DECISION-MAKING TUTOR: PROVIDING ON-THE-JOB TRAINING FOR OIL PALM PLANTATION MANAGERS

SAGAYA AMALATHAS

Department of Computer Science and Software Engineering, University of Canterbury,
Private Bag 4800, Christchurch, New Zealand
sagaya.amalathas@pg.canterbury.ac.nz

ANTONIJA MITROVIC

Department of Computer Science and Software Engineering, University of Canterbury,
Private Bag 4800, Christchurch, New Zealand
tanja.mitrovic@canterbury.ac.nz

SARAVANAN RAVAN

RANN Consulting, Malaysia
ravan@rann-consulting.com

Over the years many Intelligent Tutoring Systems (ITSs) have been used successfully as teaching and training tools. Although many studies have proven the effectiveness of ITSs used in isolation, there have been very few attempts to embed ITSs with existing systems. This area of research has a lot of potential in providing life-long learning and workplace training. We present DM-Tutor (Decision-Making Tutor), the first constraint-based tutor to be embedded within an existing system, the Management Information System (MIS) for oil palm plantation management. The goal of DM-Tutor is to provide scenario-based training using real-life operational data and actual plantation conditions. We present the system and the studies we have performed. The results show that DM-Tutor improved students' knowledge significantly. The participants found DM-Tutor to be easy to understand and interesting to use.

Keywords: Embedded intelligent tutoring system; Oil palm plantation management; Evaluation.

1. Introduction

In many professions, employees are required to have extensive practice-based training before they can perform their duties. Astronauts need to be trained for years before lift-off, and doctors need to serve as interns first. However, most organizations limit the amount of training provided to employees due to cost and time constraints, so employees have to train on the job. Placing new managers trained on-the-job is costly in its own way, as the ineffective decisions they make could make the organizations lose money and business opportunities. Current learning technologies can help create trained novices but not expert decision makers. Computer-based training can present information and test factual recall but typically cannot provide individualized coaching, assess performance or

provide feedback to students. These training technologies could be appropriate if the aim of the training was to produce trained operators who require knowledge at the novice level. However, to become truly proficient in decision making an employee requires extensive practice solving realistically-complex problems in a wide range of situations, combined with coaching or feedback from senior or more experienced peers, or experts in the domain.

Intelligent Tutoring Systems (ITSs) have been proven to provide the benefits of one-on-one teaching or training automatically and cost effectively in a variety of domains. LISP tutor (Anderson & Reiser, 1985), Andes (Van Lehn et al., 2005) and PUMP Algebra Tutor (Koedinger, Anderson, Hadley, & Mark, 1997) are examples of successful ITSs. SQL-Tutor (Mitrovic & Ohlsson, 1999; Mitrovic, 2003), NORMIT (Mitrovic, 2005), and EER-Tutor (Suraweera & Mitrovic, 2004; Zakharov, Mitrovic, & Ohlsson, 2005) are some of the many constraint-based tutors (Mitrovic, Martin, & Suraweera, 2007; Mitrovic, 2012) that have been developed and successfully implemented. In this paper, we present DM-Tutor (Decision-Making Tutor), an ITS that provides training on plantation decision-making for the oil palm domain (Amalathas, Mitrovic, & Ravan, 2009, 2010). We aim to make several significant contributions in our research. This is the first attempt to embed a constraint-based tutor with an existing system. Secondly, we investigate the benefits of providing on-the-job training through this integration. DM-Tutor has been embedded with an MIS (Ravan, 2007) that is currently being used to manage several oil palm plantations. The MIS contains extensive operational data of yield records and plantation cultivation. As the information contained is highly domain specific, managers who are new to the domain or to the MIS face difficulties in making accurate operational analyses and this affects the decisions they make. With DM-Tutor embedded within the MIS, users are able to practice plantation decision-making using real-life operational data. The goal of DM-Tutor is to help users apply theoretical concepts of plantation analyses into real-life plantation decision-making.

We present related work in the following section, and then introduce DM-Tutor in Section 3. The following section presents the pilot study, followed by Section 5, which presents the full evaluation study, and Section 6, which presents the results. The conclusions are presented in the final section.

2. Related Work

Even though ITSs have been proven as effective teaching and training tools, there have been very few attempts to embed them within other systems. ETS (Embedded Training System) (Cheikes et al., 1998), MACSYMA Advisor (Genesereth, 1979), Geometer Tutor and Excel Algebraic Tutor (Ritter & Koedinger, 1996), STIM-Tutor (Gonzalez, Burguillo, & Llamas, 2007), PAT (Personal Access Tutor) (Risco & Reye, 2009) and SBT-AID (Shatz, Bowers, & Nicholson, 2009) are some of the attempts made to embed ITSs into existing systems.

MACSYMA Advisor was developed to assist users in using MACSYMA, the algebraic manipulation system. The objective of building Advisor on top of MACSYMA

was to create an online consultant that could provide an interactive environment with intelligent feedback for MACSYMA users. As the Advisor aimed to help the user in his/her problem-solving attempts, understanding the user's problem solving approach well was critical for MACSYMA to provide comments or corrections to users. Therefore, much effort was directed towards designing the user model and using it to interpret user's actions. Advisor generated a plan of the user's expectations of MACSYMA and then worked towards debugging the plan. MACSYMA tried to identify user's error patterns and tried to communicate with the user to confirm if the assumptions made were correct. Even though most of the modules in MACSYMA Advisor were implemented, the Advisor was only an experimental system that lacked an interface and never reached the point of being made available to MACSYMA users.

Plug-in tutor agents were added on top of Excel 4.0 for Macintosh and Geometer Sketchpad to create Excel Algebraic Tutor and Geometer Tutor respectively. Plug-in tutor agents could have many potential benefits in creating a more flexible learning both in the education and workplace environments. Cost and time taken to develop ITSs could be lowered by using existing software applications to develop ITSs. Domain knowledge was stored inside the tutoring agent. Curriculum manager maintained the student model and guided the student through learning the domain knowledge and lastly, the translator handled the communication between all the components of the system. User interfaces that looked similar to the actual software were created on the tool. Tutor menus were added to allow students to access the tutors to solve problems. The tutors monitored students' actions in the systems interfaces and provided descriptive explanations and interactive guidance for students. If the user submitted an incorrect answer, a feedback message was passed from the tutoring agent through the translator and was displayed in the user interface of the systems. Initial evaluations done on Excel Algebraic Tutor (Mathan & Koedinger, 2002) showed that it provided a high learning outcome for students. To our knowledge no detailed empirical evaluation was done to analyse students' interactions with Geometer Sketchpad.

