Original Article

An Intelligent Tutoring System for Trauma Management (Trauma-Teach): A Preliminary Report

LS Ong,[1] *BSc (Comp Sc), MSc (Comp Sc)*, A Vijayan,[2] *FAMS, FRCS (Edin), FRCS (Glas)*, CS Koh,[1] *BEng MTechKE*, CC Lai,[1] *BSc, MTechKE*, CW Lim,[1] *BSocSc*, WF Loke,[1] *BBA, MTechKE*, SH Low,[1] *BBA, MTechKE*, KY Tang,[1] *BEngMech*, FL Wong,[1] *BEng (EEE), MTechKE*, KL Yong,[1] *BAppSc (Comp Eng)*

Abstract

Trauma-Teach is an interactive software for tutoring surgical trainees on medical trauma management procedures. Users of the system interact with a virtual patient suffering from trauma injuries. The task of the user is to stabilise the virtual patient, discover the underlying injuries and decide on an appropriate management plan. Artificial intelligence techniques are used to simulate the patient's pulmonary and cardiovascular systems in real time, determine the responses and results of treatments and diagnostics accordingly, model the patient deterioration if wrong actions are taken, and give a measure of reality to the system by selecting actual trauma cases from the hospital's database.

Ann Acad Med Singapore 2005;34:499-504

Key words: Artificial intelligence, Computer-aided, Decision making, Training

Introduction

In Singapore, trauma[1-7] is the leading cause of hospitalisation and the fifth leading cause of morbidity and mortality.[8,9] Common causes of trauma include road traffic accidents, industrial accidents, falls and recreational activities. In treating the trauma patient, it is essential that the processes of resuscitation and the identification of life-threatening injuries are carried out rapidly and accurately so that the injuries can be managed in a time-sensitive manner. Adopting the Advanced Trauma Life Support protocol and subsequent diagnostic and therapeutic adjuncts, a management plan should be arrived at quickly, preferably within the first 15 to 45 minutes of the patient's arrival at the emergency department of the hospital. Knowledge of the pathophysiology of trauma, experience in the varied presentations of the multiply injured patient and the application of critical therapeutic procedures are crucial to this rapid decision making process and a successful outcome in managing such a patient. Teaching this decision making and management process in a manner that is safe to the patient yet appropriate to the training needs of the surgical trainee is difficult as time is short and there is no room for error in such situations. In order to overcome these shortcomings, several computer-aided instruction (CAI) packages or computer programmes that simulate trauma scenarios have been introduced.[10,11] The drawback of these packages is that they generally lack the realism and variability seen in the clinical presentations of the trauma patients. As the scenarios are fixed in these programmes, repeated use leads to familiarity and training effectiveness is lost.

This is where Trauma-Teach, an intelligent tutoring system in trauma management, comes into play. Trauma-Teach utilises artificial intelligence (AI) techniques to build a tutoring system that aims to be more intelligent than the conventional CAI or systems comprising multiple-choice questions that tend to become non-challenging as a result of their predictability. Instead of the passive reading of texts (occasionally peppered with pictures) or multiple-choice questions that surgical trainees can memorise, this tutoring system provides intelligent interaction, realism and appropriate guidance to the user. Learning is not just based on their ability to recall facts but the correct application of knowledge and simulated diagnostic and therapeutic

---

[1] Institute of Systems Science
National University of Singapore, Singapore
[2] Department of General Surgery
Tan Tock Seng Hospital, Singapore
Address for Reprints: Ms Ong Lean Suan, Programme Director (Knowledge Engineering), Institute of Systems Science, National University of Singapore, 25 Heng Mui Keng Terrace, Singapore 119615.
Email: leansuan@iss.nus.edu.sg

manoeuvres in an active, situational learning environment. This allows the users to retain and apply their knowledge and skills more effectively according to real-life scenarios. The system also provides a summary of the actions taken so that the trainee can learn from both appropriate and inappropriate decisions.

This paper describes the highlights of the system, developed by students taking the Master of Technology in Knowledge Engineering offered by the Institute of Systems Science (ISS), National University of Singapore, in collaboration with doctors from Tan Tock Seng Hospital (TTSH) for use by surgical trainees. Copyright for the software is currently being processed by the relevant administrators of the hospital.

