Technology Enhanced Learning(TEL) Environment to Develop Expansionist-Reductionist(ER) Thinking Skills through Software Design Problem Solving

Deepti Reddy
IDP-Educational Technology,
Indian Institute of Technology
Bombay(IITB),
Mumbai, India
deeptir@iitb.ac.in

Sridhar Iyer
IDP-Educational Technology,
Indian Institute of Technology
Bombay(IITB),
Mumbai, India
sri@iitb.ac.in

Sasikumar M.
CDAC,
Mumbai India
sasi@cdac.in

Expansionist-reductionist(ER) thinking skills are important in solving ill-structured problems. Expansionist thinking is the process of exploring all possibilities, and reductionist thinking is systematically choosing from the repertoire of options. Research shows that lack of ER skills may lead to narrow problem definition or weak solution design. Problem solving is one of the important competencies that an engineer should acquire to sustain in industry. Thus, it is worth training undergraduate engineering students in ER thinking skills. Our research focus is to teach undergraduate computer engineering students in learning ER thinking skills in the context of solving software design problems. In this paper, the design of the TEL environment for ER thinking skills is discussed. The study was done with second year undergraduate engineering students to investigate the effectiveness of the TEL in learning ER skills. The results showed significant improvement in ER skills in students after the training.

Keywords—expansionist-reductionist skills, software design, prompts, adaptive feedback, TEL system.

I. Introduction

Problems are classified on a continuum from well-structured to ill-structured problems. In well-structured problems, the variables are well defined and one has to apply the known formulas or principles from the domain to solve the problem. In ill-structured problems, the goals are not well defined, possess multiple solutions and criteria to evaluate and select solution, are not clearly stated. The skills to solve well-structured problem is different from the skills to solve ill-structured problems [8].

The Expansionist-Reductionist(ER) thinking skills are important in solving ill-structured problems. Expansionist thinking is expanding the thinking by exploring all possible options and reductionist thinking is systematically choosing from the repertoire of options. In ill-structured problems both the problem space and solution space are not well defined. The ER thinking in problem space involves understanding the problem as a whole and eventually reduce to defining the sub-problems to be solved [1]. In solution space, the ER thinking involves exploring all possible alternative solutions and select single solution based on selection criteria suitable for the given problem [2].

Research shows that the quality of problem formulation improves by applying the ER thinking skills. The lack of ability to understand the problem as a whole leads to defining the problem narrowly. If the problem definition is narrow, the solution space reduces, and the tendency to solve a wrong problem increases [17].

Problem solving is an important competency for an engineer to sustain in industry. The problems encountered in industry include engineering design problems, where a product or a system has to be designed to solve real life problems. The engineering design problem characteristics are similar to ill-structured problems. Applying the ER thinking skills in solving engineering design problems will be effective in improving the design outcomes [7].

Our research focus is teaching and learning of ER thinking skills to undergraduate engineering students in the context of software design problem solving. The TEL environment is designed to enable the learning of skills in the process of solving software design problem in Data Structures domain. The Research Question (RQ) investigated here is “How effective is the TEL environment in teaching and learning of the ER thinking skills?” The study was performed with undergraduate computer engineering students from Mumbai University to investigate the RQ. The results showed significant improvement in ER thinking skills in students in the context of solving software design problem.

The literature on ER thinking skills is discussed in the next section; integration of ER in the software design problem and challenges in teaching ER skills is discussed in sections 3 and 4. In section 5, the design of the TEL system for ER thinking is explained in detailed followed by the study, discussion and conclusion.

II. Expansionist-reductionist(ER) thinking skills

The literature on creative problem solving suggests various techniques to solve ill-structured problems. The expansionist-reductionist approach is used to effectively formulate the ill-structured problem into well-defined sub-goals, which are directly solvable [6, 19]. Similarly, ER approach is used to design a solution by generating potential alternative solutions, and systematically choose a solution based on the selection criteria [2].

It is argued in creative problem solving literature that quality of the problem formulation depends on the ability to understand the whole problem space, and then decompose the problem into subcategories [1]. This process is categorized as expansionist-reductionist [19] approach in which the problem space is expanded before reducing (decomposing) the problem into sub-problems. Expansionist approach involves understanding the system as a whole by identifying the parts and interrelationship between the parts [1].