Intelligent embedded training systems have been considered a successful approach to providing training using real-life working scenarios. Trainees practice solving problems on the main application with guidance and feedback from the training system. An Embedded Training System (ETS) was developed and integrated with a Complex Information System (CIS) for military operations to provide training for novices on how to operate military devices. Trainees use a particular system for a period of time using the ETS and after training period is over, the ETS still remains within the application system to provide training when the users want to enhance their knowledge. ETS observes all the actions of the user while using the system. For each user action on the interface, a message is sent to ETS. ETS was not fully evaluated as the developers felt it was not suitable or mature enough to be evaluated with participants. However, through the initial studies performed, they were able to identify several limitations in the system and felt that the ETS's training service was not robust enough to handle a variety of users' behavior when interacting with the system.

STIM-Tutor, an ITS for the medical domain, was developed and integrated with SINCO-TB, a Health Information System (HIS) database that contained information about patients with tuberculosis. To integrate and share patient's clinical data between ITS-CBR and HIS, an additional component, HL7, a standard messaging protocol was added to the existing system architecture and was used to communicate between ITS-CBR and the HIS. The objective of sharing patients' clinical data between ITS-CBR and HIS was to provide training to medical students using real live cases and patients' actual medical conditions. When a query was received by ITS-CBR, a message was sent out to request for records in SINCO-TB that matched the query. SINCO-TB then responded with demographic information about the patients. The next message requested for patients health conditions and received a response that included the patient's complete health report including symptoms, health problems and diagnosis. Integrating patients' information between ITSs and Health Information Systems has a lot of potential in the training of new health care students and exposing them to complex and real-life situations. However, the integration of STIM-Tutor and SINCO-TB was not evaluated with actual users.

PAT was designed and implemented to run within MS Access to help students create forms and reports using MS Access. PAT consists of a domain model, user model, instructional expert and a user interface. The domain model contains the knowledge to be taught to students. The user model contains the system's assumption of the student's knowledge of the domain, student's personal characteristics and learning preferences. The instructional expert diagnoses the student's solution and provides personalised feedback and hints to users. Students access PAT through the MS Access interface, attempt exercises, and submit solutions to PAT. Their solutions are compared to the ideal solutions stored in PAT's knowledge base. Based on the errors in their solutions, students receive several levels of feedback messages from PAT. The questionnaire analysis done showed that students liked PAT and looked forward to using ITSs for their other courses as well.

SBT has been a successful approach for training on dynamic and complex real-world scenarios. Even in areas that situated tutors have been found to be unsuitable, SBT has been considered effective. However, SBT is expensive, its development, time-consuming, and required support from well trained instructors to conduct the training. SBT-AID (Shatz et al., 2009) is the automation of SBT and has tutoring components integrated within it. SBT-AID was developed and integrated into an existing military training simulation to provide adaptive training using multiple scenarios for many personnel at the same time. SBT-AID's training could be broken down to several phases: identifying task requirements, creating user profiles, selecting and applying training goals and plans, presenting various training scenarios, assessing trainees during and after training, recording trainees performance and post-task activities and finally, recording meta data of the training and history of trainees performance. However, the objective and focus of SBT-AID was not described through this work. The benefits of the integration were not

supported by any evaluation studies as evaluation studies were not conducted for SBT-AID.

3. DM-Tutor

The goal of plantation management is to optimise available human, financial and technical resources in order to maximise yield and profit. Plantation or estate management has to ensure careful planning and proper allocation of all of these resources as unplanned or misdirected resources consumption and poor internal financial control can cause a failure or a major disruption in the plantation's operations. Plantation decision making is a complex domain and involves the usage of various types of skills. In order to make good plantation management decisions, the manager must first access relevant information. Next, the manager needs to perform appropriate analyses in order to ascertain the state of various crucial parameters in plantation management. If those analyses result in the identification of any problems, the manager needs to provide a recommendation on how to improve a particular condition in a given plantation location.

Our goal was to create embedded training for a workplace environment. We decided to use the MIS for oil palm plantations, a tool used daily for managing operational activities in several oil palm plantations in Malaysia and Indonesia (Ravan, 2007). The MIS is a comprehensive system containing information and reports used by managers to analyse yield, fertilizer and costs operations in the estates and plantation. The MIS contains data required for estate cost functions such as material inventory, vehicle management, human resource management and accounts management. Managers can request various types of reports from the MIS, specifying the time period and locations. However, the MIS does not provide any intelligent support for decision making, and therefore managers need to know the types of analyses they need to perform and also how to do the analyses using the information in the MIS. Some managers lack formal training and need to obtain the necessary knowledge on the job. In order to perform analyses, managers need to drill down to the exact detailed information and carry out those analyses themselves. The goal of DM-Tutor is to provide on-the-job training for less experienced managers, using operational data from the MIS.

It is important to stress here the advantage of embedding the ITS within the real, operational environment (the MIS). By using this approach, the user can get the training in the same environment they use in their day-to-day work. Therefore, they do not need to switch to a different application – they can simply ask for a problem of a specific kind while staying in the MIS, which is a convenient way to learn. The problems that DM-Tutor provides are based on the real data from the MIS. In order to solve them, the user needs to access the reports from the MIS. Such situated learning is more beneficial than using artificial, context-free problems.

3.1. *The development of DM-Tutor*

To the best of our knowledge, there has not been an ITS for plantation decision-making, and DM-Tutor is novel in that respect. DM-Tutor was developed using ASPIRE

(Mitrovic, McGuigan et al., 2008; Mitrovic, Martin et al., 2009), an authoring system and deployment environment for constraint-based tutors. Constraint-Based Tutors (CBTs) use constraint-based modelling (CBM), a technique based on the theory of learning from performance errors (Ohlsson, 1992). In constraint-based tutors, knowledge is represented as constraints that specify conditions which the solution must satisfy, instead of generating a multitude of problem-solving paths. Constraints are used to model both the domain and the student. CBTs match students' solutions to the constraints, and generate feedback based on the satisfied and violated constraints.

