A Project Manager Skill-up Simulator Towards Problem Solving-based Learning

Masanori Akiyoshi[1], Masaki Samejima[2] and Norihisa Komoda[2]
[1]*Faculty of Applied Information Science, Hiroshima Institute of Technology, 2-1-1 Miyake Saeki-ku, Hiroshima, Japan*
[2]*Graduate School of Information Science and Technology, Osaka University, 2-1 Yamadaoka, Suita, Osaka, Japan*

Keywords: Project Management, Skill-up Simulator, Scenario Generation, Association Rule Mining.

Abstract: This paper addresses a project manager skill-up simulator which aims to provide problem-solving learning environment. Project management is inherently human-centric activities, and research work for education has been done by using simulation. The proposed simulator environment is designed to provide well-configured functionalities that make it possible to generate high fidelity scenario on project situation with event rule firing mechanisms, evaluation on a learner's operations as to the system development progress management phase. The proposed environment realizes effective scenario generation and well-insighted evaluation along with a learner's interactive operations.

1 INTRODUCTION

Recently we have faced with the problem occurred in the project management of IT system development such as delivery delay and cost increase, even if development organization and division are well-prepared at the project planning phase. Project managers are strongly expected to handle various situations at the project fields based on deep understanding of "body of knowledge on project management", for instance, PMBOK(PMI, 2009) (Project Management Body Of Knowledge) and so forth. As for the consolidation of "body of knowledge on project management", we have a world-wide used examination by PMI (Project Management Institute) and could evaluate whether project managers have it or not. As for handling various situation at the real project, OJT (On the Job Training) is mostly applied to train such skills. Success and failure factors of projects(Horine, 2005) are also argued. However OJT has drawbacks that it sometimes causes risks against practical projects and depends on the ability of tutors as excellent project managers.

This paper addresses a concrete methodology to enhance problem solving ability of learners using our developed project simulator. The simulator-based learning environment have been proposed for project management(Davidovitch et al., 2007)(Drappa and Ludewig, 2000), for instance, "SESAM"(Drappa and Ludewig, 2000) provides interactive functions to manage the system development project. The simulator just mimics some aspects of real projects and the learners can make task assignment to members in the simulator and receive evaluation of the final result. The learners may judge members' skills through Q&A (Question and Answer) style communication in the simulator. Of course it is important to mimic real project aspects in the simulator, besides we believe more significant factors when using a simulator are to provide intended scenario generation along with the learning objectives set by tutors and detailed feedback on each learner's operations. "intended scenario generation" is not patterned one, but dynamically induced one by using project model and a learner's interactive operation.

This paper proposes such a new type of project manager skill-up simulator which realizes scenario generation mechanism with event rule firing and feedback mechanism concerning a learner's defective operation identification from "reference operation" viewpoints.

Akiyoshi M., Samejima M. and Komoda N..
A Project Manager Skill-up Simulator Towards Problem Solving-based Learning.
DOI: 10.5220/0004140401900195
In *Proceedings of the International Conference on Knowledge Management and Information Sharing* (KMIS-2012), pages 190-195
ISBN: 978-989-8565-31-0

2 PROJECT MANAGER SKILL-UP SIMULATOR

2.1 Specific Features on Problem Solving-based Learning

Project management includes several phases, for instance, planning, progress management, negotiation to clients and so forth. Negotiation involves uncertainty coming from human relationships, so we focus on the phases common to planning and progress management hereafter. As for the target phases, required abilities of project managers are logical thinking-based actions to check usable resources and promised constraints, and make appropriate operations towards the situation. In a sense, PDCA (Plan-Do-Check-Action) cycle is repeated until finalizing problem-solving towards the situation.

Then what is the approach for providing such a PDCA cycle as a scenario? Patterned scenario makes it possible for tutors to set intended learning objectives explicitly. On the other hand learners operations are so restricted to follow the patterned scenario, which means the simulator just generate the same results along with the intended learning objectives and seems not to contribute to skill-up as "problem-solving". Unexpectedness is necessary for learners to do the PDCA cycle with logical thinking, which makes it difficult to guarantee the learning objectives set in advance by tutors.

Based on the above-mentioned features on "problem-solving in the project management", the issues to be addressed are as follows.

- Learning scenarios are generated from the simulator when tutors simply set the learning objectives.
- Tutors get the reference operations as a solution concerning the learning scenario even if they are either optimal or semi-optimal.
- Learners make similar operations which mimic essential ones in real projects under recognizing the learning objectives.
- Learners get appropriate feedback as to their operations based on evaluation of problem-solving process.