Several techniques are used to represent and expand the problem space. The Strategic Options Development and Analysis (SODA) technique [5] relies on the concept of cognitive mapping in order to explore a problem area. A cognitive map is used to represent a person’s thinking about an issue, and is basically a directed graph, consisting of

978-1-7281-1143-8/18/$31.00 ©2018 IEEE
DOI 10.1109/T4E.2018.00044

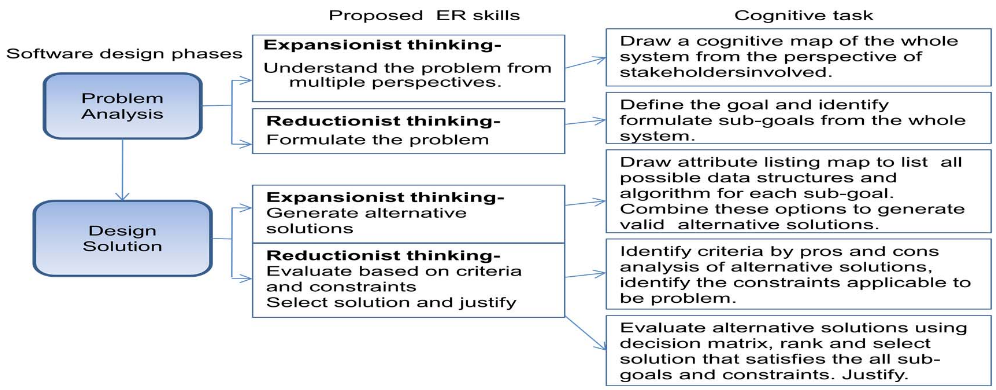

Fig. 1. Software design sub-skills identified on integration of expansionist-reductionist thinking skills and cognitive tools used and tasks performed for each sub-skill.

nodes (ideas) and arcs (connections between ideas). The MACRAME tool is used by analysts to model the pictorial representation of the problem, considering the viewpoints of various stakeholders involved in a problem. Problem is formulated using diagrammatic structure of interactions among actors from general level to specific level [12]. In another study, the expansionist approach was initiated by exploring the problem space by asking a question. For example, if the problem statement is "How might we retain new customers?", the broadening question would be, "Why would we want to retain new customers?", and the narrowing question would be, "What is stopping us from retaining new customers?" Results showed significant improvement in quantity of problem statements generated; however, there was no significant difference observed in quality and uniqueness of the problem statement [6].

A solution space is expanded by applying divergent thinking techniques like brainstorming, attribute listing, analogous thinking, etc. to generate alternative solutions. The process of reducing the solution space is to evaluate and select single solution based on the selection criteria. The convergent thinking techniques are used to evaluate the alternative ideas on various criteria, and select an appropriate idea using decision matrix [18], pros and cons analysis [10], analytical hierarchy process [16], etc.

Studies have shown that even professionals lack the ability to apply ER skills in solving ill-structured problems, and tend to formulate the problem narrowly. This results in design of sub-optimal or wrong solution [2, 6]. ER skills are important in the context of solving engineering design problems; however, there are very few studies focusing on teaching and learning of ER skills for undergraduate engineering students. Our research focus is in teaching and learning of ER skills to undergraduate engineering students in the context of software design problem solving.

III. Integration of ER Thinking Skills into Software Design Process

The software design problems are ill-structured problems, for example, in a problem such as "design a software system for a bank", the goals and sub-goals, in terms of data to be stored and operations to be performed by the software system, are not clearly defined. The problem can be solved in multiple ways, and multiple solutions exist.

We have integrated the ER approach in the process of software design to improve the quality of the problem formulation and solution design. The software design phases considered for integration of ER skills are problem analysis and solution design. We have adapted the ER techniques proposed in creative problem solving literature into software design process.

In problem analysis, the expansionist thinking is initiated by understanding the problem from multiple perspectives, and diagrammatically representing the whole system in terms of entities and interactions among them. The reductionist thinking consists of identifying the data to be stored and operations to be performed, based on the goal to be achieved. In design solution phase, expansionist thinking is the process of generating alternative solutions using combination of various data structures and algorithms. The reductionist thinking is evaluating and selecting the single solution based on the constraints and criteria. The integration of ER thinking in software design process and tasks performed for each sub-skill is summarized in fig. 1.

