Research Journal of Information Technology 3(1): 55-67, 2011
ISSN: 2041-3114

Received: December 17, 2010 Accepted: January 10, 2011 Published: March 20, 2011

A Web-based Visual Simulator for Hospital Management Using Discrete Event Simulation

[1]G.K. Raikundalia, [2]M. Mastan and [3]C.A. Bain
[1]School of Engineering and Science, Victoria University, Melbourne , Australia [2]Hansen Technologies, Doncaster, Victoria 3108, Australia
[3]Performance Analysis and Data Governance, Strategy & Planning, Ambulance Victoria, Doncaster, Victoria 3108, Australia

Abstract: Access, demand and resource management issues in hospitals continue to plague the area of hospital management. Access is the capacity to provide patients with health care in a timely fashion, demand is a community's need for health care satisfied by available health services and resource management is the use of various resources available to hospitals for managing them. Problems in these three areas lead to long waits in emergency, prolonged periods for patients on procedural waiting lists and cancellation of surgical operations. To support hospital managers in effective management of a hospital, managers require software that presents them with necessary information about resources, such as beds available for patients in emergency. This study presents a new Web-based visual solution to the abovementioned problems using discrete event simulation. This system builds on our previous work where simulations are generated using a Web site, but the Web site generates an enormous quantity of data as output. This output needs to be processed in a very short amount of time (given managers need for quick access to information) and displayed in visual manner for easy use and interpretation by managers. Our software solution is absolutely novel as our earlier work on which the new solution is based was also novel as a solution for hospital managers.

Keywords: Computer simulation, discrete event simulation, hospital management, World Wide Web

INTRODUCTION

Hospitals are inarguably fundamental and essential for the health and wellbeing of a society. Whether children are born, patients recuperate from illness or surgeries are performed there, hospitals are the form of care communities can rely on to assist them with their health. Therefore, it is abundantly clear that their efficiency and effectiveness in helping people is of great importance. In relation to this, the management of such a vital organisation as a hospital must be assisted where possible so that patients' health and wellbeing is maximized as the outcome.

Systems and technologies that can assist hospital managers in their work of handling a major and complex organisation like a hospital are of great need. Managers of hospitals have many issues with which to contend, such as insufficient funding of their activities, lack of resources, efficient use of staff and so forth. In many countries, insufficient funding and unavailability of beds or relevant staff are highly problematic and can to lead long waiting times for treatment or surgery. It is health of the community that suffers under such circumstances.

Our interest lies in the simulation of a hospital so as to assist a hospital manager in the running of a hospital. Simulation tools for various domains have been developed and made available for decades. Simulation has the benefit of "what-if" analysis and gaining an understanding of a problem before implementing a solution to the problem.

In the case of a hospital, a manager is able to execute scenarios of different situations for the hospital to know the effects of management decisions. The manager can view the effects of an increase in the number of beds in a ward or the hospital or if more staff are made available to a ward or unit. The manager may determine that, for instance, increasing the number of beds by two in a particular ward may not have as much desired effect on reducing the waiting time for patients in the Emergency Department. Therefore, the simulator can assist the manager in the prediction of effects from changing various aspects of a scenario, and from this, enable them to better manage the hospital.

Our previous work on hospital simulation has not involved a visual, graphical presentation to the manager (Bain *et al*., 2007; Raikundalia *et al*., 2009). This study has provide a textual (yet Web-based) output that, at the time the work was done, provided a new capability for assisting a hospital manager in their management activity. Clearly, a more user-friendly output is one that is in a

Corresponding Author: G.K. Raikundalia, School of Engineering and Science, Victoria University, Melbourne, Australia

graphical format as described in this study. However, at this time, there is a paucity of such visual simulators for hospital management that go beyond simulation of a hospital department or unit.

This study covers in detail the functionality of our simulator, presenting to the reader the various user interfaces of the system and how they work. To make the application of the simulator abundantly clear, two scenarios exemplifying its use are provided.

LITERATURE REVIEW

Various systems in health informatics or e-health have been developed over the years that provide simulations assisting health care practitioners clinically. We present a survey here of simulation systems that use powerful underlying technology in order to generate simulations. Without advanced computing such as grid computing or parallel computing, such simulations could not be generated. Although in our case data in volumes of hundreds of megabytes need to be processed to generate simulations, these powerful forms of computing are not required. Many other simulation systems have been developed without need for such powerful technology (indeed several systems were developed decades ago before such technology was available); however, we have covered such systems in our other papers (Bain *et al*., 2007; Raikundalia *et al*., 2009) and we have not covered them again here.

Pormann *et al*. (2000) present a simulator generator for support of simulation of wavefront conduction associated with the human heart. Simulations investigate aspects specifically involved in arrhythmias. The researchers use a modular approach so that the user can choose a collection of modules that are the closest in reflecting the simulation that wish to carry out. The simulator is generated from this collection of modules.

The researchers run simulations on an IBM SP parallel computer and a cluster of workstations. They found a relatively small problem that there were speed-ups of 12 on 16 processors and 22 on 32 processors. In the case of a problem that was scaled, they could “simulate 128 times as large a domain on 128 processors with only a 9 processor run” (Pormann *et al*., 2000). Using the greater memory available with parallel computers, very realistic simulations of atria were possible.

Axner (2007) covers a problem solving and simulation environment, HemoSolve, for addressing image-based computational hemodynamics problems. The system, which is completely parallel, is used for visualisation of biomechanical processes in vascular systems. HemoSolve also provides tools to simulate surgery, such as bypass replacement. The researcher develops an initial version and then improves that version with a newer, faster version using a performance prediction model to determine who to make the system work faster.

Axner (2007) presents example improvements in the new version:

- The time taken for time-harmonic blood flow simulation in the carotid artery with bypass placement is about one-ninth faster in new version
- Experiments that took 3 h to execute in the previous version take about 20 min in the new version

The work of Dibble *et al*. (2007) involves simulation of pandemic influenza of cities in the USA. The researchers investigated a human transmitted H5N1 strain of influenza. They have developed an agent-based laboratory that implements a “light-and-fast agent-based model of the spatial and temporal spread of pandemic influenza”. The researchers developed a parallel job controller for testing different combinations of scenarios of influenza as well as a platform-independent application that uses high-performance computing resources ranging from local clusters to TeraGrid supercomputers. Dibble *et al*. (2007) found that this agent-based system was sufficiently fast to generate results via thousands of replications and parameter sets applied.

The coverage of a simulation of the spatial dynamics of an influenza pandemic in an artificial society occurs in Stroud *et al*. (2007). The simulation is created to represent the demographics of southern California using the researchers’ EpiSimS simulator. The simulator is intended to support pandemic planning by determining the correlation between characteristics of local demographics and local pandemic severity. The simulator is founded on cluster computing, involving a high performance computing Linux cluster where its architecture consists of 1,290 nodes and 10.2 TB of RAM is used. The researchers found from simulation that average household size in a community is a strong predictor of the local severity of an influenza pandemic, yet population density is a weak predictor of such severity.

Barrett *et al*. (2008) developed a scalable parallel algorithm, EpiSimdemics, for addressing the spread of infectious disease (like pandemic influenza). Simulation is of realistic social contact networks of the magnitude of 100 million people. Scalability of the algorithm is important because without this the simulation can only be used in the case of small populations. The researchers evaluate a multi-processor implementation of their EpiSimdemics on tera-scale systems to find that the algorithm scales effectively.

