse

After Birth Cry
- Cried
- Not Cried

Resuscitation
- Not Required
- Oral Suction _Manual Sl
- Oral Suction _Injection
- Endotracheal Intubation

APGAR Score

Gest Age <<Back Forward>>

KID: A Paediatric Adviser

Gestational Age Estimation

Edema
- S0 Edema On Feet Hands
- S1 Edema On Tibia
- S2 No Edema

Skin Texture
- S0 Very Thin Gelatinous
- S1 Thin And Smooth
- S2 Medium Thickness
- S3 Slight Thick With Crack
- S4 Thick Perchment Deep

Plantar Creases
- S0 No Skin Creases
- S1 Faint Red Marks On Ant Half
- S2 Definite Red Marks More Than Half
- S3 Indentations Over Ant Third
- S4 Deep Indentations More Than Ant Third

Skin Colour
- S0 Dark Red
- S1 Uniformly Pink
- S2 Pale Pink Variable Over Body
- S3 Pale: Pink Only Over Ears Or Lips Or Palm Or Soles

Ear Form
- S0 Pinna Flat No Incurving
- S1 Partial Incurving of Edge
- S2 Partial Incurving of Whole Upper Pinna
- S3 Well Defined Incurving

Laungo Hairs
- S0 No Laungo
- S1 Long And Thick Over Whc
- S2 Hair Thining
- S3 Small Amount Of Laungs
- S4 Half Of Back Devoid Of Lı

<<Back Forward>>

KID: A Paediatric Adviser

Gestation Age Estimation

Skin Opacity Over Trunk

- S0 Numerus Veins Venules Over Abdomen
- S1 Veins And Tributaries Seen
- S2 Few Large Vessels Seen Over Abdomen
- S3 Few Large Vessels Indistinctly Over Abdo
- S4 No Blood Vessels Seen

Nipple Formation

- S0 Nipple Barely Visible No Areola
- S1 Nipple Well Defined Areola less 7 m
- S2 Areola Stippled Diameter Less 7mm
- S3 Areola Stippled Dia More 7mm

Breast Size

- S0 No Breast Tissue Palpable
- S1 Breast Tissue Less 5mm
- S2 Breast Tissue More 5mm Less 10mm
- S3 Breast Tissue More 10mm Both Sides

Ear Firmness

- S0 Pinna Soft No Recoil
- S1 Pinna Soft Slow Recoil
- S2 Ready Recoil Cratilage Illform
- S3 Instant Recoil With Firm Pinna

Genitalia Male

- S0 Neither Testis In Scrotum
- S1 One Testis High In Scrotum
- S2 One Testis Down In Scrotum

Genitalia Female

- S0 Labia Major a widely Separated
- S1 Labia Minora Partially Covered
- S2 Labia Minora Completely Covered

<<Back Forward>>

KID: A Paediatric Adviser

Gestational Age Estimation

Posture

- S0 Posture0
- S1 Posture1
- S2 Posture2
- S3 Posture3
- S4 Posture4

Leg Recoil

- S0 Recoil 180
- S1 Recoil 90 To 180
- S2 Recoil Less 90

Popliteal Angle

- S0 Pop Angl
- S1 Pop Angl
- S2 Pop Angl
- S3 Pop Angl
- S4 Pop Angl
- S5 Pop Angl