IV. Challenges in Teaching-Learning of Expansionist-Reductionist Thinking Skills

The research on ill-structured problem solving shows that students lack the ability to apply appropriate cognitive skills while solving complex problems, which leads to weak formulation of problem and solution design [3, 20].

Based on empirical evidence, cognitive and metacognitive prompts are effective in developing metacognitive skills like setting goals, planning, monitoring in the process of learning a concept or problem solving [20, 3]. Feedback plays an important role in identifying the lacuna, and taking actions to improve on the task [11].

The first version of the TEL system for teaching ER skills in the context of solving software design problems named as Fathom, was designed for undergraduate engineering students [4]. The features were the following.

1. Cognitive and metacognitive prompts to systematically guide students through the phases of problem solving.
2. Cognitive activities to be performed were explained with examples.
3. Domain specific hints
4. Cognitive tools to perform activities: pros and cons table, decision matrix.

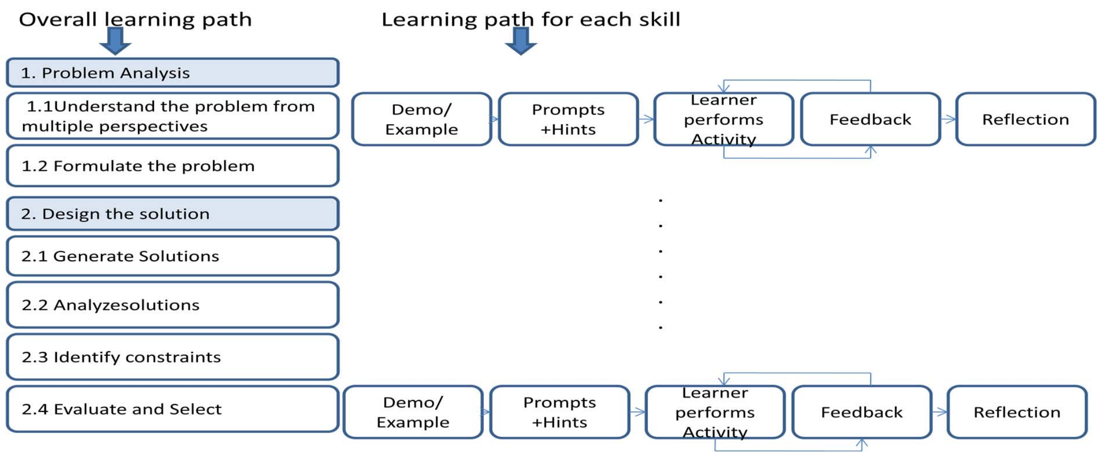

Fig. 2. Design of learning activities in Fathom

The study was done to investigate the effectiveness of Fathom with second year undergraduate engineering students. The research study showed that some of the activities were difficult to understand. This resulted in low scores (mean 1.5/3) for activities such as: understand the problem, identify goal and subgoals, identifying constrains, select solution using decision matrix and justification.

In the second version of TEL system, more scaffolds were added to improve the quality of the responses.The design of the improved version of the TEL system: Fathom-V2, for teaching and learning of expansionist-reductionist(ER) thinking skills is discussed in the next section.

V. Design of Fathom-V2: TEL for ER skills

The TEL system is designed based on the pedagogical principles suggested in literature on computer-based scaffolds for developing cognitive skills and metacognitive skills [3, 20], and TELoTS framework [15]. The pedagogy is based on learning by doing and reflection activities, as stated by Schon, "*The student cannot be taught what he needs to know, but he can be coached*."

The ER activities under each problem solving phase- Problem Analysis and Solution Design are listed in Fathom on the left hand side of the screen, as shown in fig. 2. These are clickable buttons, which prompt the learners to perform the ER thinking activities. In each learning activity, the scaffolds are as follows.

- Question prompt to guide the thinking process towards the targeted skill [20].
- Cognitive activities to be performed are explained and demonstrated with videos, examples and hints [3].
- Adaptive feedback is generated based on the performance of the learner in applying the targeted skill. If the performance is good then the positive feedback is generated, else the feedback addresses the corrective actions to be taken to fulfill the gap between learner's competency and desired competency level.
- After the activity, in-action reflection is done to allow the learner to reflect on how much the activity helped in achieving the targeted skill. Learners are posed with a test question in which a new solved problem is shown as a case study, and is asked to rate the response. This activity ensures that learners reflect on their learning of the targeted sub-skill, and apply it to evaluate a new problem.