Although not involving any form of high performance computing, the work of Tan *et al*. (2005) may be mentioned here. In Tan *et al*. (2005), these researchers address application and illustration of complex adaptive system theory as a method to

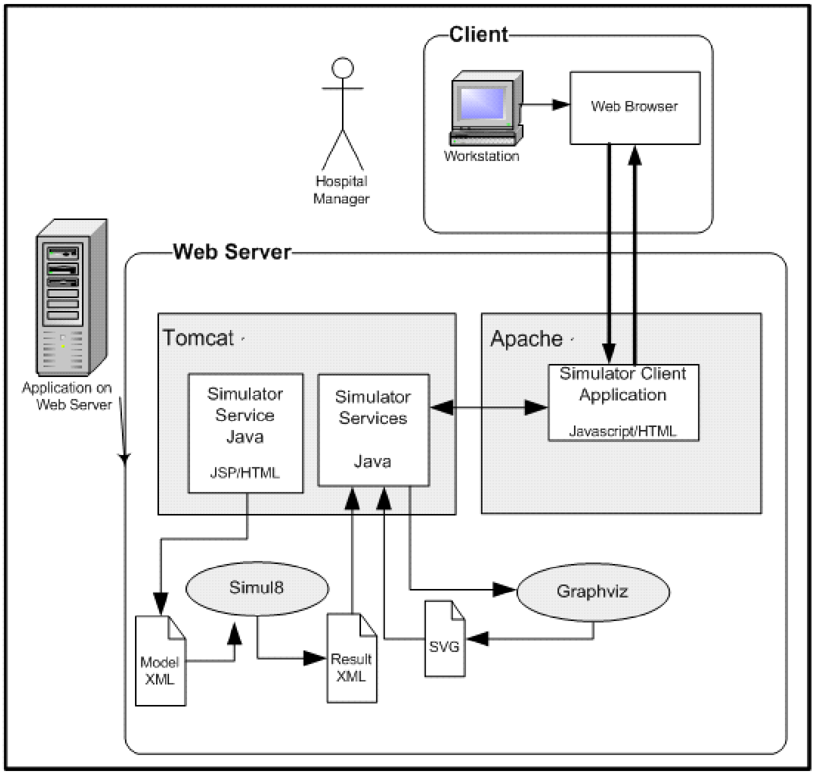


Fig. 1: Architecture diagram of simulation tool

understand how health care and services delivery systems evolve. Thus, their work applies to the area of complex systems. The researchers state that a complex system “exchanges resources with the environment and consists of interconnected components that work together” and that a complex system “encompasses a large number of interacting parts and has structure and behavior that are difficult to understand and predict”. Hence, the researchers provide their analysis of health care and services delivery systems and discuss issues for future systems.

MATERIALS AND METHODS

Architecture: It is imperative that the simulation tool an HM uses must be simple, intuitive and quick to use. We have developed such a Web-based simulation tool. This means that HMs need not have powerful computers to install and run this tool. The tool is installed on a web server and HMs can access this application via the Internet. The only requirement is to have a browser and access to the Internet to use this application.

The initial model of the simulation tool featured an input interface with a novel textual prototype (Bain *et al*., 2007). The input interface is used to create models of hospitals or clinics that are to be investigated and to run simulation against these models. Underlying this software is powerful simulation software called Simul8 version ©12.0. The software architecture is depicted in Fig. 1. The simulation results are presented in a text report as seen in Fig. 5.

The simulation tool presented in this paper is an extension to the initial prototype. This development was focused on improving the presentation of the textual results generated by the prototype. The plan is to present the results in a manner that is simple and easy to understand for the hospital managers and at the same time useful in decision-making. In a typical analysis, the user will create models for different scenarios and simulate the results. HMs will have to study the text files for each of the scenarios before making decision. It becomes a tedious job going through each of the text result files and comparing them before making decisions. The objective of the presentation layer is to analyse the XML files storing the data of the text reports and present data in a graphical format that will be useful for HMs in making informed decisions.

The input interface can be seen on the right-hand side of Fig. 2 (by default it occupies a third of the web page). The input model will not be covered here as it has been covered in Bain *et al*. (2007) and Raikundalia *et al*. (2009). The input module provides an