Scope of the Project

Trauma-Teach focuses on the effects and the subsequent management of blunt trauma to the torso with its attendant injuries to the liver, spleen and hollow viscus. Figure 1 shows the scope of the project which covers only blunt abdominal trauma; the vital signs considered are blood pressure, heart rate, oxygen saturation and respiratory rate. The system session starts with the creation of a scenario (patient's initial parameters include injuries sustained and other vital signs). The selection of the initial parameters can be carried out by the learner or randomly generated based on the lesson objective. The system keeps a library of anonymous past cases seen by the hospital and randomly generates (from this library) a mechanism of injury, the injuries inflicted on the victim and the severity of each injury. From the generated injuries and their severity, the system evaluates and simulates the victim's condition over a period of time.

As the session progresses, the condition of the patient may improve or deteriorate, depending on whether the appropriate action is taken. Without the learner's intervention, the patient's condition deteriorates in line with the type and severity of the injuries, going into shock and ultimately death. This added "time pressure or urgency" is applied to the learner to simulate a real-life scenario whereby a decision must be made quickly. Whenever the learner performs an action, the system evaluates whether the action has a diagnostic or therapeutic value and alters the patient's condition accordingly. The outcome, such as the vital signs, is shown through various graphical images. The learner's performance is assessed and given a score, based on whether all of the required treatment protocols were carried out within the allocated time.

Finally, as any good human tutor would do, explanations relevant to the training session are given, and the outcome of the actual patient case is displayed to remind the learner of the real world, thus enriching the learning experience of the system user.

Design of the System

Figure 2 shows the main functions or components of the system, namely scenario creation, patient modelling and scoring.

Scenario Creation

This starts the training session. The patient's initial parameters, such as the mechanism of injury, place of injury (or injury site) and information about the cause of injury, will be retrieved from a case base of trauma patient records. The case base stores various historical patients' data including demographic information, mechanism of injuries, injuries sustained and other relevant case information. The selection of the patient's initial parameters and the information about the injuries sustained can be selected by the user or randomly generated by the system. Upon initialisation of the parameters, the severity of each injury will be used for simulating the vital signs. For example, a patient with 2 minor injuries might have little or no change in initial blood pressure while the heart rate might be elevated.

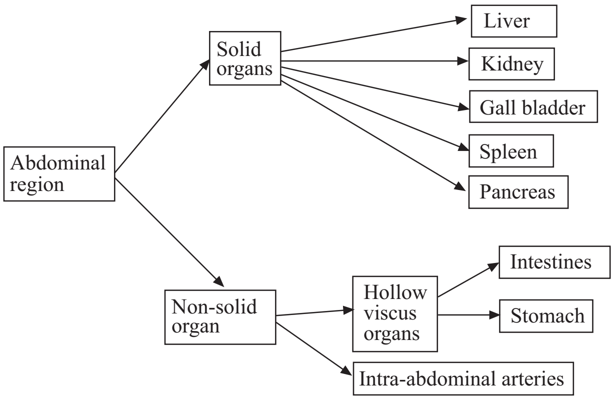


Fig. 1. Scope of the project covering blunt abdominal trauma.

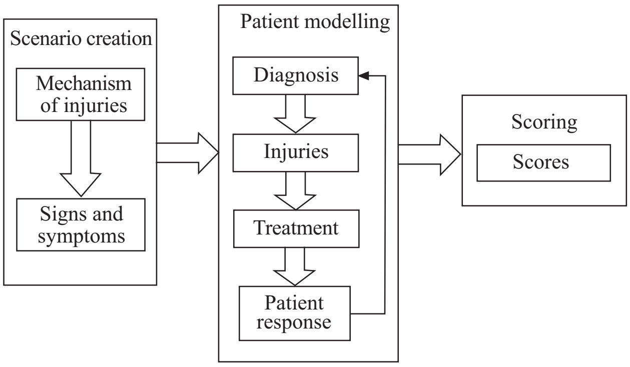


Fig. 2. Overview of Trauma-Teach process flow.

Patient Modelling

The patient modelling combines 2 processes: a patient deterioration process and the patient response process. Patient deterioration models the deterioration of the patient's vital signs due to the sustained injuries, taking into account the passage of time. Injuries will result in continual blood loss, inhibit respiratory functions and damaged internal organs. The patient response process models the effects or outcome of the doctor's actions on the patient. Diagnostic actions will reveal the patient's conditions to the doctor in greater detail. For example, physical examination will reveal bruising, and x-rays will reveal fractures. Therapeutic actions will be applied to stabilise the patient's condition. For example, the infusion of fluids will restore blood volume and may improve the pulse rate. The effects of inappropriate actions could range from no effect to further aggravation of the patient's conditions, and may even result in the death of the patient.