The design of the overall learning activities in Fathom-V2 is shown in fig. 2.

The design enables the learner to see the overall path to be followed to solve the problem, as shown in the left hand side in fig. 2. In each step, the learning activities are designed to learn the targeted skill. The learning activities for each sub-skill are explained below.

A. Learning Activities in Fathom

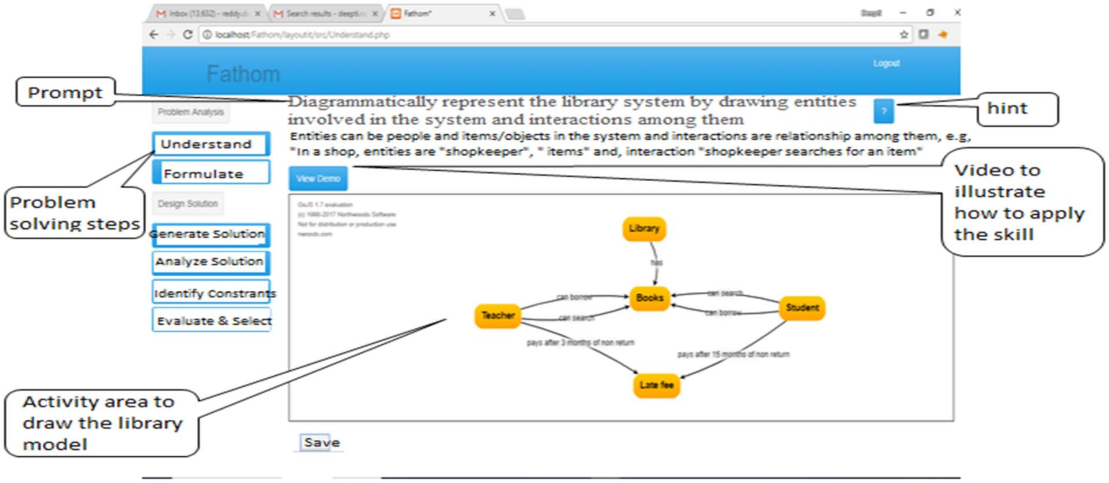

Fig. 3. Activity page for understand the problem from multiple perspective.

After the learner logs into the Fathom, a training problem is posed. The problem posed is a library management problem, stated as *"The college library maintains books on various subjects taught in various departments like arts, science and commerce. The library staff is involved in issuing books to students and teachers. The students need to return the book in 15 days while teachers can return in three months. The library staff maintains records of all books in the library including issue of books, return of books and collection of late fee if the book is not returned in time. Librarian decides to provide online service for students or teachers to search availability of books in the library. Your task is to design a system to solve the above problem using appropriate data structure and algorithm."*

1) Understand the problem activity

On clicking the "understand the problem" button on left hand side menu, the activity page is shown with the following scaffolds: demo, hint, prompt and activity area (fig. 3).

Prompts and hints: The scaffolds are designed to direct the learner's thinking towards understanding of the problem from the perspectives of stakeholders and entities involved in the system.

The prompt to trigger the process of understanding the system is *"Diagrammatically represent the library system by drawing entities involved in the system, and interactions among them"*. The hint button "?" gives detailed explanation of what is entity and interactions with examples. The "view demo" button will play a video demonstrating the activity for a shop inventory problem.

Activity area: The activity area is designed to allow the learner to draw the cognitive map of the library problem. The nodes represent the entities or stakeholders, and the links between the entities represent the interactions among stakeholders and entities, as shown in fig. 3.

Feedback: On clicking the save button, the system will evaluate the response based on the number of entities and interactions identified, and generate a feedback. For example, if more than 5 nodes and interactions are identified, then the positive feedback generated is *"Excellent, you have identified 6 entities and 8 interactions"*; however if few(1-2) nodes and interactions are drawn then the corrective feedback generated is *"Look at the highlighted nouns(yellow color) and verbs(green color)in the given problem, and check if you have missed some of the important stakeholder/entities and interactions among them."* The system will highlight the nouns and verbs, and prompts learner to add them into the diagram.

Reflection: The reflection questions are asked to reflect on the skill by evaluating response for a new problem. This process of evaluating response for a new problem will enable learner monitor his/her own thinking.