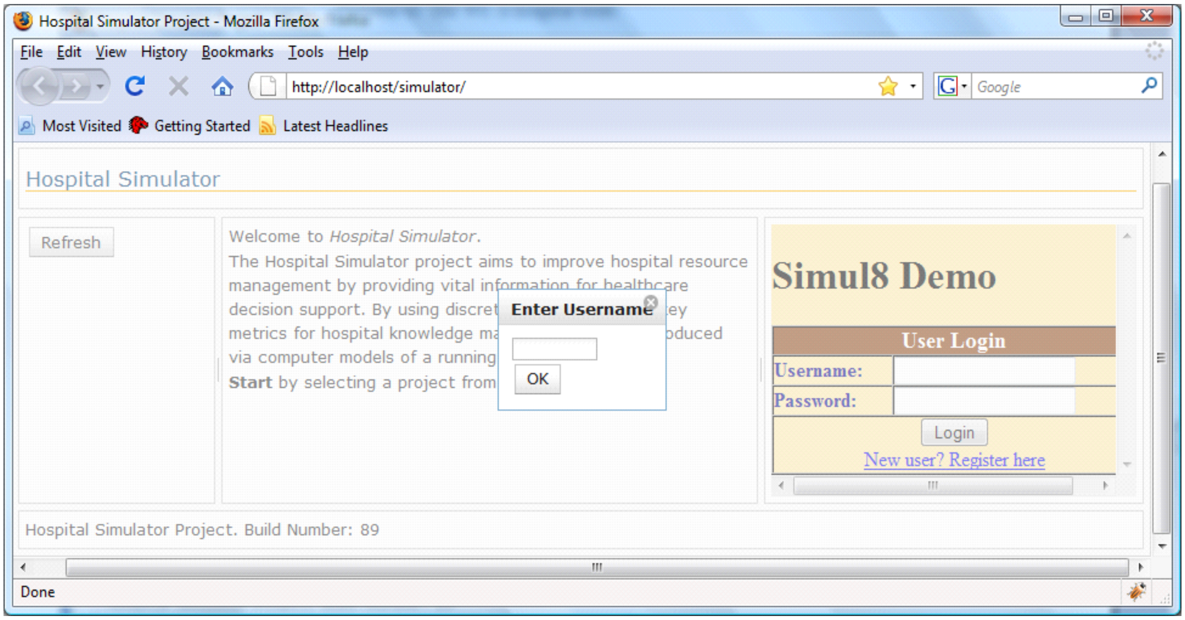


Fig. 2: Main page of simulation tool

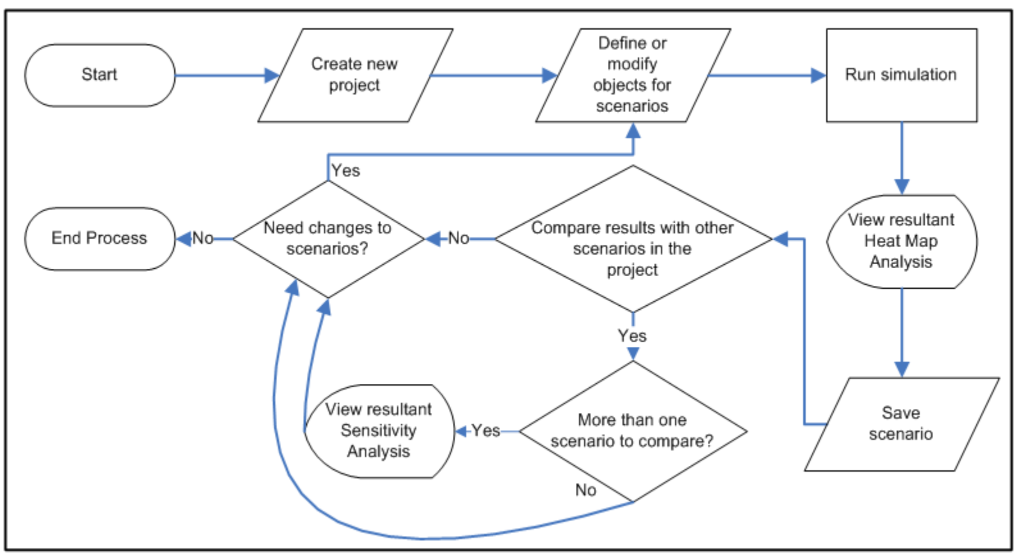


Fig. 3: Flow chart for setting and running projects in simulation tool

interface to create parameterized objects that represent a hospital or clinic that is to be investigated. The model objects are stored in an XML file. Running a simulation against these models will result in a simulation result file, which is again an XML file. The details of the report either in the text format (input interface) or graphical representation are based on these simulation result XML files. A reporting module presents the simulation results in a graphical manner that is easy to read and interpret. Graphing software "Graphviz" is used in rendering the graphs on the screen. The model objects represent real entities in a hospital. Please refer to Fig. 1 for an architecture diagram of the system. Raikundalia *et al.* (2009) describe details of the input module functionality and setting procedure.

System functionality: The main page (Fig. 2) will be presented to the user when first logged onto the Web-

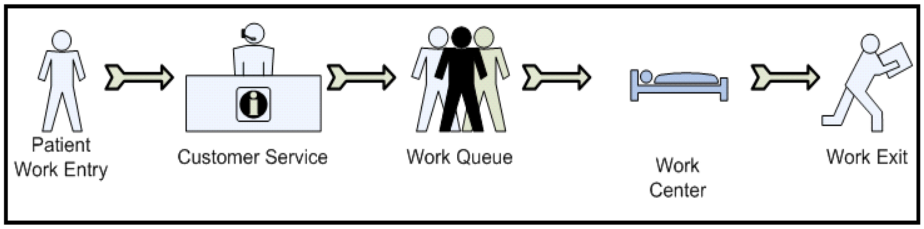
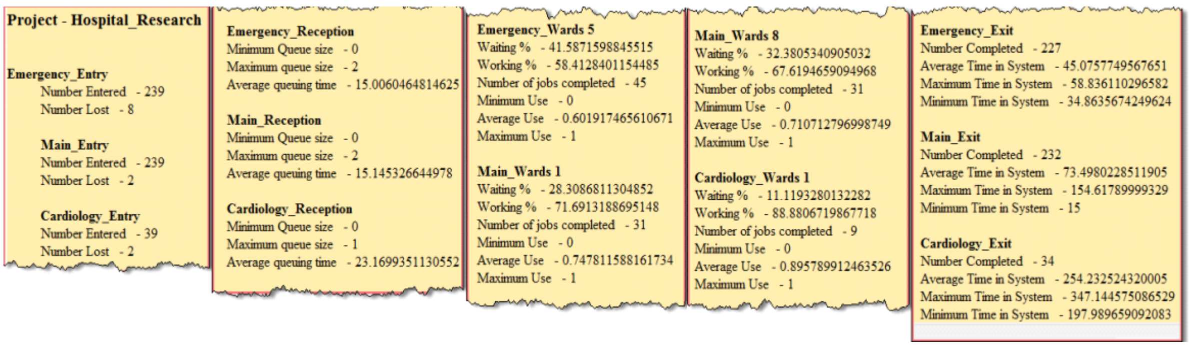


Fig. 4: Model of patient flow through hospital



Project - Hospital_Research

Emergency_Entry
Number Entered - 239
Number Lost - 8

Main_Entry
Number Entered - 239
Number Lost - 2

Cardiology_Entry
Number Entered - 39
Number Lost - 2

Emergency_Reception
Minimum Queue size - 0
Maximum queue size - 2
Average queuing time - 15.0060464814625

Main_Reception
Minimum Queue size - 0
Maximum queue size - 2
Average queuing time - 15.145326644978

Cardiology_Reception
Minimum Queue size - 0
Maximum queue size - 1
Average queuing time - 23.1699351130552

Emergency_Wards 5
Waiting % - 41.5871598845515
Working % - 58.4128401154485
Number of jobs completed - 45
Minimum Use - 0
Average Use - 0.601917465610671
Maximum Use - 1

Main_Wards 1
Waiting % - 28.3086811304852
Working % - 71.6913188695148
Number of jobs completed - 31
Minimum Use - 0
Average Use - 0.747811588161734
Maximum Use - 1

Main_Wards 8
Waiting % - 32.3805340905032
Working % - 67.6194659094968
Number of jobs completed - 31
Minimum Use - 0
Average Use - 0.710712796998749
Maximum Use - 1

Cardiology_Wards 1
Waiting % - 11.1193280132282
Working % - 88.8806719867718
Number of jobs completed - 9
Minimum Use - 0
Average Use - 0.895789912463526
Maximum Use - 1

Emergency_Exit
Number Completed - 227
Average Time in System - 45.0757749567651
Maximum Time in System - 58.836110296582
Minimum Time in System - 34.8635674249624

Main_Exit
Number Completed - 232
Average Time in System - 73.4980228511905
Maximum Time in System - 154.61789999329
Minimum Time in System - 15

Cardiology_Exit
Number Completed - 34
Average Time in System - 254.232524320005
Maximum Time in System - 347.144575086529
Minimum Time in System - 197.989659092083

Fig. 5: Simulation results as a textual output

Table 1: List of distributions supported

Normal	Fixed	Triangular	Rounded uniform	Uniform
Exponential	Erlang	Log Normal	Weibull	Gamma
Beta	Pearson V	Pearson VI	Gauss	Binomial
Poisson	Geometric	Bernoulli	Negative binomial	

based simulation tool. As seen in the Fig. 2 there are individual logins, one for the reporting interface on the left-hand side (LHS) of the screen and for the modelling interface on the right-hand side (RHS) of the screen.

The first step in using the tool starts with defining, the problem that needs to be investigated. For example how is the system going to cope with 10% extra patients in the next six months? How many more beds do we need to operate at 80% working capacity in the next six months? Once the problem is defined, the next step is to collect the current hospital operational information like the number of patients going through the hospital or clinic on a daily, weekly or monthly basis, and the time each patient spends in the system. Once this information is at hand, the next step is to use the simulation tool to build the simulation model and start analysis.

The flow chart in Fig. 3 details the various stages required in defining the simulation models and analysing the results. The first step is to create a model that represents the hospital or clinic being investigated. The next step is to run a simulation against this model. Simulation results can be viewed in text or graphical format, analysed and then applied to informed decisions.

Input module: The input module provides a parameter-based simple modelling interface to create and modify different models of hospitals or clinics that are to be investigated. The hospital or clinic models defined in this tool are based on typical patient flow through a hospital as depicted in Fig. 4.

The users do not have to understand how this powerful simulation software tool works. The modelling software requires four objects to be defined: Work Entry (WE), Waiting Queues (WQ), Work Centres (WC) and Work Exit (WEx). These objects are synonymous to real processes in a typical hospital. The WE can be the Main Reception of a clinic where patients report. The WQ is the waiting area where patients have to wait before they are attended to. The WC can be a ward or a clinic. Finally, the WEx is where the patients leave the hospital.