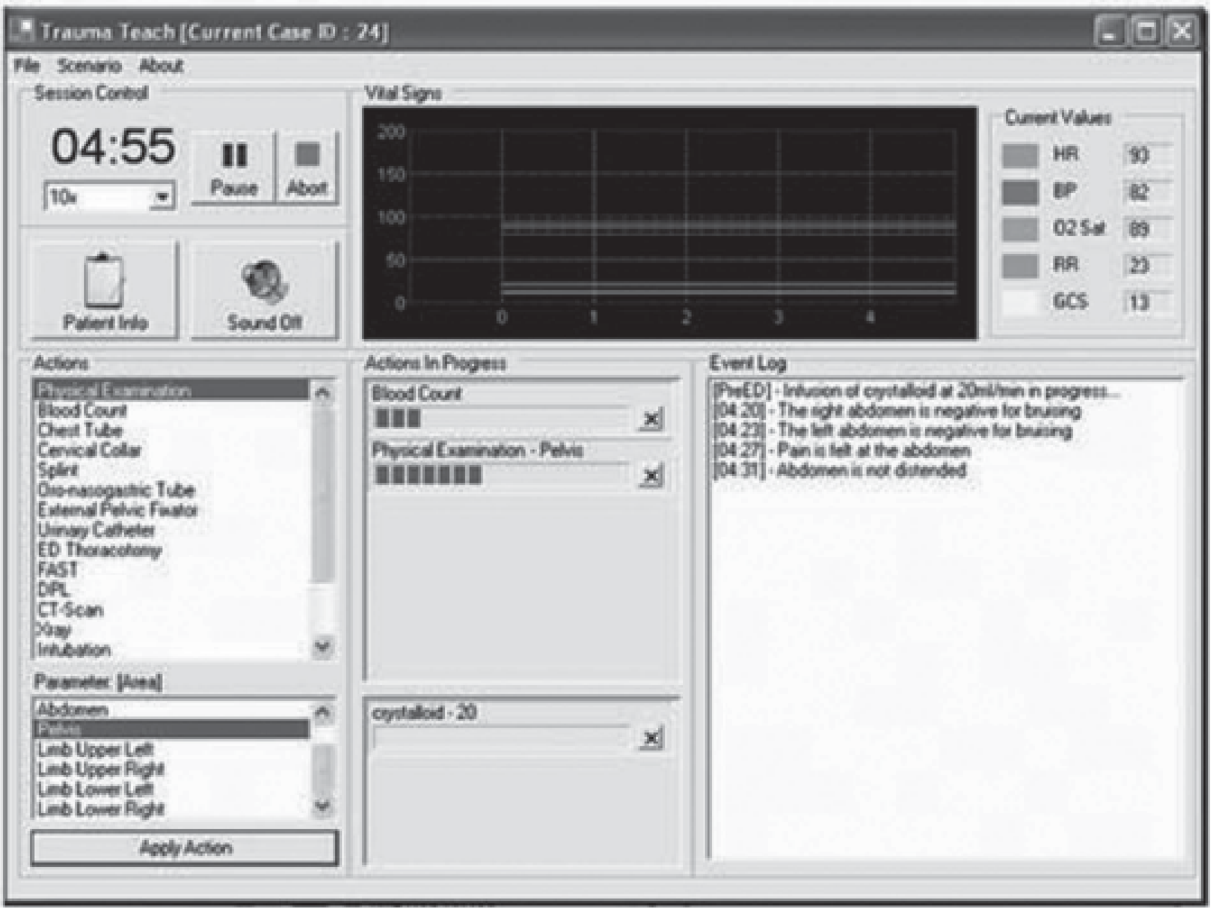
Scoring

This is performed at the end of the session. The scoring component evaluates the user in terms of whether he knows and applies the correct procedures for the given scenario, within the allocated time and following the correct protocols in making the correct post-emergency department decision. A simple score on a scale of "A" to "F" (fail) will be computed to reflect the appropriateness of the user's actions in response to the injuries sustained by the simulated atient.

In order to compute the scoring, a list of correct solutions to a patient model is retrieved from the patient case base. The objective of the list is to provide the scoring inference engine with a common reference when assessing the user's performance. The process of selecting the solution from the list may be random or iterated through the set in the list. The inference engine then compares the action taken by the user with the diagnostic and treatment solution from the list. Scoring rules are used to compute the final overall score. The score will be higher if the user has performed more correct actions and appropriate decisions.