2.2 Configuration of PM Skill-up Simulator

In the section 2.1, the PDCA cycle is argued on "planning" and "progress management" as a prerequisite for the simulator. Hereafter our proposed simulator addresses the "progress management" from the issues of "problem solving" viewpoints, because the model and related functionalities are more concrete than "planning" ones. Figure 1 shows the configuration of the project manager skill-up simulator.

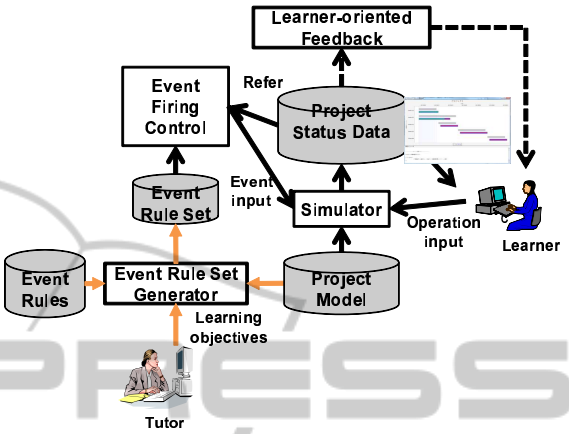

Figure 1: Configuration of PM Skill-up Simulator.

The project model mainly consists of "Project", "Module", and "Person" as they exist in the real project. "Project" defines a set of modules to be developed and members to be assigned to the project. The dependency of modules is depicted as "parent module" and "child module". "Module" defines each estimated man-hour and technical domain such as "database", and its difficulty grade ranked with "A (difficult)", "B (normal)", and "C (easy)". "Person" defines each member's skill grade on technical domains ranked with "A (expert)", "B (normal)", and "C (novice)". These three level description on module difficulty and person skill generates project dynamics, e.g. mismatching between module difficulty and person skill is main cause of schedule delay and quality loss in a project. And the most crucial task of a learner as a project manager is to detect/predict such undesirable situation and make proper operations, for instance, "overtime directive", "collaborative work directive with expert members" and "member assignment change". "collaborative work directive with expert members" is especially found in the real situation of debugging. The process of debugging generates new bugs and some low-skilled level members cannot solve such iterative chain of generating bugs without supervised collaboration. The simulator also models latent bugs which makes it difficult to estimate final product quality. The result is evaluated using indices such as Q(uality), C(ost), and D(elivery), where "Q" corresponds to the total bugs, "C" does the overtime cost, and "D" does the completed date against the initially scheduled date.

What we explained so far is normal situations in a sense, since the project model generates daily project dynamics and operations by a learner are mostly normal reactions if he/she has knowledge on project management such as PMBOK and experiences on real projects such as small delay and standard volume occurrence of bugs. In addition to imitated model of a project, "event rules" are used to make some disturbances towards project dynamics that reflect the learning objective by tutors. For instance, the learning objective is "quality management is tough on the project", then unexpected explosive occurrence of bugs is triggered by the corresponding event rule firing to the situation.

Scenario is generated based on the above-mentioned intertwined interaction of model-driven dynamics, event rule firing, and a learner's operations. Project status is displayed using "Gantt chart panel" and operations are done through "interaction panel" interface.

As shown in Figure 1, "Event rule set generator" and "Learner-oriented feedback " are necessary mechanisms of our proposed simulator to realize unpatterned scenario-based learning. "Event rule set generator" is to generate necessary event rule set from all event rules so that the learning objective is guaranteed. "Learner-oriented feedback" is to detect a learner's defective operations by comparing them to "reference operations" and provide related questions to a learner.

2.3 Event Rule Set Generator

"Event rule" is defined as disturbance to cause unexpected effects on the project status that reflect the learning objectives set by a tutor. Figure 2 shows the descriptive forms for depicting such a rule. For instance, if the learning objective is concerning "Quality Management", the simulator should generate the situation on "quality" even if a learner makes appropriate operations and vise versa. As indicated in Figure 2, descriptive parts are combinations of each part items, that is, 27 rules are used. Of course all the rules are not necessary to guarantee the project status, which means the number of rule set is 2^{27}. There inherently exist so many variances as to the project status in the simulator under intertwined interaction of project model dynamics and a learner's operation. Therefore it can not be feasible to set such a rule set by hand.

Figure 3 shows our proposed mechanism to induce such a rule set by using agent programs that perform various operations against the project status. Agent programs are characterized to make either

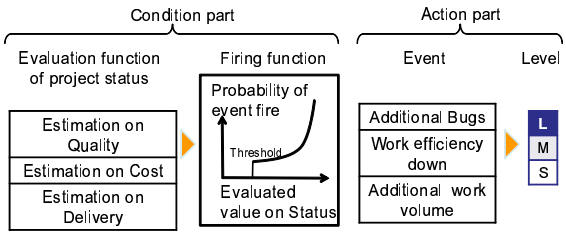

Figure 2: Description of Event Rule.