A powerful simulation tool, Simul8 (version ©12.0), is used to initiate and run discrete event simulation on these models. The results of the simulation on these models are output as XML files. A sample report is presented in Fig. 5.

If analysis is done on an existing hospital or clinic then the initial model will be based on real information. The simulation results for this model can be validated against actual data to ensure the model built accurately represents the real system. The software supports different types of distributions to replicate real-time patient flow through a hospital. The distributions listed in Table 1 are supported by this application.

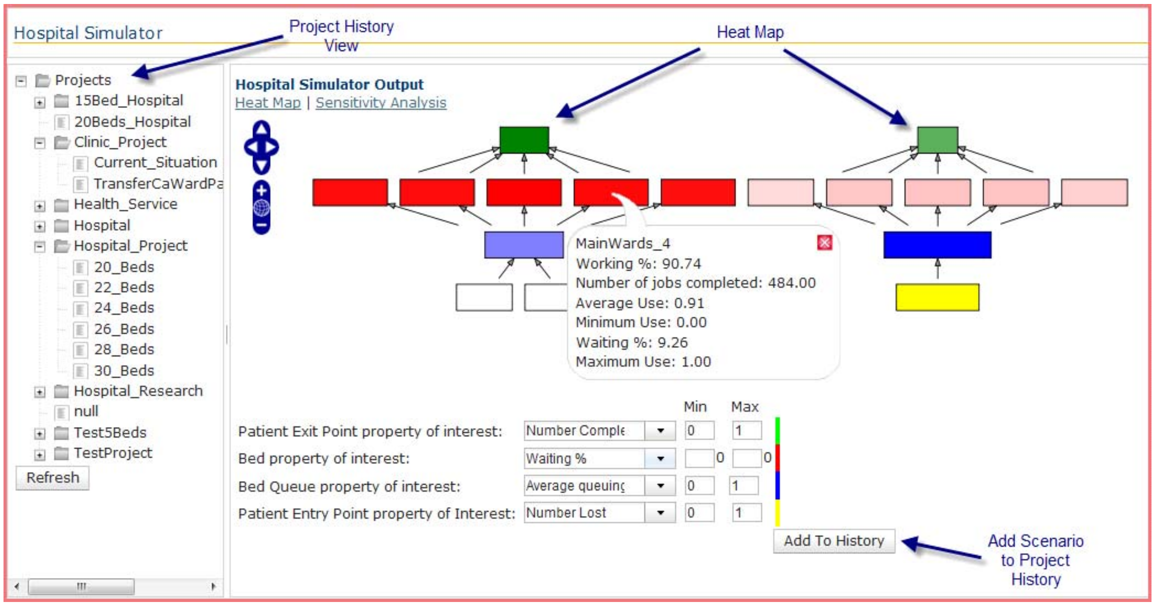


Fig. 6: Report module: Heat map analysis with history view

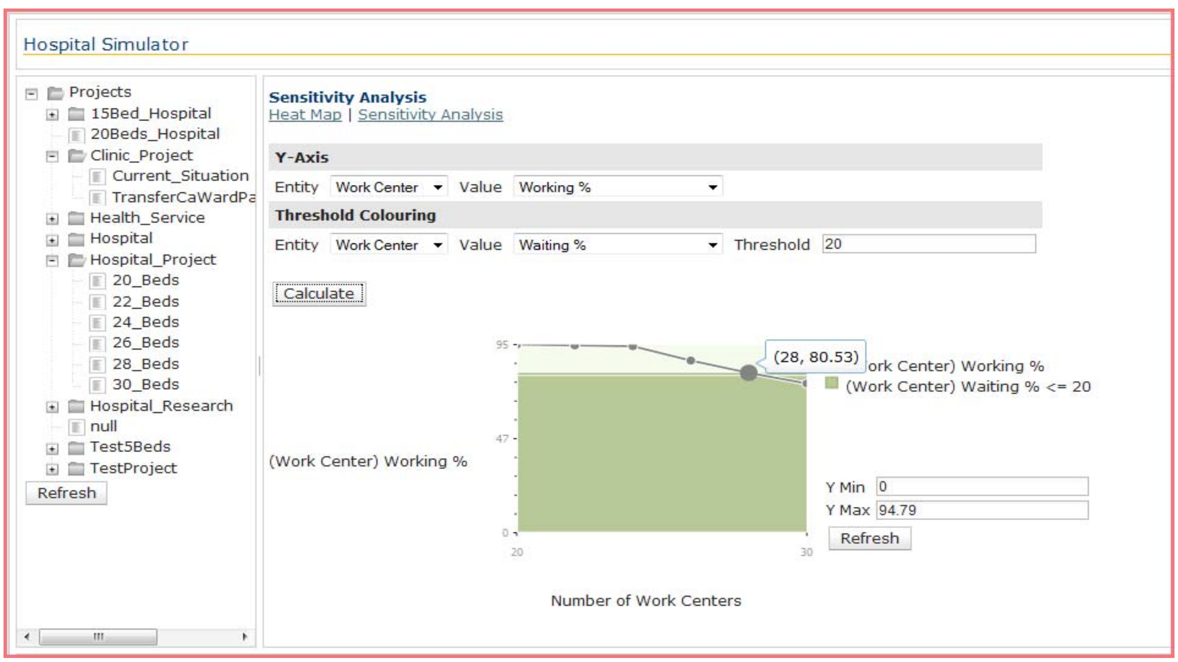


Fig. 7: Report module: Sensitivity analysis with history view

Once a model is established that replicates the real system, variations can be made to this model to simulate the effects of these changes. The HMs can make informed decisions based on the simulation results for the various scenarios simulated. Raikundalia *et al*. (2009) describe details of the input module and also walk through the process of how to create models and run simulations against them.

Report module: The reporting module has the following two styles for presenting the simulation results:

- Heat map analysis (Fig. 6)
- Sensitivity analysis (Fig. 7)

As seen in Fig. 5 the simulation result text report provides comprehensive information about the scenario