2) Formulate problem- Goal and sub-goals.

a) Goal

Prompt, Hint, Demo: The prompt is *"Write the broad goal to be achieved."* The hint shows the example for the shop inventory problem.

Activity: The textbox is provided to write the goal.

Self-evaluation, Feedback: On saving the response, the system shows a list of answers at various levels (bad, average, good and excellent) as shown below.

Self-evaluation activity- Goal

Choose one of the options below which matches closely to your response

1. Design library system.
2. Design library software system for teachers and students
3. Design software system for library for teachers and students to search availability of books in the library.
4. Design software system for library to make it convenient for teachers and students to search if the book is available in library.
5. None of the above

If learner selects option 1, the feedback generated is as follows.

Feedback- Your goal is too broad, it can be improved by incorporating following points. i) Who are the primary users or for whom the system is being designed? ii) What is the primary purpose of the system? Rewrite the goal above.

b) Sub-Goals

Prompt, Hint: The prompt for the sub-goal is *"In the diagram below, identify the entities and interactions for achieving the goal and write as sub-goals"*. The library system's cognitive map is shown, and the learner is prompted to identify the entities and interactions to be included as functionalities in the software system (fig. 4).

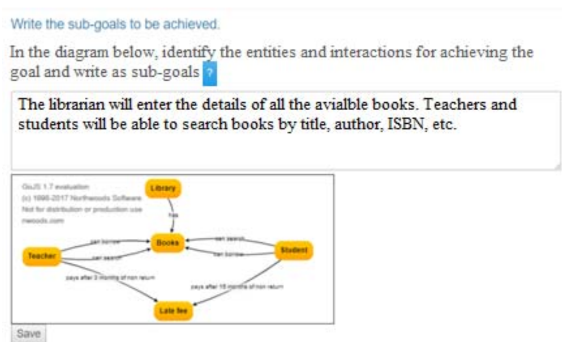

Fig. 4. Activity to write subgoals

Self-evaluation, Feedback: After the user saves his/her response, the self-evaluation activity is posed with list of possible sub-goals (correct, incorrect), and the user is asked to select the one he/she has identified. Based on the selection, appropriate feedback is generated by the system as shown below.

Self-evaluation activity - Sub-Goal

Choose one of the options below which matches closely to your response.

1. Teachers/students will be able to search for the availability of books
2. Librarian/staff will insert books in to the system
3. Librarian will be able to update availability of the books when the book is issued/returned.
4. Librarian will be able to calculate the fine and if the book is returned after due date.
5. Others(not listed above)

If option 1, 2 are chosen

Feedback- Very good, you have defined sub-goals from the point of view of the teachers/students and librarian. But, you have missed some of the operations from the librarian point of view. Complete the activity in the step1- Understand the problem, and then identify operations to be included in the sub-goal.

3) Generate solutions activity

This activity is divided into two parts; the first activity is drawing the attribute listing map, and the second activity is to generate multiple solutions.

a) Attribute listing map- Prompt, Hint, Demo

Prompt, Hint: The prompt posed is, *"Create the attribute listing map by identifying the attributes of the design and listing various options."* The prompt is supported by a hint and demo to help in listing the key attributes (data structures and algorithms)and its values (array, list, tree, linear search, etc.),and generate multiple designs by modifying the value of attribute or combining two or more attributes for each solution.

Activity: The drawing tool is provided to the user to draw the attribute list map as shown in fig. 5.

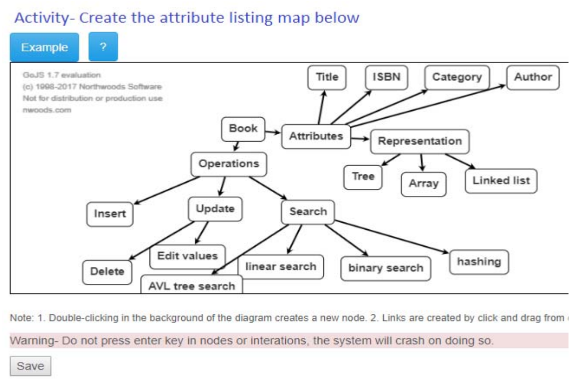

Fig. 5. Attribute list map

b) Generate solutions

Prompt, Hint: The prompt is *"Generate multiple solutions by changing or combining the values of attributes."* The hint shows the examples of solutions generated for a shop-inventory problem.