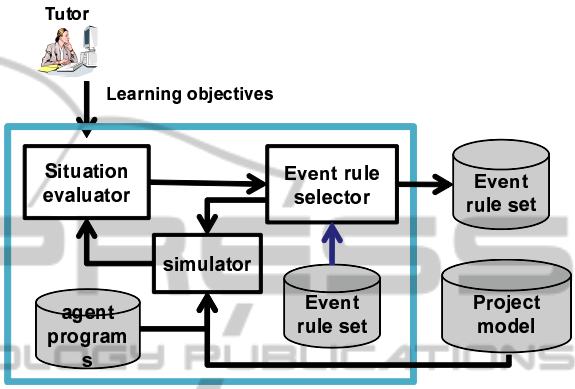

Figure 3: Event Rule Set Generator.

proper or improper operations for the learning objective. First "Event rule selector" sets a certain set of event rules. Then by using "project model", "agent programs", and the "selected rule set", the simulator outputs the project status log and "QCD (Quality, Cost and Delivery)". "Situation evaluator" judges whether the intended project status is generated along with the learning object which gives a score on the "selected rule set". Again a pair of "selected rule set" and its score is inputted to the "Event rule selector". This repetitive process continues until one "rule set" as optimum is decided. To realize this optimization, GA(Genetic Algorithm) is used.

2.4 Learner-oriented Feedback

The project result from "QCD" viewpoints is output of the simulator which is derived from a learner's operations. Simple evaluation of a learner is done by using these indices, however, it does not lead to skill-up from "problem solving" viewpoints. Excellent project managers are considered to have judgment criteria, so called "project management principle", and make proper operations(hereafter, we call them "reference operations") along with it. Therefore acquisition of such criteria is positioned as one of the goals when providing the "problem solving-based learning". If a learner makes some mistakes on his/her operations with lack of such criteria, the

learning simulator should point out the mistakes and give related questions to confirm his/her knowledge on it. The "reference operations" are derived from applying the "project management principle" to the project status in the simulator. So this paper assumes that such "project management principle" and "reference operations" are known in advance, to focus on the feedback problem to a learner.

2.4.1 Mechanism Overview

The feedback consists of two mechanisms, e.g. "identification mechanism on learner defective operations" and "question generation mechanism". Figure 4 shows the configuration of the feedback, using a learner's log and a reference operation-induced log. The log includes several set of results with project models, because "a learner's defectiveness" is detected across several project models.

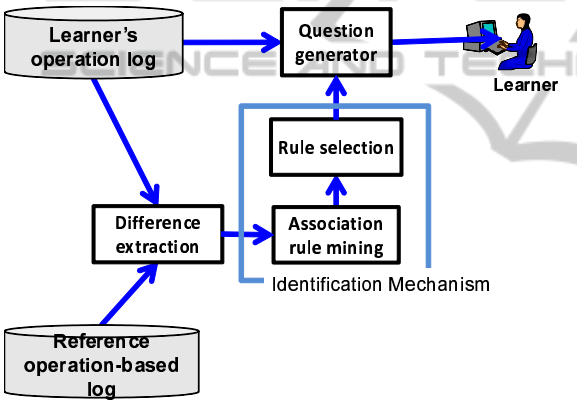

Figure 4: Overview of Learner-oriented Feedback.

2.4.2 Difference Extraction

The difference of operations is done for each module log in which the following items are used.

- the times of "overtime directive" and "collaborative work directive"
- first operation timestamp

The predicates are "more/same/less" for the times and "late/not late" for the timestamp.

The data representing the difference includes the above items, module name, operation type, predicates, and the project status attributes indicated in Table 1.

2.4.3 Identification Mechanism

There exist three types of defective operations; 1) defective ones frequently happened in several project models, 2) defective ones frequently happened in a certain project model, and 3) defective ones happened by chance. The defective operations frequently happened in several project models are the target operations which are caused by the lack of knowledge on "project management principle".

Table 1: Project Status Attributes.

Name	Value
module difficulty	three level; "A", "B", "C"
module work volume	real number
skill mismatch	logical value; true, false
delivery delay	logical value; true, false
is on critical path	logical value; true, false

Figure 5 shows the process flow for identifying our target operations. Association rule mining process generates association rules between "project status attributes" and "difference predicates", which means applying these rules makes it possible to filter "defective operations happened by chance". Then among the generated association rules, exact matching ones to "project management principle" are extracted. The extracted rules makes it possible to filter defective operations frequently happened in a certain project model.