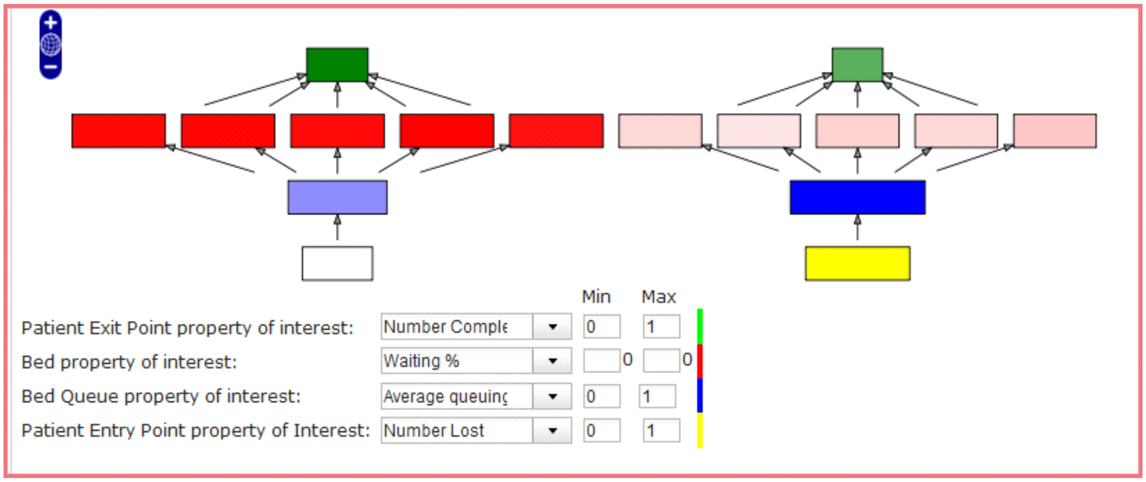


Fig. 8: Heat map analysis

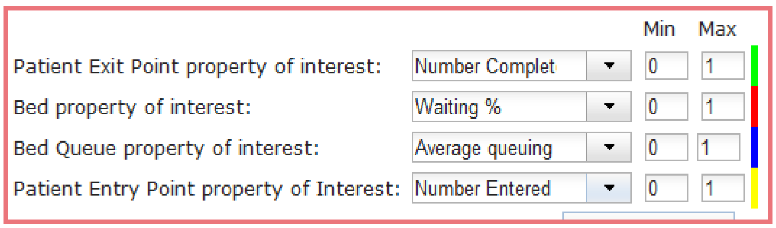


Fig. 9: Heat map list of object type

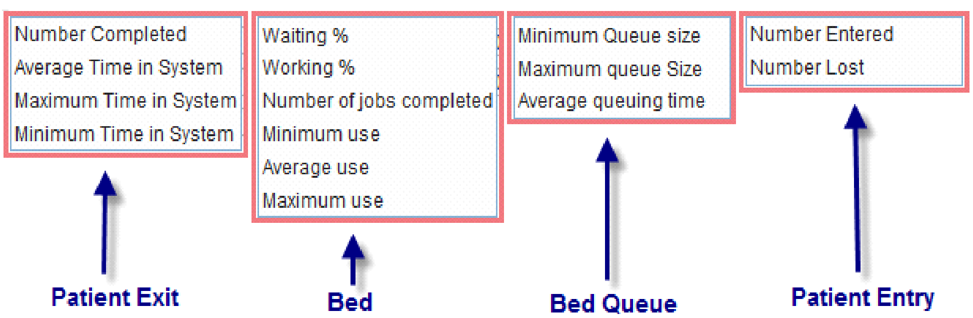


Fig. 10: Heat map lists of values for each object type

for which the simulation was executed. If HMs were to use this text report for managing a hospital, they would find it very difficult to apply. It becomes a challenging task to remember the results of other scenarios when comparing results. Instead the reporting module presents this text report graphically (visually). The Heat Map Analysis (HA) mode is capable of presenting data only the HMs are interested in. The colour coding highlights areas of concern even without seeing the details of results. The bubble speech on each of the objects presented will provide all the details about that object when clicked on them. The Sensitivity Analysis (SA) mode is a very handy tool for HMs to selectively compare data between different scenarios within the project. Undoubtedly a visual, graphical presentation of results is the most user-friendly and useful to a HM than a cumbersome text report. All information for each of the scenarios can be analysed by HA or SA.

Heat map analysis: A Heat Map is a graphical representation of similar objects in the system. A particular colour is used to represent this simulation. The intensity of the colour represents the value of a variable for the object. Larger values are represented as darker colours while smaller values are represented by lighter colours.

HA is used to analyse individual scenarios within a project. As seen in Fig. 8 all the four object types (WE, WQ, WC and WEx) created when defining the scenario in the input module are always represented in this graph. Each object type is represented by a single colour. For example, WCs (beds) is represented in red, WE (Patient Entry) in yellow, WQ (Patient waiting Queue) in blue and WEx (patient discharge) in green.

The intensity of colour depends on the values of the chosen property for each object type. Each object type in the module has different properties. HA can be done on various properties for each of the model objects. HA on beds can be done on Waiting % (beds free), Working % (beds occupied), Number of jobs completed, Minimum use, Average use and Maximum use. The complete list of properties on which HA can be performed is displayed in Fig. 10.

As seen in Fig. 8 the colour of bed objects varies from light pink to dark red. The bed objects in the graph represent the property Waiting% of the beds in this scenario. The light pink in this case represents that the beds are waiting for a very short time as compared to beds represented in dark red. In other words beds in light pink are busier than the beds represented in dark red. On the contrary the beds represented in dark red indicate that they are waiting for a longer time as compared to beds in pink. The representation of the colours will be exactly opposite if Working % is selected as the "Beds property of interest". The Min and Max values (Fig. 9) can be entered to display a particular range of values in rendering this graph.

Sensitivity analysis: SA is a technique for systematically changing input parameters in a mathematical model and determining the effects of such changes. The SA graph in this tool provides a means to compare properties of one object against other properties on the same or other object. For example in Fig. 13c the amount of time the beds are occupied (working %) can be checked against the number of patients served (jobs completed) by beds. Similarly the Working % of beds (beds occupied) can be checked against maximum allowed Waiting % (beds unoccupied). SA is a very effective tool for comparing results and presenting only the data HMs are interested in analysing.

The user may choose to view any property of the object by selecting the entity and the value of interest for that entity as parameter for the y-axis (Fig. 11). The complete list of values that can be chosen for each of the four objects is presented in Fig. 12.

By selecting the y-axis variable and the threshold, the requested data for all scenarios under the project is presented in the graph. For example in Fig. 13a, the y-axis entity is WC (beds) and the value of interest is Working%. The Threshold entity is WC and the value of interest is Waiting%. There are six circles on the graph