Developing ITSs in ASPIRE is a semi-automatic process. The author starts by describing the instructional task and specifying whether it is a procedural task or not. For procedural tasks, the author needs to specify the steps making the task, as well as their order. When solving a procedural task, the student would need to complete the current step before being allowed to work on the following step. Next, the author needs to describe the domain in terms of its ontology, and also provide examples of problems and their solutions. From this information, ASPIRE automatically generates constraints (syntactic and semantic), as well as a form-based interface. Such interfaces are of course not suitable for all instructional tasks, and ASPIRE allows the author to provide Java applets to be used instead of the default interface. ASPIRE provides all the functionality necessary for modeling students. Its pedagogical module also provides several options regarding pedagogical strategies, such as problem selection mechanisms and availability of various levels of feedback. The author can test his/her system before it is available online. ASPIRE also provides support for creating student accounts, classes of students with pre-selected pedagogical strategies and monitoring performances of students. For more information about ASPIRE please see (Mitrovic, McGuigan et al., 2008; Mitrovic, Martin et al., 2009). DM-Tutor contains 60 syntactic and 140 semantic constraints.

DM-Tutor is the first CBT to be embedded within an existing system. As stated earlier, we wanted DM-Tutor to provide scenario-based training on plantation decision making by using real-life operational data from the MIS. The approach used in the development of DM-Tutor had to be suitable to its integration with the MIS. Problem-solving steps in DM-Tutor had to be carefully planned and matched to the actual way managers make plantation decisions. Figure 1 presents the overall architecture of DM-Tutor, consisting of a student modeler, pedagogical module, interface module, constraints that represent domain knowledge, and a database of problems and solutions. Please note that since DM-Tutor was developed in ASPIRE, the student modeler and pedagogical module are provided by ASPIRE. We have developed Java applets that are used for some steps during problem solving.

The student model contains information about student's knowledge and is updated every time the student interacts with DM-Tutor. The student's solution is matched to the constraints that are relevant to the task the student is attempting, in order to identify any mistakes in the student's solution. The lists of relevant, satisfied and violated constraints serve as a short-term student model, which is then used to update the long-term model of the student's understanding of the particular task in DM-Tutor.
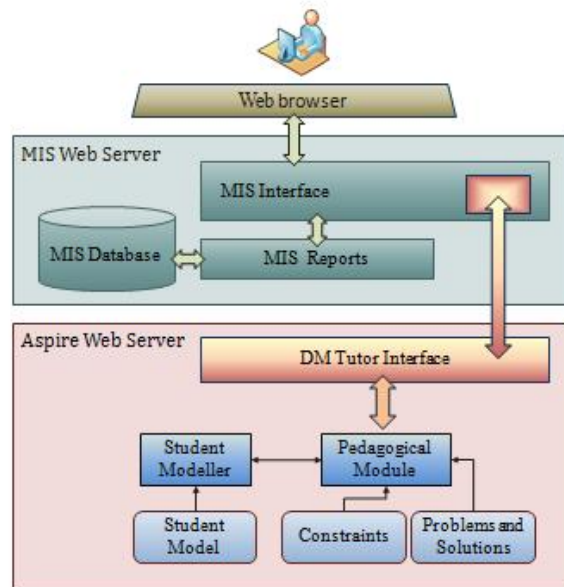
Figure 1. Architecture of DM-Tutor embedded with the MIS.

As mentioned earlier, ASPIRE provides the author with the facility to specify which levels of feedback will be available in the ITS. For DM-Tutor, we have the following levels: correct/incorrect, error flag, hint, detailed hint, all errors and full solution. The initial level of feedback only notifies the user whether the solution is correct or not. When a student submits an incorrect solution for the second time, an error flag is provided, pointing out the part of the solution which is wrong. If the student's third submission is also incorrect, a hint informs the student of the domain principle which was violated by his/her solution. If the student is still unable to correct the mistake, a detailed hint is given to the student. Detailed hints provide more descriptive information of the error to help students understand their error better. The hint and detailed hint messages come from the violated constraint; during development of DM-Tutor, we defined those messages for all constraints. The all errors level provides the list of hints for all errors identified in the solution. Students can request a specific level of feedback at any time, but they are prevented from seeing the full solution until they make three attempts at answering the given question. DM-Tutor therefore offers the feedback levels which are common to constraint-based tutors, e.g. Mitrovic and Martin (2000). Similar feedback levels are available in most ITSs. The teacher can easily modify the feedback levels which are offered by the system, by modifying the relevant parameters in ASPIRE.

The MIS is a web-based system and is accessed via a web browser. We modified the MIS to allow its integration with DM-Tutor. By embedding DM-Tutor into the MIS, we were able to leverage on the operational knowledge contained in the MIS to provide on-

the-job training to employees through DM-Tutor. As the MIS is a workplace system used extensively by employees every day, it was necessary to ensure that the functionality of the system remained the same after the integration with DM-Tutor. Figure 2 presents a screenshot of the MIS. Various types of reports available in the MIS are listed in the left pane. The right pane shows the yield report for the selected plantation in 2008.

Since the employees use the MIS in their daily work, it was important to keep the modifications to the MIS interface as minimal as possible, in order to reduce the time needed to adjust to the new interface. As we wanted DM-Tutor to provide learner-paced, task-oriented training to the users, we made the integration of DM-Tutor to the workplace system permanent and available to the employees always. The user can access DM-Tutor from the MIS, by clicking on the link at the bottom of the left pane. The user can exit DM-Tutor at any point and continue with their daily work using the MIS. DM-Tutor updates the student model with the current knowledge learned and waits for the next time the user wishes to continue learning. To the users, it would appear as if they are accessing DM-Tutor from the MIS menu and to them DM-Tutor appears to be a part of the MIS.

3.2. *The instructional tasks in DM-Tutor*

DM-Tutor contains three types of instructional tasks: *Yield Gap analysis*, *Fertilizer*

Figure 2. A screenshot showing a view of the MIS with various analyses.

Management analysis and *Yield Forecasting* (Amalathas, Mitrovic, Ravan, & Evison, 2010, 2011). Users log in into the MIS, select to use DM-Tutor and are then presented with the problem selection page. From that time, ASPIRE starts to monitor the user's problem-solving actions. Students can select to practice any problem from any of the three types. All tasks are of procedural nature, so each task is divided into a number of problem-solving steps and the student's solution is checked at every step. When the user submits an incorrect solution, DM-Tutor provides feedback to help guide them towards the correct solution.