User Interaction

Figure 3 is an example of the user interface of the system, and shows a case scenario. The simulation will proceed according to the elapsed time shown (in the top left corner).



HR 90
BP 82
O2 Sat 89
RR 23
GCS 13

```
[PreED] - Infusion of crystalloid at 20ml/min in progress...
[04:20] - The right abdomen is negative for bruising
[04:23] - The left abdomen is negative for bruising
[04:27] - Pain is felt at the abdomen
[04:31] - Abdomen is not distended
```

Fig. 3. The Trauma-Teach user interface.

As time passes, the vital signs will be updated automatically. At this point, the user will have to make a decision. The user can choose an action from a list of common clinical actions with their respective parameters. On selecting and applying an action, the user will monitor the process until completion through the progress bar shown in the middle of the screen. An action in progress may be aborted or left to completion. During the execution of an action, built-in constraints restrict the user from performing other irrelevant actions at the same time. For example, while waiting for the result of a computed tomography (CT) scan, an x-ray cannot be performed. The result of each action will be shown in sequential order in the Event Log. Depending on the type of actions performed, the simulated patient will respond accordingly. The effect ranges from negative to none to positive. The final decision to be made by the user would be to redirect the patient either to a ward or an operating theatre.

System Implementation

There were many challenges facing the ISS students in the development of this system. The depth and breadth of medical knowledge[12] (the human body is very complex and detailed, ranging from physical level to genetic level), uncertainty of data (how accurate is this test result?), missing information (vital to diagnosis or treatment, but not yet available or impossible to obtain) and the final interaction between all these various issues, build up to a very complex system.

The fundamental AI techniques used in this project were case-based reasoning (CBR),[13] rule-based reasoning (RBR)[14] and fuzzy logic.[15,16] CBR was used to add a measure of reality to the scenarios by retrieving actual patient cases from the library of cases. For patient initialisation, a similarity (or closeness) formula was used to calculate the overall similarity of each case in the case base with respect to the selected initialisation parameters. All the cases were then ranked according to the closeness of the match. The highest ranked case was then selected.

RBR was used to represent and search through medical knowledge and pedagogical know-how. For patient response, RBR was used to search through the rule base to infer the response to each user action. The conclusion generated from the rules determines the outcome. The rule inferencing engine uses a forward-chaining or data-driven approach to compare facts asserted in the working memory against the conditions (*If* component) of the rules in the rule base to determine the next applicable rule to fire. When a rule is fired, the conclusion of that rule is used to infer the next rule conclusion, and so on (chaining of rules). The example below shows a fact (describing an injury) being asserted. Similar facts can also be asserted for patients with multiple injuries.

```
(injury_description  (injury_desc  spleen_laceration)(injury_grade  4))
```

Once the fact(s) have been asserted, rules will then infer the injury type and organ type. At the same time, other information will be inferred as well, including the injury grade.

```
;; rule 1 - Spleen with Contusion
(defrule  spleen_contusion
    (declare (salience 10))
    (injury_description  (injury_desc  spleen_contusion)
(injury_grade?                severity))
    =>
    (assert (injury (injured_organ spleen)(injury_type  contusion)
    (injury_type_CF  1.0)(injury_grade  ?severity)))
    (onReturnAction "spleen with contusion with Injury Grade")
)
```

Similar rules like the example above (spleen with contusion) were used to determine the appropriate response to an action taken by the system user.

The automatic response of the human body to injury may be equated to the functioning of a preprogrammed water heater. When there is a significant loss of blood, the veins will contract. Similarly, to maintain the preset temperature, the water heater will lower or raise the temperature by adjusting the heating element. The water heater is functioning as a controller. Such a controller may be implemented using fuzzy logic. In Trauma-Teach, fuzzy logic was used to manipulate variables (vital signs) such as blood pressure, heart and pulse rate, thus simulating minute-by-minute changes in the vital signs of the patient. Based on the patient's current condition (injury severity), each vital sign was adjusted to reflect the deterioration as time passes. Fuzzy logic allows overlapping or vague concepts to be processed and overcomes limitations such as lack of information (without which "normal" processing could not be carried out). For example, in evaluating the severity of an injury, a grade 3 could be interpreted as "moderate" for one doctor but "critical" for another. In this case, the fuzzy rules corresponding to both levels of severity will be executed and aggregated based on the level of certainty of each rule. In addition to allowing overlapping concepts, fuzzy logic also enables the exploration of rare conditions that might not have occurred before. The user is thus able to study and share his or her various observations with others.