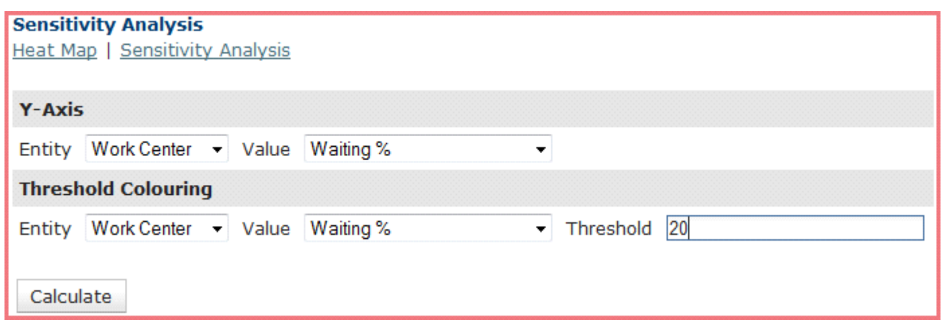


Fig. 11: Sensitivity analysis property selection details

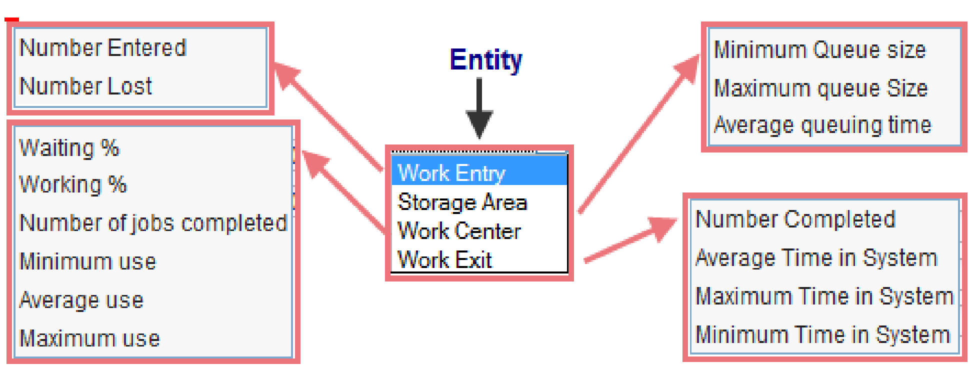


Fig. 12: Entity and value lists of properties

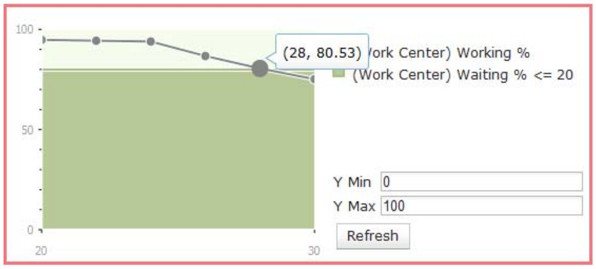


Fig. 13a: SA of WC - waiting% as threshold

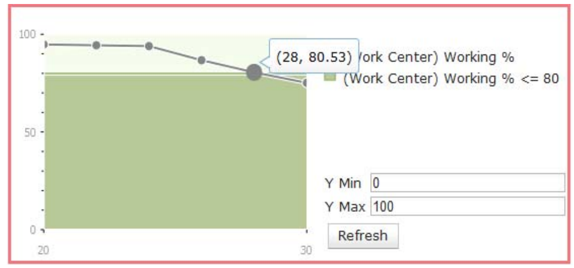


Fig. 13b: SA of WC - working% as threshold

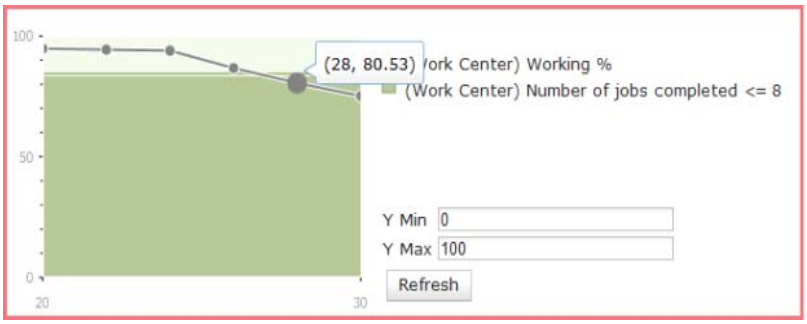


Fig. 14a: SA of WC - patients served as threshold

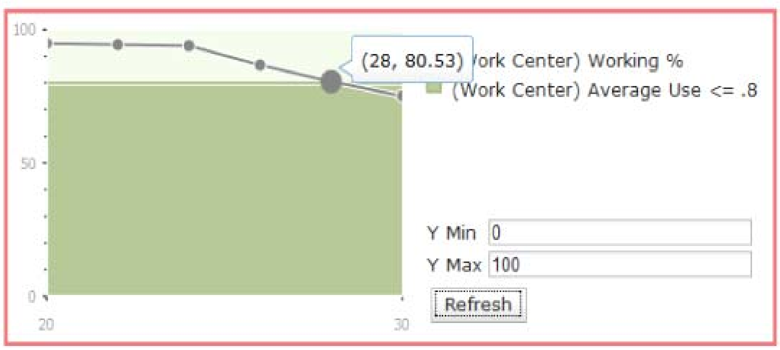


Fig. 14b: SA of WC - average use as threshold

and each one of them represents a scenario within the project. The bubble speech on the circle will indicate the number of beds and value of the selected variable. In this case the number 80.53 represents the Working% of WC in a 28-bed scenario. We can also see how threshold values are presented. In this case the threshold is set to 20% for Waiting% (recall Fig. 11). Looking at the graph it is clear that if the hospital has to operate at a capacity where the hospital beds are not occupied for 20% of their capacity then the hospital will have to operate at least 28 beds, to cater for the current work load.

Figure 13b displays exactly the same results where the y-axis entity is WC (beds) and the value of interest is Working%, but this time the threshold is selected as 80% WC working capacity (bed occupancy). If the hospital has to operate at an 80% bed occupancy rate then there needs to be at least 28 beds in the hospital. Bed numbers of less than 28 mean that the bed occupancy will be above 80% of their capacity.

Figure 14a depicts a graph where the threshold is the number of patients treated at each bed. Similarly Fig. 14b depicts the average use of beds as a threshold against 80% occupancy.

As see in the above graphs (Fig. 13a, b, 14a, b), SA is very handy for HMs in providing the information they need in decision-making. SA provides an interface to display only the information the user is interested in. The rest of the information is filtered out. The x-axis for these graphs always represents the number of WCs (e.g., beds, clinics, etc.). To utilise this functionality it is mandatory that the number of WCs must be different for each scenario within the project. If the number of WCs remains

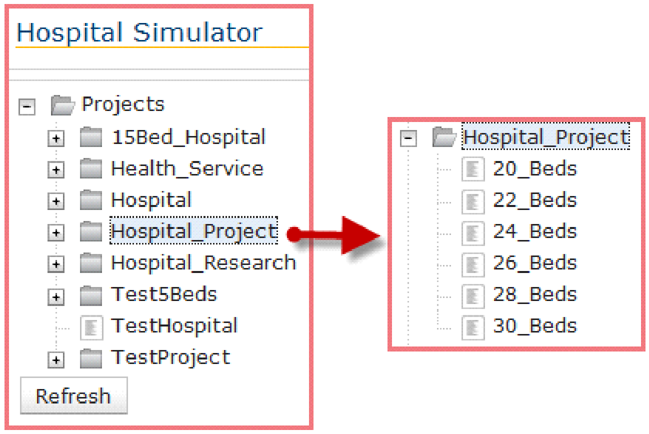


Fig. 15: Project history

the same between different scenarios then the sensitivity analysis will display a single circle making this presentation model unsuitable to compare results between different scenarios.

History view: The first step in using this simulation tool is to create a project in the input module, define the hospital or clinic model that has to be investigated and run a simulation against this model. Once this is done the project is automatically displayed in Project History View (HV) of the report module (Fig. 6). The user clicks the Refresh button shown in Figure 15 to refresh the tree view with the new project created. The HA graphs for the latest simulation can be viewed by clicking on the project in HV. After analysing the results the HM may save this scenario under the project for future reference. The scenario can be saved by clicking the "Add to History" button (Fig. 6) in HA and giving it a name. Figure 15 shows the various projects created on LHS of screen. On the RHS of the figure we can see the different scenarios (e.g., 20_Beds, 22_Beds) simulated.

It is mandatory that more than one scenario has to be saved for the project with varying WC numbers for SA. Every investigation model developed will belong to a single project. The model can be executed multiple times with minor changes to investigate the impact of the changes. The results for each run (scenario) can be stored under the project. All projects and scenarios are displayed on the LHS of the screen. If the projects are no longer required to be displayed they can be archived and removed from the HV.

Scenarios: The following example illustrates how this simulation tool will assist the HMs in their decision-making.

Scenario 1: Consider a hypothetical hospital with 20 beds. It is known that the hospital is very busy and often has to divert patients to other hospitals due to lack of capacity. The aim of this investigation is to analyse the current situation and find out how many more beds are needed in the hospital to cater for the current patient numbers while operating at an 80% occupancy rate.

The first step in the analysis process is to create a project and define a hospital model in the input module. Please refer to Raikundalia *et al*. (2009), for more details on creating a project and defining a hospital model. Hospital_Project in Fig. 15 represents this case study. The HM used the normal distribution in defining the patient entry and the patient stay in the hospital. The next step is to run the simulation and compare the results with real records to validate the accuracy of the model.