The problems posed by DM-Tutor contain real-life plantation situations and actual estate conditions. The user needs to access the relevant reports from the MIS to obtain the information required for the current problem. Information on chosen reports is automatically sent from the MIS to DM-Tutor. DM-Tutor then checks whether the user selected the correct reports for the problem, evaluates the solution and provides appropriate feedback. We did not want DM-Tutor to be monitoring all the actions of the user, but only actions related to learning. We feel this is necessary for a just-in-time training tool so as not to disrupt the employees daily work routine involving the MIS.

The three types of tasks in DM-Tutor focus on three different problem areas for the oil palm plantation domain. The Yield gap analysis trains students to identify potential yield (Y_p) and calculate yield gap for the plantation by using the actual yield (Y_a) values found in the MIS. For this task, the student is expected to access various Yield Analysis reports from the MIS and to perform the following steps:

(1) Estimate Y_p for a production area (Estate, Field)
(2) Calculate yield gap for each field/estate (yield gap = $Y_p - Y_a$) and identify the field/estate with the highest yield gap
(3) Recommend a management action to improve yield in the area with the highest yield gap

Figure 3 illustrates the interface users see as they are attempting a problem in the *Yield Gap Analysis* task. The window on the right shows an example of DM-Tutor yield analysis problem, the text of which is: *TH Plantations consist of seven estates. Given that potential yield is the highest yield recorded for a production period, identify potential yield for TH Plantations in the year 2008 and the month of March. Calculate yield for each estate and identify estate with highest yield.*

Users work on their solutions in the solution workspace below the problem workspace. The pane on the extreme right shows the feedback message the user received from DM-Tutor based on the evaluation of their solution. On the left there are two windows: the upper one shows the top view of the MIS, while the lower window shows the detailed MIS report for Yield Analysis, from where users obtain information required to solve the problems. DM-Tutor evaluates the student's solution step by step, immediately after the student submits his/her solution for the current step. Only when the solution is correct, as in the example in Figure 3, the student is allowed to move on to the next step of the problem.

Figure 3. A screenshot of DM-Tutor showing the Yield Gap analysis task.

To solve this problem, the user first needs to identify potential yield. Potential yield is estimated by comparing yield values of the plantation locations for a given time period. The highest yield value produced is considered as the potential yield. In the next step, the user needs to calculate the yield gap for a given plantation location for a certain time period. The user needs to access the relevant Yield Analysis report within the MIS. Next, the user has to calculate yield gap value for all estates to identify the estate having the biggest yield gap when compared to potential yield of the estates. The estate with the highest yield gap would require immediate management attention for yield improvement. Therefore, in the final step of the problem, the user needs to select a suitable recommendation from the list of answers provided to improve yield in that particular location.

The Fertilizer Analysis task focuses on making effective decisions on fertilizer utilization in plantations. For this task, DM-Tutor expects the student to access the Yield Analysis reports and Material Consumption reports from the MIS. The student is required to perform the following steps:

(1) Calculate Partial Factor Productivity (PFP) to identify how much yield is produced for each kg of fertilizer nutrient using the formula:

PFP = Bunch yield (kg/ha) / Fertiliser nutrient (FN) (kg)

(2) Calculate Agronomic Efficiency (AE) to identify how much additional yield is produced for each kg of fertilizer nutrient applied using the formula:

AE = (Bunch yield +FN – Bunch yield 0FN)/ Fertiliser Nutrient (FN)

(3) Given various fields/estates with different palm growth stages, the student has to identify the correct type of nutrient combinations and accurate fertilizer placement.

Given various plantation conditions that reduce the efficiency of the fertilizer/nutrient applied, the student has to provide management actions to resolve the problems. Figure 4 presents an example of a *Fertilizer Analysis* problem in DM-Tutor: *Partial Factor Productivity is used to identify how much yield is produced by each kg fertilizer used and Agronomic Efficiency or AE is used to identify how much additional yield is produced for each kg of fertilizer nutrient applied for an estate. Calculate PFP and AE for Estate Bukit Lawiang for the month of April, 2006. Given yield without fertilizer is 1.10MT/ha. Identify palm age and select suitable fertilizer placement and give supporting reason for each selection.*

The objective of this analysis is to determine whether fertilizer usage in the plantation is within environmental and operational requirements. The student needs to calculate the partial factor productivity (PFP) and agronomic efficiency (AE) for a given plantation location in the MIS for a specified time period. PFP and AE are calculated to determine if the fertilizer input in the plantation is appropriate for the age of the palm trees. The user also needs to identify suitable locations for fertilizer placements according to the

Figure 4. DM-Tutor with MIS showing the problem solving page for Fertilizer Analysis task.

specified palm tree age. In the example shown in Figure 4, the user has made incorrect fertilizer placement choices for two fertilizers, *Urea* and *Borate*. Therefore DM-Tutor has provided a feedback message informing the user of the mistake and suggesting the user trying again or requesting more feedback from DM-Tutor.

The Forecasting Analysis trains the students to carry out yield forecasting using the data available in the MIS. To do yield forecasting analysis in DM-Tutor, the student is expected to access yield analysis reports for different periods of times and also access various information from the reports. The student needs to perform the following steps:

(1) For a given estate/field within the MIS, calculate yield forecast for a future period based on the current yield of the plantation area:

   Yield forecast = current yield (MT/Ha) * future plantation size (Ha)

(2) Compare the calculated yield forecast to the actual yield for the new period and analyse the difference between them. For a given low production period of an estate/field, students will be required to analyse conditions that caused low yield production. Students will be evaluated on the accuracy and efficiency of accessing the relevant reports and information from the MIS to support his/her answer.

(3) For each problem condition, the user has to identify a management action to improve yield in the given plantation location.

Figure 5 illustrates an example of the Forecasting Analysis task in DM-Tutor: *Based on yield for October 2006 for Estate Ladang Belian, forecast yield for October 2007.*

Figure 5. DM-Tutor with MIS showing the problem solving page of the Fertilizer Analysis task.