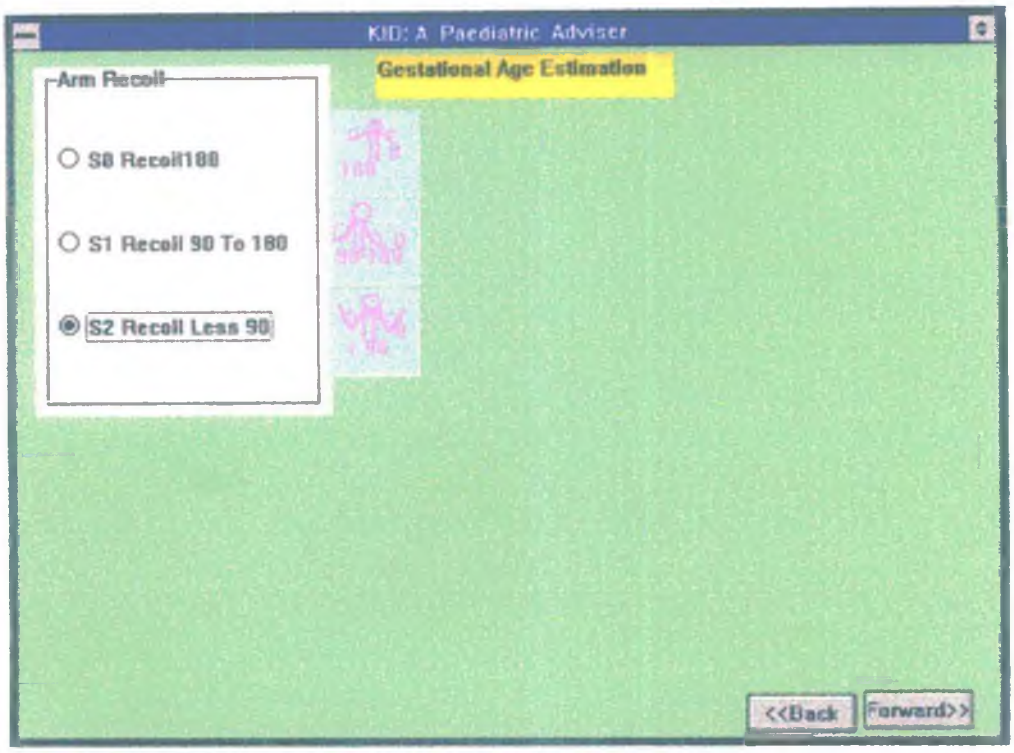
Square window

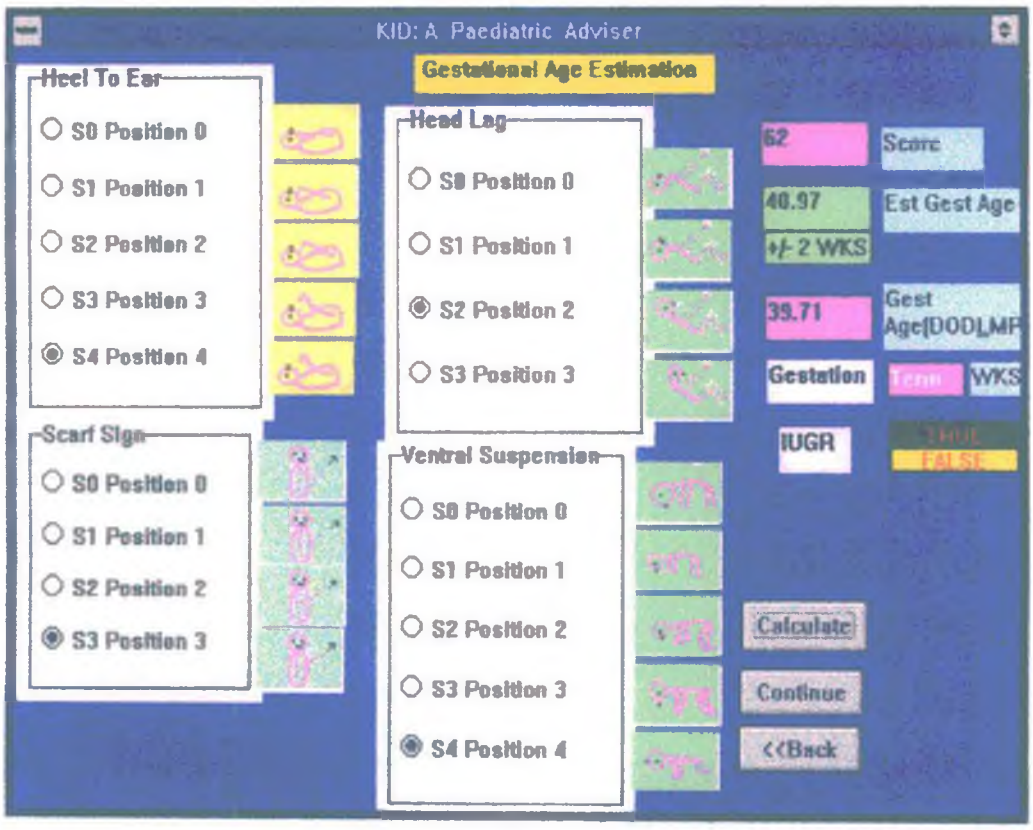
- S0 Sq Win90
- S1 Sq Win60
- S2 Sq Win45
- S3 Sq Win30
- S4 Sq Win 0

Ankle Dorsiflexion

- S0 Dorsiflexion90
- S1 Dorsiflexion75
- S2 Dorsiflexion45
- S3 Dorsiflexion20
- S4 Dorsiflexion0

<<Back Forward>>


KID: A Paediatric Adviser
Gestational Age Estimation
Arm Recoil
S0 Recoil180
S1 Recoil 90 To 180
S2 Recoil Less 90
<<Back
Forward>>


KID: A Paediatric Adviser
Gestational Age Estimation
Heel To Ear
S0 Position 0
S1 Position 1
S2 Position 2
S3 Position 3
S4 Position 4
Scarf Sign
S0 Position 0
S1 Position 1
S2 Position 2
S3 Position 3
Head Lag
S0 Position 0
S1 Position 1
S2 Position 2
S3 Position 3
Ventral Suspension
S0 Position 0
S1 Position 1
S2 Position 2
S3 Position 3
S4 Position 4
52
Score
40.97
Est Gest Age
+/- 2 WKS
39.71
Gest Age(DOD,LMP
Gestation
Term
WKS
IUGR
FALSE
Calculate
Continue
<<Back

List of Publications

LIST OF PUBLICATIONS OF M. G. GOSWAMI

1. M. G. Goswami, M. Chatterjee, and R. K. Samanta, "Fetal Growth Performance for North-Bengal Districts - Knowledge Acquisition through Ultrasonographic Imaging", Proc. ERC - 95, Computer Society of India, 71-76.