HA graphs can be used to view the results. Information on the individual objects can be obtained by clicking on them. The bubble speech for an object will present all its information. Figure 16 details the simulation results representing the current issues. We can see that 239 patients arrive at the hospital and 34 of these patients are sent back or diverted to other hospitals due to lack of beds. The occupancy rate of the beds is above 94% and 9 patients are treated at each bed.

The HM saves the results of the first simulation under the project. Scenario 20_Beds (Fig. 15) represents the actual conditions. The HM must decide what changes will be made in the hospital to improve the situation. In this case the HM has decided to add two extra beds for each new run. The HM makes the changes to the model and simulates the results. We have simulated six scenarios each time incrementing by two hospital beds. Adding two extra beds indicated that only 15 patients will be sent back without being attended while the bed occupancy rate increased to 97%. Once the HM has simulated two or more scenarios they can perform a SA on the results. As seen in Fig. 13a-b, the hospital will need a minimum of 28 beds to operate at 80% bed occupancy rate to cater for the current patient numbers.

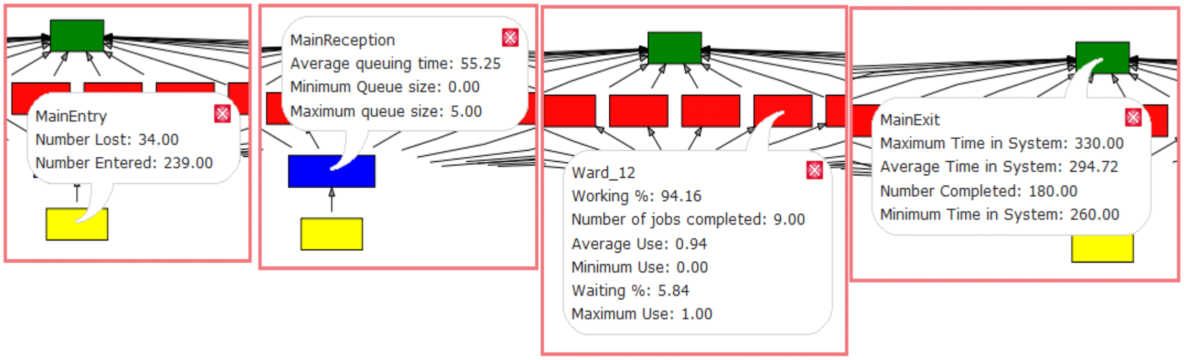


Fig. 16: Bubble speech for heat map analysis

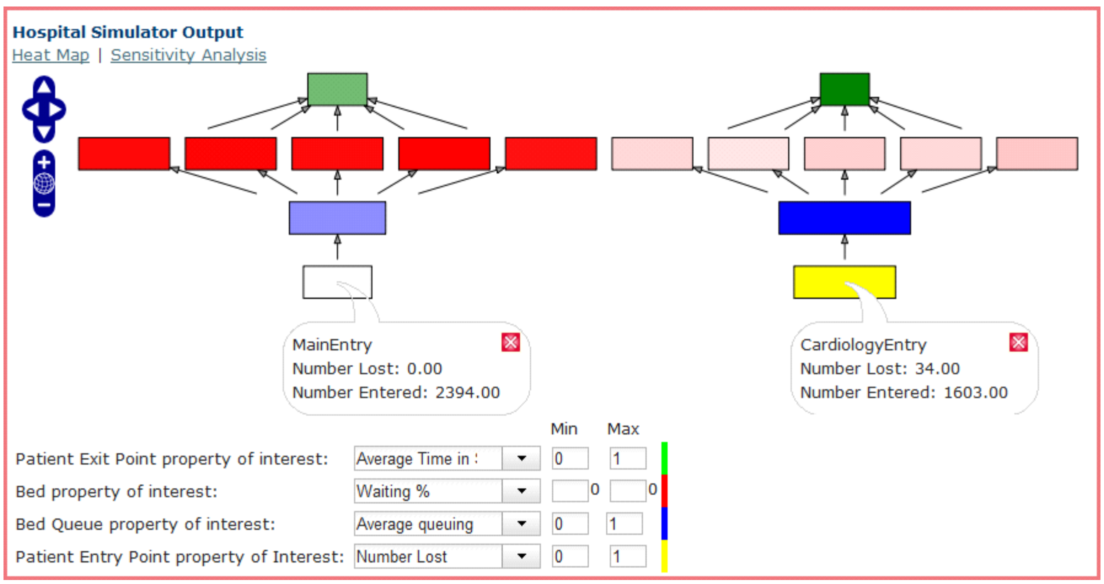


Fig. 17: Heat map analysis

Scenario 2: The Clinic Project is a case study on a hypothetical healthcare centre that has two types of wards: a Main Ward (MW) and Cardiology Ward (CW). It is known that CWs are always busy and there are occasions where patients are sent back due to lack of capacity. The aim here is to perform "What-If" analysis on the current situation and to answer the following questions:

- Is it possible to divert some of the cardiology patients to the MWs assuming that MWs are equipped to treat cardiology patients?
- Does the diversion have any impact on the MW patients?

The first step in the analysis process is to create a project and define a healthcare model using the input module. The HM used the fixed distribution and normal distribution in defining patient entry into the hospital and used the normal distribution to simulate patient stay in the hospital. Please refer to Raikundalia *et al*. (2009), for more details on creating a project and defining a hospital model. The next step is to run the simulation and compare the results with real records to validate the accuracy of the model.

HA graphs can be used to view the results. In Fig. 17 we can see details of the simulation results representing the current issues. Just by looking at the red and pink colour of the WCs, HMs get an idea that CW idle time is lesser than MW idle time. In other words the CW occupancy rate is higher than the occupancy rate of MW. HMs can retrieve all details of any WC by just clicking the rectangular box. As seen in Fig. 18 the bubble indicates that MW occupancy rate is 89% as compared to