Estate Belian suffered the problem when FFB arrived at mill, they were found to be overripe making yield lower than forecasted. Identify the relevant reports and information that has to be accessed to understand the situation. Give your recommendation on how a bigger yield could be obtained. In the situation shown in Figure 5, the user has successfully calculated the forecasted yield value for the given estate. In the next problem-solving step, the user is given a problem scenario where the fruit bunches from Estate Belian were found to be overripe when they reached the refinery mill. For this problem-solving step, the user has to determine which information she/he requires from the MIS to identify what caused the problem and also provide a recommendation on how to prevent similar problems from happening in the future.

4. Pilot Study

We conducted a pilot study of DM-Tutor in February 2010 with a group of employees working for an oil palm plantation company in Kuala Lumpur. The purpose of this study was to gain feedback on DM-Tutor, so that any technical issues, interface problems and system usability issues could be identified and solved before the full evaluation. We informed the employees two weeks before the pilot study was planned to take place. We requested their participation to test the usability of DM-Tutor as a training system for the oil palm decision-making domain. Participation was on a voluntary basis and the employees could withdraw from the study at any time, without any adverse effect on their employment. We wanted them to understand that the purpose of the study was to evaluate the system and not their work performance. As the participants were employees at the executive, management or management trainee level, they were familiar with the MIS because they use a similar version of the system for their work.

We provided the participants with a demonstration of DM-Tutor, describing the different tasks contained inside DM-Tutor, problem selection and the various levels of feedback messages available. The participants were given a pre-test before interacting with the system. The pre-test contained three questions, one for each type of tasks in DM-Tutor. Participants spent 15 minutes doing the pre-test, after which they were asked to interact with the system for one hour. Lastly, they worked on the post-test for another 15 minutes and also completed the questionnaire. The post-test also consisted of three questions, of similar complexity to those in the pre-test. The questions used in the tests were shorter and simpler versions of the actual problems in DM-Tutor.

Out of the 22 participants who initially volunteered, only 19 stayed through the whole study. Table 1 presents the basic statistics from the pilot study. On average, the participants interacted with the system for about 27 minutes. Only three participants interacted with the system for more than one hour. The possible reason for this could be that the study was held during office hours and the participants had to complete their pre-assigned daily work load as well. The number of problems attempted by participants is higher than the number of problems they managed to solve. To solve the given problems, the participants made around 38 submissions of answers to DM-Tutor (please note that a submission is a partial answer, which covers only the current step of the task). On

Table 1. DM-Tutor interaction results.

Log data from DM-Tutor	Mean (sd)
Interaction time (min)	27.20 (24.43)
Number of problems attempted	2.85 (1.23)
Number of problems solved	2.05 (1.05)
Number of submissions made	38.05 (8.96)
Number of hints seen	39.95 (19.24)
Pre-test result (%)	50 (16)
Post-test result (%)	78 (21)
Gain (%)	28 (0.15)

average, the participants received 39.95 hints. Only eight participants completed the post-test. Therefore, we only report the pre/post test results for those 8 participants in Table 1.

We found that the performance on the post-test is significantly higher than the performance on the pre-test (t = 1.89, p = 0.005). From the pre-test and post-test results we could observe that the participants' knowledge increased after using the system; the improvement between the post- and pre-test performance is strongly correlated with the time the participants spent with the system (r = 0.92). From the 19 questionnaire responses obtained, we found out that 74% of the participants thought that DM-Tutor was easy to use. 52% of the participants found the feedback messages in DM-Tutor to be helpful for problem solving. When they were asked if they liked DM-Tutor's interface, 63% of the participants said that they did. One of the participants commented that he liked this new version of the MIS.

When asked if DM-Tutor was able to teach them any new plantation decision making analyses, all the participants answered positively. One participant said that she learned from DM-Tutor how to use the information in the MIS to make better decisions. Another participant said that she learned what yield gap analysis and fertilizer efficiency were, from DM-Tutor. When the participants were asked if they felt that by integrating the MIS with DM-Tutor they were able to learn the plantation analyses better, all answered positively. One participant said that he liked to see the MIS and the teaching system together. Another participant stated that she liked the idea that she could stop learning and continue to work with MIS when she needed to and another participant was happy that she could choose the analyses that she needed to learn and that she could check the analysis again if she was not sure.

Overall, the only negative comment about DM-Tutor was that the feedback messages were not always helpful. A number of participants commented that they could not understand some hints provided by DM-Tutor. They also felt that the hints were not descriptive enough. On the basis of those findings, we have modified the feedback messages to make them more explicit and easier for the users to understand. After the pilot study, we made further enhancements to the system. We modified the MIS to send information about the selected reports directly to DM-Tutor so that the students did not have to type the report details into DM-Tutor but rather DM-Tutor knows this by the

report selection actions done by the students in MIS. The enhanced version of DM-Tutor was used in the full evaluation study.

5. Evaluation Study

In October 2011 we conducted the full evaluation study, the objective of which was to evaluate the effectiveness of DM-Tutor. This study was conducted with volunteers from the Bachelor of Agriculture program at the Putra University Malaysia. The students were enrolled in the Industrial Crops I course (AGR3608). The curriculum of this course includes topics such as oil palm plantation industry, plantation management techniques, commercial field plantings, oil palm harvesting, quality control and processing. As the AGR3608 curriculum covers many aspects of oil palm plantation management, the students enrolled in the course have a strong theoretical knowledge related to DM-Tutor. Although the participants were not plantation managers, they had relevant background knowledge accumulated during their studies. Furthermore, being final year students, the participants we chose for the evaluation study are appropriate as they were soon to start working on real plantations. Our participants are therefore comparable to less experienced managers from oil palm plantations.

The students were informed about the evaluation study one week before it was planned to take place. We explained to the students that their participation would help us evaluate the effectiveness of DM-Tutor as a teaching and training tool for oil palm plantation decision making. We wanted to ensure that the students understood that we were not trying to test their knowledge but rather how helpful the system was in training them in plantation analyses and decision making. The participants were informed that their participation would be on an entirely voluntary basis and that they could remove themselves from the study at any time without any adverse effect on their studies. All participants received instruction about the three types of analyses of DM-Tutor in a lecture, one week before the study.

For this study, we divided participants into two groups: the control group used only the MIS to carry out analyses as managers normally would in oil palm plantations, and the experimental group performed the same analyses using the MIS embedded with DM-Tutor. Both groups attempted the same nine problems (three of each type of task). The difference was that the control group solved the problems on paper while accessing reports from the MIS, while the experimental group solved the problems in DM-Tutor. We administered pre- and post-tests, in order to measure students' understanding of the oil palm plantation decision-making before and after training. DM-Tutor recorded all actions participants made and we used the logs to understand their interaction and learning while using DM-Tutor. A questionnaire was used to gain subjective data and comments on DM-Tutor.