Testing: Validation and Verification

Validation and verification processes are critical in determining whether the system meets the user specifications and its output is correct. Validation determines if the completed expert system performs the functions in the requirement specifications and serves its intended purpose. Verification ascertains that the expert system correctly implements the requirement specifications and all modules are integrated and perform well i.e., the Trauma-Teach system is built right. The individual modules were tested to ensure that they performed according to the specifications. Subsequently, the system was tested as a whole, to ensure that all the modules were properly integrated and achieved the level of correctness and completeness expected by the user.

Testing of Scoring Module

Three possible scenarios were defined:

Scenario 1: Perform all critical actions within the time frame, and check that the module gives a high grade.

Scenario 2: Perform all critical actions but exceed the time frame, and check that the module gives a lower or failed grade, depending on the grade conditions stated below:

- Grade A
  (Minimum Time) to (Minimum Time + 25%)
- Grade B
  (Minimum Time + 26%) to (Minimum Time + 50%)
- Grade C
  (Minimum Time + 51%) to (Minimum Time + 75%)
- Grade D
  (Minimum Time + 76%) to (Minimum Time + 100%)
- Grade F
  Greater than (Minimum Time + 100%)

Scenario 3: Perform only some critical actions within the time frame, and check that the module gives a grade F.

Based on a particular test scenario, bearing in mind the critical actions that must be performed specific to that injury type, various diagnostic and treatment actions were performed. The results at the end of the test should tally with the expected results from the scoring algorithm. For example, in scenario 2 (Grade B) and injury (spleen laceration grade 3a), the critical actions that must be performed for this particular injury include: injecting crystalloid and blood, carrying out a physical examination and a CT scan of the abdomen, doing a blood count, and taking a chest x-ray. To calculate the minimum time required for all the important actions, a table storing the minimum time for each action is referenced and added to get the total time in minutes. Assuming that this total time for scenario 2 is 44 minutes, taking the duration of 55 to 66 minutes [(Minimum Time + 26%) to (Minimum Time + 50%) = (44 + 26% * 44) to (44 + 50% * 44)] results in a Grade B.

Table 1. Functionality Feedback and Analysis of 5 Major System Modules

System module	Test result	Analysis
Patient initialisation	Have more parameters for case selection, for example, to use injury grade to select case.	Current design sufficient for a small case base. However, the case representation and retrieval algorithm will be modified to include more parameters for case selection.
Patient deterioration	To include more "realistic" fluctuations in the vital signs, so as to more accurately simulate actual situations.	As there was no empirical time-series data available for vital signs, limited data were fabricated with the help of the doctor prior to the live user test. More fluctuation in vital sign representation can be achieved by refining granulation in the fuzzy sets for each vital sign.
Patient response	The processing time for CT scan and blood count is preventing other actions from being carried out. Modify the processing time for certain actions to make the system more "real", for instance, to shorten the processing time for x-ray.	The current design combines both execution and processing time of CT scan or blood count into one. A more appropriate design would be to split it into 2 time durations so that other actions can be performed while the the previous action is still being processed.
Scoring	Doctor stated that it was not necessary to include blood count as one of the critical criteria in scoring. Final decision was one of the criteria used for scoring. Need to fine-tune the final decision criteria of sending patients to high dependency ward or intensive care unit.	Blood count as a critical action criterion was formulated during knowledge acquisition. This has been changed according to the feedback given. This feature will be incorporated once more information is available.
Graphical user interface	Interface to include the amount of blood or crystalloid (fluids) given to the patient at any particular point of time. To have more detailed images showing results of actions. For example, to use abdominal quadrant to show the injury region instead of text.	This will be included as future system enhancements. Medical images will be added under future system enhancements.

CT: computed tomography

Usability Testing Process

Finally, usability testing with an experienced doctor from TTSH was carried out using different test scenarios (for example, different injury types and severity grades for spleen, liver and hollow viscus injuries). The doctor was asked to give feedback on the overall look and feel of the user interface, appropriateness of the information displayed, such as changes in vital signs and actions lists, intuitiveness of the system and finally, the overall user satisfaction in using the system.

Discussion of Test Results

Before each test scenario, the doctor was asked to provide the expected actions he would perform with respect to the given test scenario. After the test, an analysis was performed to study the implication behind the difference between the expected actions and the actual actions that were taken using the system.