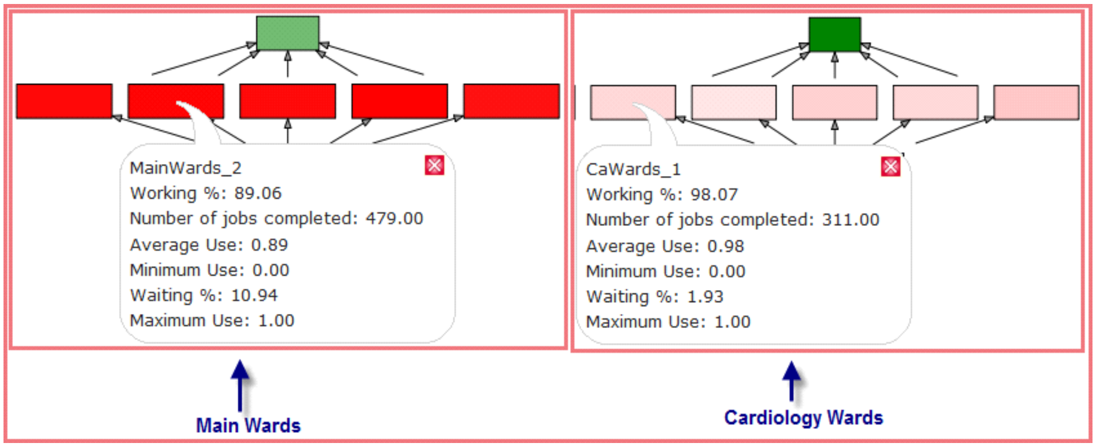


Fig. 18: Heat map analysis showing wards occupancy rate

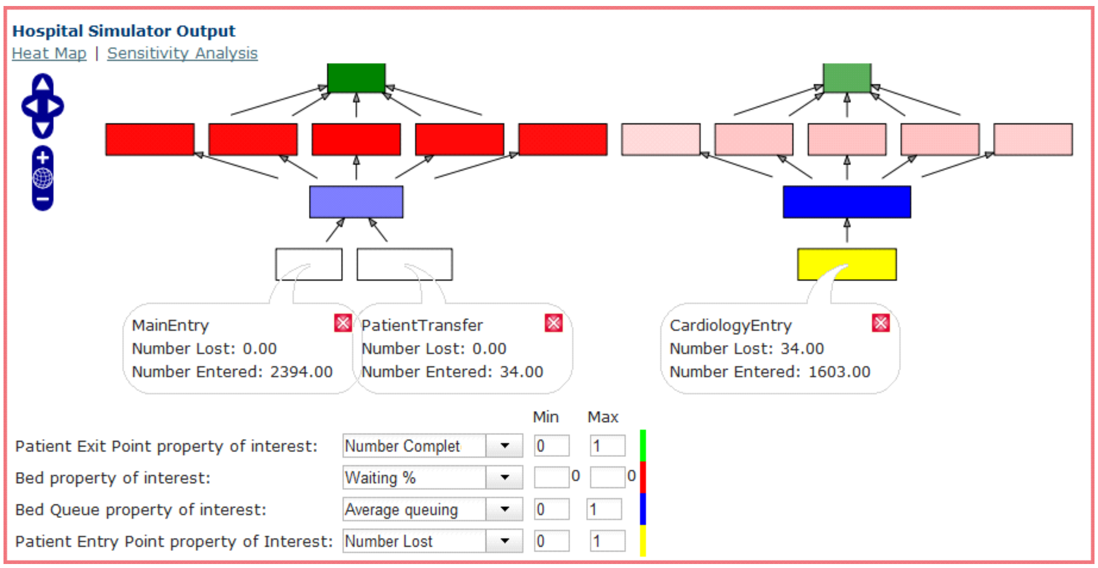


Fig. 19: Patients transferred from CW to MW

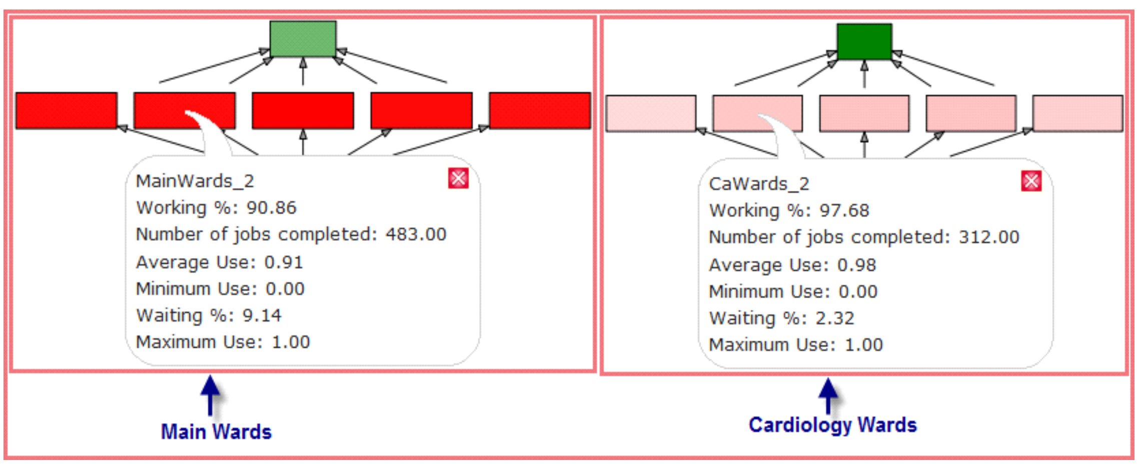


Fig. 20: Work centre details after patients transferred from CW to MW

98% occupancy of CW. The simulation of the current situation indicates that 34 patients could not be treated in CWs due to lack of capacity.

The above analysis indicates that CWs are operating to full capacity and the MWs still have capacity to treat extra patients.

In scenario 2 cardiology patients who were sent back from CWs are treated in MWs assuming that they have the required expertise and spare equipment to treat them. As seen in Fig. 19 all patients have been treated and no patients had to be sent back. Again looking at the bubble (Fig. 20) for the WC, it can be seen that the MW occupancy rate has gone up to 91%. Based on these results the HM will be able to make decisions with confidence. More scenarios can be simulated to find out what is the maximum number of patients that can be diverted from Cws.

RESULTS AND DISCUSSION

Certain limitations currently exist in our system and they are as follows:

- The application provides functionality to save the history of a scenario, however, there are no options to delete the history from the application. The possibility of deleting the history will have to be considered in the next version of the system.
- The current simulation module does not support a partial day in defining working hours. A better option is to attach a calendar where we can define working hours as well as weekends and public holidays.

CONCLUSION

This study covered our Web-based, discrete event-based simulator for hospital managers. Hospital managers use the functionality and user interfaces described in this paper to execute scenarios in determining the effects to changes in these scenarios. By being able to make changes to a scenario, for example, number of beds, managers is generated visualizations they can apply to the management of the hospital. We covered two example scenarios illustrating how the simulator is used by the hospital manager.

ACKNOWLEDGMENT

We would like to acknowledge the source of funding for this project: 2006 Victorian Partnership for Advanced Computing e-Research Program Grants Scheme - VU Special Round 8. In addition, we would like to thank staff of Victorian Partnership for Advanced Computing (VPAC) who were involved in the development of the simulator: Alan Lo and Angus MacCauley. We also thank the CEO of VPAC, Prof Bill Appelbe, for his initial advice on the project. Finally, thanks go to Thiagarajan Ramadoss and Devang Mehta for their initial work on the grant proposal.

REFERENCES

Axner, L., 2007. High performance computational hemodynamics with the lattice boltzmann method. Ph.D. Thesis, University of Amsterdam, The Netherlands.

Bain, C., G.K. Raikundalia and S. Mehta, 2007. Hospital management knowledge discovery using discrete event simulation. Proceedings of First Australasian Workshop on Health Knowledge Management and Discovery, Ballarat, Australia, 30 January - 2 February, pp: 209-218.

Barrett, C., K. Bisset, S. Eubank, X. Feng and M. Marathe, 2008. EpiSimdemics: an efficient algorithm for simulating the spread of infectious disease over large realistic social networks. Proceedings of the 2008 ACM/IEEE Conference on High Performance Computing November, Austin, Texas, 15-21 November, pp: 1-12.

Dibble, C., S. Wendel and K. Carle, 2007. Simulating pandemic influenza risks of U.S. cities. Proceedings of 2007 Winter Simulation Conference, Washington, DC, 9-12 December, pp: 1548-1550.

Pormann, J., C. Henriquez, J. Board, D. Rose, D. Harrild and A. Henriquez, 2000. Computer simulations of cardiac electrophysiology. Proceedings of the 2000 ACM/IEEE Conference on Supercomputing, Dallas, Texas, 4-10 November.

Raikundalia, G.K., C. Bain and S. Mehta, 2009. Towards an advanced computing solution for hospital management using discrete event simulation. Int. Rev. Comput. Software, 4:15-25.

Stroud, P., S. Del Valle, S. Sydoriak, J. Riese and S. Mniszewski, 2007. Spatial dynamics of pandemic influenza in a massive artificial society. JASSS-J. Artif. Soc. S., 10(2).

Tan, J., J. Wen and N. Awad, 2005. Health care and services delivery systems as complex adaptive systems. Commun. ACM, 48: 37-44.