2. A. K. Saha, M. G. Goswami, M. Chatterjee and R. K. Samanta, "Child Growth and Development, Expert Systems and Inexact Information Management", Proc. CSI-95, Hyderabad, India, Tata Mc-GrawHill, 258-267.

3. R. K. Samanta, A. K. Saha, M. G. Goswami and M. Chatterjee, "On the Development of a Consultation System for a paediatric growth Performance of North-Bengal districts of India", Proc. National Conf. on Software Engineering and its Applications, 22-24 Aug.'96, Inst. Pub. Enterprise, Hyderabad, 127-134.

4. M. Chatterjee, A. K. Saha, M. G. Goswami, and R. K. Samanta, "Artificial intelligence and expert systems technology : its role in medicine", CONVERSE (Themis Pharmaceuticals, Bombay), vol.1, no.2, 13-22, 1996.

5. R. K. Samanta, A. K. Saha, M. G. Goswami and M. Chatterjee, "Selecting an appropriate expert system tool : from practitioner's floor", CSI Communications, Vol.20, no.7, January, 1997, 24-32; Vol.20, no.8; February, 1997, 19-22.

6. R. K. Samanta, A. K. Saha, M. G. Goswami and M. Chatterjee, "Artificial Intelligence and Paediatrics", 3rd. Nat. Conf. of IAP, 16-18 Feb, 1997, Ranchi, India.

7. A. K. Saha, M. G. Goswami, M. Chatterjee, D. Dutta Majumder and R. K. Samanta, "A Paediatric Expert System for North-Bengal districts of India", CSI Communications, May 1997, 15-21; June 1997, 21-25.

8. R. K. Samanta, A. K. Saha, M. G. Goswami and M. Chatterjee, " KID (Ver 1.0) : An Expert System for Managing Neonates", Proc. 3rd. International Conf. on Modelling and Simulation, Melbourne, Australia, Oct. 29-31, 1997, 151-156.

9. A. K. Saha, Mridula Chatterjee, M. G. Goswami and R. K. Samanta, "A Fuzzy, Knowledge Based Neonatal Resuscitation Management System", 3rd. Int. Conf. on Cognitive Systems (ICCS'97), Dec 13-15th, 1997, Delhi, 627 - 636.

10. M. G. Goswami, A. K. Saha, Mridula Chatterjee and R. K. Samanta, " Fetal Growth Performance of North Bengal Districts using Ultrasonographic technique", Abstract, 41st All India Conf. on Obst. & Gynocology, Dec,1997, North Bengal Medical College, Siliguri, 95- 96.

11. M. G. Goswami, A. K. Saha, Mridula Chatterjee and R. K. Samanta, "A study on fetal growth performance of North-Bengal districts using ultrasonographic technique", Abstract, Fifth West Bengal State Science Congress, March 21-23, 1998. University of North Bengal, 128.

12. A. K. Saha, M. G. Goswami, Mridula Chatterjee and R. K. Samanta, "An expert system approach to child care management for North-Bengal districts", Abstract, Fifth West Bengal State Science Congress, March 21-23, 1998. University of North Bengal, 125.

13. R. K. Samanta , A. K. Saha, M. G. Goswami, and Mridula Chatterjee, " On Fuzzy Concepts in Paediatrics within Expert Systems Framework", VIVEK, NCST, Bombay, vo1.11, No. 1, January, 1998, 3 - 15.

14. Mridula Chatterjee, A. K. Saha, M. G. Goswami, and R. K. Samanta, "A Study on Fetal Growth Performance", Proc. of XXIInd International Congress of Paediatrics, Amsterdam, The Netherlands, 9-14 August, 1998, P091.

15. A. K. Saha, M. G. Goswami, M. Chatterjee and R.K. Samanta, "Potential Issues in the Development of KID : A Consultation System for Child Growth and Development of North Bengal Districts of India", Modelling, Measurement & Control, D, vol. 17, no. 1, AMSE press, France, 1998, 1 - 19.

16. M. G. Goswami, A. K. Saha, Mridula Chatterjee and R. K. Samanta, "A RBR Augmented CBR Approach to a Medical Consultation System" , Proc. Nat. Conf. on Medical Informatics, 6-7 April, 1999, Hyderabad, 99 - 107.

17. M. G. Goswami, A. K. Saha, Mridula Chatterjee and R. K. Samanta, " Developing Medical Expert Consultation Systems using Web-Based Knowledge Acquisition Approach", Eastern Regional Conference-1999, Computer Society of India, Calcutta, June 11-12, 1999.

18. M. G. Goswami, A. K. Saha, Mridula Chatterjee and R. K. Samanta, " Advances in Medical Informatics and the Future", Int. Conf. on Computer Modelling, Simulation and Communication (CMSC-99), Jaipur, India, 1-3 December, 1999 (to appear).