Activity: The text box is provided to write solutions, and button to add more solutions. The solutions generated by the user are shown in fig. 6.

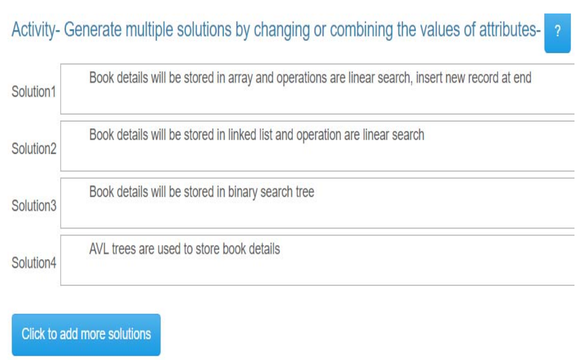

Fig. 6. Generate solution activity

Feedback: If the user has generated less than 3 solutions, then the attribute listing map is shown, and the feedback generated is *"You have generated only 1 solution. Refer attribute listing map to generate more valid solutions using various data structures and operations."*

4) Analyze solutions

Prompt, Hint: The prompt provided is “Identify criteria to evaluate the solutions” and the hint given is “Criteria are common parameters used to compare multiple solutions. For example, execution time is one of the criteria.”

Feedback: If the user has identified less than one criteria, then the feedback generated is *“You have identified only 1 criterion. Perform pros and cons analysis(use pros and cons table) of the solutions, and then identify the common criteria.”*The pros and cons table is preloaded with solutions and user is asked to write advantages and disadvantages in the table, and identify the common criteria based on the analysis.

5) Identify constraints

Prompts, Hint: The prompt is *“Identify the constraints on the criteria to be achieved in the above problem”.* The hint given is *“Constraints are the mandatory conditions to be achieved in the given problem. Constraints are value (low, high) associated with the criteria. For example, if execution time is a criterion, then constraint is that the execution time should be low”.*

Self-evaluation, Feedback: The self-evaluation is done by asking user to select from the list of constraints, as shown below.

Self-evaluation activity- constraints

Choose one of the options below which matches closely to your response

1. Execution time for search/insert/delete/update/traverse operation is less than 1 sec
2. Implementation difficulty should be low
3. Memory allocation(static, dynamic) should be dynamic
4. Memory space needed should be less than 1 MB
5. Others

6) Evaluate and select

Prompt, Hint: The prompt posed is*, “Evaluate alternative solutions and select optimal solution using decision matrix below.”* The hint is *“Decision matrix is used to select the solution that satisfies the goals/sub-goals and constraints.”*

Activity: The decision matrix is preloaded with solutions in the first column, and constraints in first row as shown in fig. 7. The learner has to evaluate solutions against sub-goals, constraints, and accordingly rank and justify.

Solutions	**The librarian will enter the details of all the available books**	**Teachers and students will be able to search books by title, author, isbn etc.**	**search time should be less than 1 second**	**Rank**
Book details will be stored in array and operations are linear search, insert new record at end	No	Yes	No	4
Book details will be stored in linked list and operation are linear search	Yes	Yes	No	3
Book details will be stored in binary search tree	Yes	Yes	Yes	1
Book details will be stored in AVL tree	Yes	Yes	Yes	2

Fig. 7. Decision table to evaluate solutions and rank

VI. Study

The study was conducted to investigate the effectiveness of Fathom. The research question investigated is “How effective is Fathom in teaching and learning of the expansionist and reductionist thinking skills?”

A. Participants

Total 47 students from second year computer engineering participated in the study. The study was conducted at the end of the Data Structure course, thus ensuring that the students had enough domain knowledge to solve design problems in Data Structures.

B. Experiment Design

The experiment methodology is pre-test– intervention-post-test as shown in fig. 8.