For example, in test scenario 1, the doctor stated 6 actions that he would perform in a real situation, but he only performed 4 actions using the system. Blood count was not performed, and the ATLS primary survey (airway, breathing and circulation) was also not done as these actions were not included in the list of possible user actions. For the subsequent test scenarios, the doctor performed all expected actions. Table 1 shows the feedback and analysis for each major function of the system.

The doctor rated the effectiveness of the user interface and the ease of use of the system highly, with both evaluation criteria scoring above 70%. He was also highly satisfied with the system performance, giving it an extremely high score of 90%. However, he gave a score of about 55% with respect to the realism depicted by the system. He felt that more could be done to enhance the realism of the system. The doctor gave the system an overall score of 69%. After the preliminary testing by the experienced doctor, the software is currently being field-tested (formally) by junior doctors and medical students. Based on their feedback, we will be working on the next version of Trauma-Teach and will subsequently publish the results.

Conclusion

Computer systems that emulate human intelligence[17] (for example, in diagnosing diseases or prescribing treatment) are not actively used in day-to-day healthcare despite the potential benefits, such as relieving the information burden of doctors. Much of this field is still under research, and useful commercial products are still not widely available. However, medical expert systems are now being seen as a viable teaching aid and in supporting medical research.

A tutoring system such as Trauma-Teach can provide a low-cost environment for learning, anytime and anywhere, with a virtual tutor to evaluate a learner's performance, guide and provide feedback without the dire consequences of mistakes. This translates into benefits for patients in terms of improved patient care, greater access to doctors with specialised training and optimised time management. The survivability of patients increases as doctors become more experienced in handling any scenario.

With further work, such as adding more multimedia or graphics and extending the problem domain to cover other types and causes of injuries, Trauma-Teach is a potentially useful tool for doctors.

REFERENCES

1. Driscoll P, Skinner D, Earlam R. ABC of Major Trauma. 3rd ed. London: BMJ Books, 2000.
2. Goris RJA, Trentz O. The Integrated Approach to Trauma Care (The First 24 Hours). Berlin, Heidelberg: Springer-Verlag, 1995.
3. Ferrera PC, Colucciello SA, Marx J, Verdile V. Trauma Management: An Emergency Medicine Approach. New York, USA: C.V. Mosby, 2000.
4. Alexander RH, Procter HJ. Advanced Trauma Life Support Course for Physicians. Chicago, IL: American College of Surgeons, 1995.
5. Online resources for trauma management. Available at: http://www.trauma.org/resources/index.html. Accessed October 2004.
6. David Baldwin's Trauma Information Pages. Available at: http://www.trauma-pages.com. Accessed May 2005.
7. Online Medical Textbooks and Physician Reference Articles. Available at: http://www.emedicine.com/med/TRAUMA.htm. Accessed October 2004.
8. Ministry of Health. Health Facts Singapore 2004, Principal Causes of Death. Available at: http://www.moh.gov.sg/corp/publications/statistics/principal.do. Accessed May 2005.
9. Ministry of Health. Health Facts Singapore 2004, Top 10 Conditions of Hospitalisation. Available at: http://www.moh.gov.sg/corp/publications/statistics/top10.do. Accessed May 2005.
10. Kaye J. Understanding TraumaAID, 1994. Available at: http://www.cis.upenn.edu/~traumaid/TechDoc/techdoc.htm. Accessed March 2005.
11. Clarke JR, Hayward CZ, Santora TA, Wagner DK, Webber BL. Computer-generated trauma management plans: comparison with actual care. World J Surg 2002; 26:536-8.
12. Tortora GJ, Grabowski SR. Principles of Anatomy and Physiology. 8th ed. New York, USA: Biological Sciences Textbooks, Inc., 1996.
13. Kolodner J. Case-based Reasoning. Mateo, CA: Morgan Kaufmann, 1993.
14. CLIPS – C Language Integrated Production System. Available at: http://www.ghg.net/clips/CLIPS.html. Accessed February 2004.
15. Orchard RA. FuzzyCLIPS User's Guide Version 6.04A. Ottawa, ON: Institute for Information Technology, National Research Council Canada, October 1998.
16. Fuzzy Extension to the CLIPS Expert System Shell (FuzzyCLIPS). Ottawa, ON: Institute for Information Technology, National Research Council Canada, 2002. Available at: http://iit-iti.nrc-cnrc.gc.ca/projects-projets/fuzzyclips_e.html. Accessed February 2004.
17. Giarratano JC, Riley G. Expert Systems – Principles and Programming. Boston, MA: PWS Series in Computer Science, 1994.