At the beginning of the session, each group had a brief reminder about the analyses and was given a demo: the control group saw the demo of the MIS only, while the experimental group was given the demo of the MIS and also of DM-Tutor. The experimental group participants did the pre-test, logged in into the MIS, and then selected

DM-Tutor from the MIS menu. They selected problems from Yield gap analysis, Fertilizer analysis and Forecasting analysis. The participants were requested to complete three problems from each task. After completing the analyses, they proceeded to do the post-test and the questionnaire. The control group participants did the pre-test, logged in to the MIS to attempt the same nine problems the experimental group did but on paper, by using only the MIS. After completing the analyses, they proceeded to the post-test and completed the questionnaire.

The pre-test and post-tests contained three problems each, one from each of the analyses covered by DM-Tutor. The tests were of the same level of complexity. Each question was worth two marks, with the maximum of six marks per test. The pre/post-test questions were designed in consultation with oil palm plantation and the MIS experts. The questions in the pre-test and post-test are different from the problems in DM-Tutor, but are similar in nature. These questions were designed based on the some of the daily operational tasks using the MIS.

6. Results and Analysis

68 volunteers participated in the evaluation study. We had hoped for an even distribution between the experimental and control group, but because the rooms allocated for the evaluation were of unequal sizes and some computers were not in a working condition, we ended up having 28 students in the experimental and 40 students in the control group. The students were requested to be available for 120 minutes for the evaluation study. The statistical analyses of the interaction data are presented in Table 2.

We found no significant difference between the pre-test scores of the two groups, indicating that the two groups are comparable. Both groups improved significantly from pre- to post-test. There is a significant difference between the post-test results of the two groups. The experimental group participants learned significantly more than the control group, proving that the treatment was effective. This behavior is further emphasised when

Table 2. Statistics from the study (standard deviation given in parentheses).

Group	Experimental	Control	Significant
Students	28	40	
Pre-test score (%)	47.5 (10.4)	50.4 (9.7)	no
Post-test score (%)	85.1 (10.5)	63.7 (9.9)	$t=8.4, p<0.01$
Improvement pre-to-post test	$t=13.8, p<0.01$	$t=14.4, p<0.01$	
Gain (%)	37.6 (14.4)	13.3 (5.8)	$t=8.4, p<0.01$
Normalised Gain (%)	70.6 (20.9)	27.2 (10.5)	$t=8.5, p<0.01$
Problem-solving time (min)	96.1 (24.2)	75.6 (5.2)	$t=4.4, p<0.01$
Attempted problems	7.4 (1.6)	7.2 (1.4)	no
Solved problems	6 (2.1)	4 (1.1)	$t=4.6, p<0.01$
Submissions	89.6 (33.9)	N/A	
Hints seen	91.6 (42.9)	N/A	

we analysed gain and normalised gain scores: they show significant differences between the experimental and control group. Please note that we computed normalised gain as gain/(100 – pre-test score). We computed effect sizes (Cohen's d) for both gain and normalised gain, and they are large (1.62 and 1.61 respectively), thus proving the learning effectiveness of our system.

There is a significant difference in the interaction time between the experimental and control groups. Although participants from both groups attempted similar number of problems, the experimental group solved significantly more problems than their peers. Please note that the average number of hints received by experimental group participants is higher than the number of submissions as students sometimes asked for feedback on the All Errors level, which provides hints for all errors made (the student might make several errors in one attempt).

Only 40% of the control group students completed all steps of each problem (the remaining students completed only some parts of problems). It could be that when students in the control group did not know how to answer a question, they stopped attempting that question and moved on to the next problem. The experimental group participants received feedback from the system each time they submitted a wrong solution, were guided by feedback messages to complete the problem correctly and moved on to the next problem. Both groups attempted almost the same number of problems, but there is a significant difference in the number of solved problems. As plantation decision-making is a complex domain for novices, it could be that students in the control group found some problems to be too difficult.

The domain knowledge of DM-Tutor consists of constraints. If constraints represent appropriate units of knowledge, learning should follow a smooth curve with a decreasing trend in terms of constraint violations. We analysed the student logs to identify relevant constraints for various problem states. We then calculated, for each participant in the experimental group, the probability of violating individual constraint on the first occasion of application, the second occasion, and so on and averaged the probabilities across all the constraints and all students. The resulting learning curve is shown in Figure 6.

The graph shows a regular decrease in the probability that can be approximated by a power curve overlaid on the figure and has a very close fit to the data set with an R^2 power-law fit of 0.97. The initial probability of violating a constraint is approximately 19%. This probability can be considered high and due to the fact that even though students did have theoretical knowledge of plantation management aspects, they did not know initially how to apply that knowledge to the actual or real-life plantation management analyses and problems. After four occasions, the probability of violating a constraint dropped down to 11%, close to 50% from the first attempt. This supports our earlier observation that students have improved their knowledge in oil palm plantation management analyses significantly and effectively through interacting with DM-Tutor embedded into the MIS. The graph also confirms our earlier understanding that students in the experimental group did make mistakes when submitting solutions initially but on
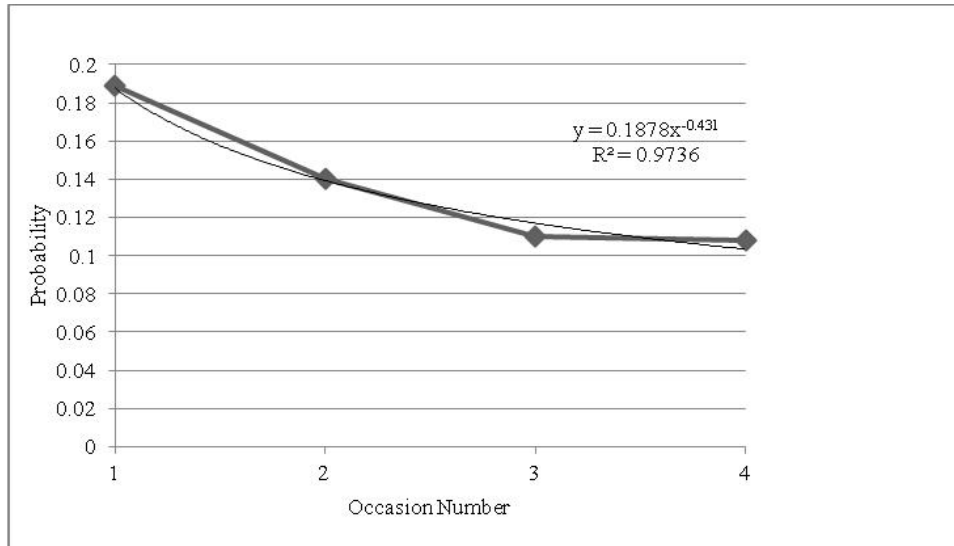
Figure 6. The learning curve for the experimental group.

subsequent attempts managed to submit correct solutions for the problem and moved on to the next problem.