C. Experiment procedure

The pre-test was taken before the intervention. The students were given a worksheet to solve a shop inventory

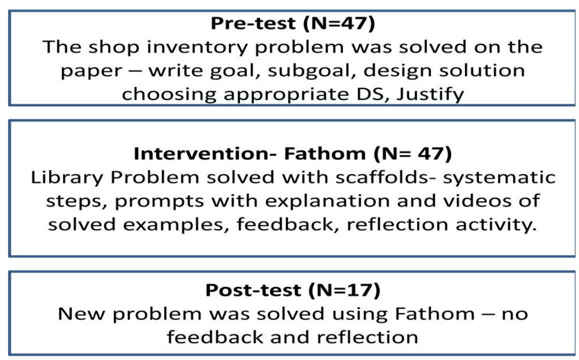

Fig. 8. Experiment design

problem: “Design a software system for supermarket to display items below threshold.” The problem solving steps were given as listed below.

1. Write the broad goal to be achieved.
2. Write the sub-goals to be achieved in terms of data to be stored and operations(insert/delete/search) to be performed.
3. Design solution using appropriate data structure and algorithms.
4. Justify why the selected data structure is appropriate for the given problem.

Immediately after pre-test, the learners interacted with Fathom for 2 hr (http://www.et.iitb.ac.in/~deepti/Fathom/). The training problem was a Library problem, which was solved in Fathom with the help of scaffolds: problem solving steps, prompts with explanation and videos of solved examples, feedback, and reflection activity provided with fathom.

For the post-test, students were informed to select a new problem in Fathom to solve. The scaffolds that were faded included feedback and the reflection activity. The scaffolds retained were the problem solving steps, prompts, hints and demo videos. Only 17 students responded for the post-test.

D. Data Collection instrument

The data collection instrument for the experiment were as listed below

- Pre-test scores for each step (Shop inventory)
- Activity scores for each step (Library)
- Self-evaluated score
- Log data
- Post-test score for each step (new problem)
- Student perception rating

VII. Results of the experiment

The responses of the pre-test problem, training problem and post-test problem were evaluated using a rubric on a scale of 1- low , 2- medium, 3- high performance(available at https://www.scribd.com/document/385091077/Rubric-for-Rating-the-ER-Skills). Following statistical tests were performed to answer the research question (RQ).

T-test between pre-test and training scores (table 1).

- T-test of pre-test and post-test scores (table 2).
- Percentage of rating of student perception survey on usefulness of activity in learning is shown in fig 9.

TABLE I. Pre-test, Activity scores

N=47	Understand the problem	Formulate the problem	Generate solutions	Identify criteria and constraints	Evaluate solutions	Select solution and Justify
Pretest	-	1.75	1.5	-	-	0.90
Activity	2.55	2.29	2.16	2.31	1.84	1.92
T-test(p value)	-	0.01	0.02	-	-	0.00

TABLE II. Pre-test, post-test scores

N=17	Understand the problem	Formulate the problem	Generate solutions	Identify criteria and constraints	Evaluate solutions	Select solution and Justify
Pre-test	-	1.5	1.2	-	-	0.7
Post-test	2.41	2.4	2.11	2.4	2.4	2.3
T-test(p value)	-	0.00	0.00	-	-	0.00

VIII. Discussion

The RQ, "How effective is Fathom in teaching and learning of the expansionist-reductionist thinking skills?" is answered based on the comparison of pre-test, activity and post-test scores, and student perception survey. The activity and post-test scores show that students were able to exhibit ER skills in understanding the problem from the perspectives of all stakeholders, and identifying sub-goals. They were also able to generate multiple solutions and identify criteria to select the solution and justify. The results showed significant improvement in the quality of problem formulation, solution design and justification during activity and post-test. The students were able to transfer the skills to a new problem in the post-test with no explicit feedback given by the system. The test scores are triangulated with student perception survey rating. More than 80% of the students agreed or strongly agreed that the activities in Fathom helped them to perform ER thinking.

During the pre-test, the students lacked the ability to understand the problem from the perspectives of multiple stakeholders, which led to weak formulation of sub-goals. For example, for the shop problem posed "Design a software system for supermarket to display items below threshold", most of the students identified the operations only from the perspective of the shopkeeper, for example, one of the responses given was "*store and display the shop items"*. They lacked the ability to understand the system from the perspective of customers, vendors, etc. They were not able to visualize the whole system. In Fathom, the activity of drawing the system model from the perspectives of various stakeholders helped in visualizing the system as a whole, and in effective identification of the sub-goals.