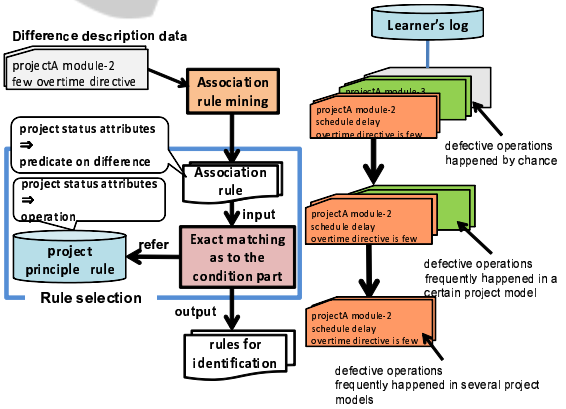

Figure 5: Process Flow of Identification.

2.4.4 Question Generation Mechanism

Questions related to defective operations frequently happened in several project models are generated by using templates with "choice list style". The example on output dialogue from the simulator is shown in Appendix A.

3 EVALUATION

3.1 Result of Event Rule Set Generation

To verify the intended scenario generation by our proposed mechanism, we assume the following condition to the simulator. A tutor sets "quality loss project" as the learning objective, and 5 modules and 3 members are used as the 30 days-long project model. Table 2 shows the generated event rules on the descriptive parts.

Table 2: Generated Rules on Quality Loss.

No.	Evaluation function	Event	Level
1	Estimation on Quality	Additional bugs	M
2	Estimation on Cost	Work efficiency	S
3	Estimation on Quality	Additional bugs	L

As indicated in Table 2, the part of "Evaluation function" is used as to "Estimation on Quality" that means disturbances from bugs management are triggered when "quality" is well-maintained. Therefore this will make such a situation to care about "quality" and do some operations for it. We also evaluate how many times of the "quality loss" project status happens in the project as to agent programs entirely. Average times of such project status is 2.67, and it seems to be reasonable for the reason that "quality loss" status happens every 10 days in the 30 days-long project.

3.2 Result of Learner-oriented Feedback

We used 10 project models for 2 learners; learner-A and learner-B. Investigation by hand detected 3 defective operations as to the learner-A and 2 defective operations as to the leaarner-B. Then our proposed mechanism extracted rules as shown in Table 3.

Table 3: Extracted Rules.

	learner-A	learner-B
initial association rules	99	38
rules for identification	2	1

Table 4 shows the identified rule and the number of generated questions. As indicated in Table 4, unidentified defective operation still remains; "adequate skill against the module $\Rightarrow$ collaborative work directive is many". This was caused by the reason that various action part descriptions for the condition part "adequate skill against the module" existed in the log, which causes filtering such association rules by our proposed mechanism.

Table 4: Generated Questions.

learner	rule(number of question)
A	low skill against the module $\Rightarrow$ collaborative work directive is few (7)
	low skill against the module $\Rightarrow$ collaborative work directive is late (6)
B	low skill against the module $\Rightarrow$ collaborative work directive is late (5)

4 CONCLUSIONS

This paper addresses a project manager skill-up simulator which aims to provide problem-solving learning environment. The distinctive features of the simulator is to provide un-patterned scenario generation along with a learning objective and feedback as to a learner's defective operations from the lack of knowledge on "project management principle". Experimental results shows our proposed functionalities does work and make sense from "problem solving-based learning" viewpoints. Our future work is how to modify the scenario generation and feedback along with the progress of a learner's study.

ACKNOWLEDGEMENTS

This work was partially supported by KAKENHI; Grant-in-Aid for challenging Exploratory Research (23650537).

REFERENCES

Davidovitch, L., Shtub, A., and Parush, A. (2007). Project Management Simulation-Based Learning For Systems Engineering Students. In *Proceedings of International Conference on Systems Engineering and Modeling 2007*, ICSEM '07, pp.17–23.

Drappa, A. and Ludewig, J. (2000). Simulation in software engineering training. In *Proceedings of the 22nd international conference on Software engineering*, ICSE '00, pp.199–208.

Horine, G. (2005). *Absolute Beginner's Guide to Project Management (Absolute Beginner's Guide)*. Que Corp., Indianapolis, IN, USA.

PMI (2009). *The PMBOK Guide*. Project Management Institute, fourth edition.

APPENDIX

```
The project status derived from your
operation at "module 4" of "project 2"
is shown below.
Which operation is the most proper one
to improve your operation?
Please choose it from the choices.

[Your operation]
overtime directive;
      7th day, 10th day, 13th day

[Your result]
Bugs;
 4.58567
Cost;
 3.5
Developpment term;
 14

[Choices]
1. 4th day; progress check
   8th day; overtime directive
  10th day; progress check

2.11th day; overtime directive &
            progress check
  12th day; progress check

3. 4th day; overtime directive
   7th day; overtime directive &
            progress check

4.12th day; overtime directive &
            progress check
  15th day; overtime directive
```