The students from both groups completed a questionnaire at the end of the study. Table 3 gives the mean responses of the participants regarding their experience with the systems. The answers were on a five point scale, ranging from 1 (*Not at all*) to 5 (*Very much*). We used the Mann-Whitney U test to analyse responses. There is a significant difference (U = 319.5, N_C=40, N_E=28, p<0.01) between the groups when we asked if the problem solving questions were easy to attempt. The students in the experimental group found the questions easier to attempt possibly due to the help they received from the feedback messages. There was also a significant difference (U=133.5, N_C=40, N_E=28, p<0.01) between the two groups when we asked if they found the system easy to use. As expected, the students in the experimental group found DM-Tutor easier to use then students who did problem solving using MIS alone. As we can observe from Table 3, students in the experimental group found the feedback messages helpful.

We wanted to know what students thought about the system; therefore we included several open-ended questions in the questionnaire. When we asked them if they liked the

Table 3. Scores from questionnaires.

	Experimental	Control	Significant
Questions easy to attempt?	3.6 (0.8)	2.9 (0.6)	U=319.5, p<0.01
Ease of use	3.9 (0.6)	2.7 (0.6)	U=133.5, p<0.01
Feedback helpful	4.1 (0.6)	N/A	N/A

interface, and 85% of students said they did. A number of students from the control group said they found the MIS to be difficult to use and that it took them a long time to understand what to do. Some students commented that they felt like they needed more time to explore the MIS interface. This is understandable as the MIS and the oil palm plantation management domain is a complex environment.

94% of the experimental group students said that they found DM-Tutor easy to understand, easy to use and helpful in teaching them plantation analyses. When we asked the students if they thought that by using the MIS embedded with DM-Tutor they were able to learn the plantation analyses better, all the students in the experimental group answered positively. We received many positive comments from the students. One student said that MIS integrated with DM-Tutor is a very good idea as he could see how to extract data from MIS and use it for plantation decision-making. Another student said that she has learned from DM-Tutor in a very short time how to understand fertilizer requirement and the feedback in DM-Tutor has been very helpful. Yet another student said that integration of MIS and DM-Tutor makes it easier to understand plantation decision making. Many students said that they thought DM-Tutor was interesting and that it gave good guidance and help for novice users. A number of students said that they felt that it taught them how to understand complex analyses in an easy way with the help of feedback messages and that the problem solving actions were effective in helping them understand plantation decision-making better.

7. Conclusions

By embedding DM-Tutor, a constraint-based tutor, into the MIS for oil palm plantations, we provided scenario-based training using real-life operational data and actual plantation conditions. DM-Tutor is the first constraint-based tutor embedded within an existing live system. Embedding DM-Tutor into the MIS allows us to leverage on the knowledge contained within the MIS for the development of the ITSs domain model.

We conducted two evaluation studies to assess the effectiveness of DM-Tutor. We conducted a pilot study with oil palm plantation employees, which showed that the users have interacted well with DM-Tutor. They have used the information from the MIS to answer the questions posed by DM-Tutor. The participants have also utilized the feedback from DM-Tutor to solve problems. It was interesting to hear from some participants that they learned from DM-Tutor how to perform some of the analyses, as they have not previously been taught formally about them, and their knowledge only consisted of what they found out from experience. The participants found the integration of DM-Tutor into MIS a convenient way to learn.

Plantation managers are busy people, and it was not possible for us to recruit another group of managers to conduct a full study. For that reason, we conducted the full study with a group of final-year students, who were about to start working on real plantations. Those students have learned about plantation management, but lacked practical experience. We gave demos of the MIS to the students and taught them how to access the necessary data for the various types of analyses of interest. The students were divided

into the control and experimental group. The study included pre-test and post-test, system interaction and a questionnaire. Both groups solved exactly the same questions, but the experimental group solved them in DM-Tutor while the control group solved them on paper by using data from the MIS. The results of the full study show that all students improved significantly between pre- and post-test. The large effect size (1.62) shows that DM-Tutor is an effective training tool for the oil palm plantation decision making. Questionnaire responses show that students found DM-Tutor easy to understand and interesting to use. The students were happy to learn various plantation analyses from DM-Tutor and were confident that by using DM-Tutor they would be able to understand real-life plantation decision-making.

Previous tutors including Geometer Tutor, Excel Algebraic Tutor, ETS, SBT-AID and xPST monitored and tracked all the actions the users while they use the embedded systems. Tracking all the actions of the user is complicated and at times redundant because not all user actions in an external application are related to their learning. This is evident in all the systems we have reviewed. Students often cancelled work halfway or wandered off to explore the system on their own and the tutoring systems were left not able to monitor these actions. To be able to track user's actions effectively, the above tutoring systems had restricted several possible solution paths for the users to follow and did not allow the users to attempt anything outside the planned solution path. Certain components and functions of the systems discussed above had to be disallowed from being active as the tutoring systems could not track students' actions when they access these components. As our aim was to create integration with an everyday workplace system with extensive functionalities, we needed to ensure that the integration did not block any component of the application system from working in the same manner even after the integration. Users are free to explore the MIS and access any analyses or reports that they wanted to look at. DM-Tutor does not monitor all the actions of the users in the MIS but only the actions related to learning using DM-Tutor. In our integration approach we have used the MIS that contains operational knowledge of oil palm plantation management. Through embedding DM-Tutor with the MIS, we hoped to provide plantation decision making by leveraging on the operational knowledge the MIS contains. DM-Tutor provides tutoring, monitoring of user's learning actions and feedback as an embedded ITS.

Several conclusions can be made from the presented research. Firstly, ITSs aimed at providing on-the-job training should be embedded into the application environment normally used by employees to perform their jobs. The application system should be modifiable to some extent to allow coupling with the ITS. The integration with the ITS should not restrict the functionality of the application by disallowing or blocking any of its components, or demand modifications to the normal way of operation. The ITS embedded into the application system should monitor students learning actions and provide appropriate feedback.

DM-Tutor is the first ITS for the oil palm domain, and also the first constraint-based tutor to be embedded within an existing application system. We have also created a

framework for embedded ITSs and proven its effectiveness in providing scenario-based tutoring and training for a workplace environment. DM-Tutor provides life-long workplace learning, anywhere and anytime.

Acknowledgments

References

Amalathas, S., Mitrovic, A., & Ravan, S. (2009). Towards an ITS for decision-making on managing oil palm plantations. *Workshop Proceedings ICCE 2009* (pp 157–162), Hong Kong: APSCE.

Amalathas, S., Mitrovic, A., & Ravan, S. (2010). Intelligent Tutoring System for the oil palm industry. *Proc. Int. Oil Palm Conference (IOPC)* (pp. 245–267). Indonesia: IOPRI.

Amalathas, S., Mitrovic, A., Ravan, S., & Evison, D. (2010). Developing an Intelligent Tutoring System for palm oil in ASPIRE. In S. L. Wong et al. (Eds.), *Proc. 18th Int. Conf. Computers in Education* (pp. 101–103). APSCE.

Amalathas, S., Mitrovic, A., Ravan, S., & Evison, D. (2011). Evaluation of DM-Tutor, an ITS for training on plantation decision making. *Proc. ICCE 2011 Workshop on Applications of Information and Communication Technologies in Adult and Continuing Education* (pp. 577–584). APSCE.

Anderson, J. R., & Reiser, B. J. (1985). The LISP Tutor: It approaches the effectiveness of a human tutor. *Byte*, *10*(4), 159–175.

Cheikes, B. A., Geier, M., Hyland, R., Linton, F., Rodi, L., & Schaefer, H. P. (1998). Embedded training for complex information systems. *Artificial Intelligence in Education*, *10*, 314–334.

Genesereth, M. R. (1979). The role of plans in automated consultation. *Proceedings of the 6th Int. Joint Conference on Artificial Intelligence* (pp. 311–319).

Gonzalez, C., Burguillo, J. C., & Llamas, M. (2007). Intelligent Tutoring Systems and Health Information Systems. *Proc. 18th Int. Workshop on Databases and Expert System Applications* (pp. 633–637).

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *Artificial Intelligence in Education*, *8*, 30–43.

Mathan, S. A., & Koedinger, K. R. (2002). An empirical assessment of comprehension fostering features in an intelligent tutoring system. In S. Cerri, G. Gouarderes & F. Paraguacu (Eds.), *Proc. 6th Int. Conf. Intelligent Tutoring Systems* (pp. 330–343).

Mitrovic, A., & Martin, B. (2000). Evaluating the effectiveness of feedback in SQL-Tutor. In Kinshuk, C. Jesshope & T. Okamoto (Eds.), *Proc. Int. Workshop on Advanced Learning Technologies* (pp. 143–144). Palmerston North, New Zealand.

Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a Constraint-Based Tutor for a Database Language. *Artificial Intelligence in Education*, *10*(3-4), 238–256.

Mitrovic, A. (2003). An intelligent SQL-Tutor on the Web. *Artificial Intelligence in Education*, *13*(2-4), 173–197.

Mitrovic, A. (2005). The effect of explaining on learning: A case study with a data normalization tutor. In C.-K. Looi, G. McCalla, B. Bredeweg & J. Breuker (Eds.), *Proc. Artificial Intelligence in Education* (pp. 499–506). Amsterdam, IOS Press.

Mitrovic, A. (2012). Fifteen years of Constraint-Based Tutors: What we have achieved and where we are going. *User Modeling and User-Adapted Interaction*, *22*(1-2), 39–72.

Mitrovic, A., Martin, B., & Suraweera, P. (2007). Intelligent tutors for all: Constraint-based modeling methodology, systems and authoring. *IEEE Intelligent Systems*, *22*(4), 38–45.

Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., & McGuigan, N. (2009). ASPIRE: An Authoring System and Deployment Environment for Constraint-Based Tutors. *Artificial Intelligence in Education*, *19*(2), 155–188.

Mitrovic, A., McGuigan, N., Martin, B., Suraweera, P., Milik, N., & Holland, J. (2008). Authoring Constraint-based Systems in ASPIRE: A Case Study of a Capital Investment Tutor. *Proc. ED-MEDIA 2008* (pp. 4607–4616).

Ohlsson, S. (1992). Constraint-based Student Modeling. *Artificial Intelligence in Education*, *3*(4), 429–447.

Ravan, S. (2007). MIS for Palm Oil. RANN Consulting. http://www.rann-consulting.com

Risco, S., & Reye, J. (2009). Personal Access Tutor-Helping students to learn MS-Access. *Proc. 14th Int. Conf. Artificial Intelligence in Education* (pp. 541–548).

Ritter, S., & Koedinger, K. R. (1996). An architecture for plug-in tutor agents. *Artificial Intelligence in Education*, *7*(3-4), 315–347.

Shatz, S., Bowers, C., & Nicholson, D. (2009). Conceptual and architectural design process of an SBT-AID training system. *Proc. 18th Conf. Behavior Representation in Modeling and Simulation* (pp. 145–146).

Suraweera, P., & Mitrovic, A. (2004). An intelligent tutoring system for Entity-Relationship modeling. *Artificial Intelligence in Education*, *14*, 375–417.

Van Lehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R. H., Taylor, L., Treacy, D. J., Weinstein, A., & Wintersgill, M. C. (2005). The Andes physics tutoring system: Lessons Learned. *Artificial Intelligence in Education*, *15*, 147–204.

Zakharov, K., Mitrovic, A., & Ohlsson, S. (2005). Feedback Micro-engineering in EER-Tutor. In C.-K. Looi, G. McCalla, B. Bredeweg & J. Breuker (Eds.), *Proc. Artificial Intelligence in Education* (pp. 718–725), ISBN 1-58603-530-4. Amsterdam: IOS Press.