During the pre-test, the students lacked the ability to generate potential alternative solutions, which led to selection of suboptimal or wrong solution. For example, for the shop problem, some of the students selected wrong data structure such as priority queue/stack/queue, while some selected array or linked list (which could be one of possible solution) but were not able to justify. The selection of wrong solution may be due to lack of understanding the problem or selecting the data structure which they have studied recently or could recall easily. In Fathom, the activity of listing the possible data structures and algorithms for each data item and operations, respectively helped students to generate alternative solutions. Later, the process of identifying selection criteria and constraints helped to systematically select the optimal solution for the given problem.

The prompts helped in expanding and reducing the thinking at appropriate points in problem solving. The example illustrated the process of applying expansionist-reductionist thinking for the shop problem which students had solved in the pre-test. The process of doing the ER activities helped in practicing the ER skills, and the adaptive feedback helped the students to improve on their skills. The feedback was addressing the level of expertise achieved, and gave positive and corrective feedback accordingly. Overall, the process of demonstrating the skills with an example, doing the activity and feedback helped in reflecting on their learning process, and improving the ER skills.

Based on the log data analysis, it was found that very few students modified their responses on the basis of the feedback. The process of self-evaluation was not found to be effective as most of the students either over-rated or under-rated their performance, and thus need to be further investigated.

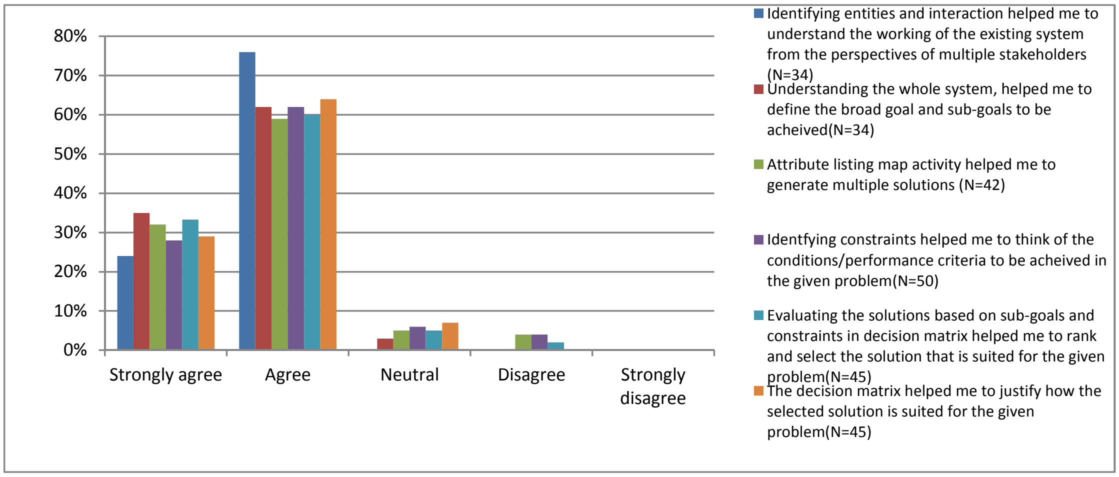

Fig 9. Student perception survey rating

IX. Conclusion

The expansionist-reductionist(ER) thinking skills are important in solving software design problems. The ER skills help in effectively formulating the problem and designing the solution by understanding the problem from multiple perspectives, and generating alternative solutions. The TEL system Fathom is designed and developed for teaching and learning of ER skills in the context of solving software design problems. The features of the TEL include structured problem solving, ER activity area, demo videos, feedback and reflection activity. Fathom was tested and evaluated by conducting a research study with second year computer engineering students. The results showed significant improvement in learning the ER skills for students.

References

[1] Ackoff, Russell L. "The future of operational research is past." *Journal of the operational research society* (1979): 93-104.

[2] Basadur, Min, Mitsuru Wakabayashi, and George B. Graen. "Individual problem□solving styles and attitudes toward divergent thinking before and after training" Creativity Research Journal 3.1 (1990): 22-32.

[3] Bannert, Maria, and Christoph Mengelkamp. "Scaffolding hypermedia learning through metacognitive prompts." International handbook of metacognition and learning technologies. Springer New York, 2013. 171-186.

[4] Reddy, P. D., Iyer, S., & Sasikumar, M. (2017, July). FATHOM: TEL environment to develop divergent and convergent thinking skills in software design. In Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on (pp. 414-418). IEEE.

[5] Eden C, Ackermann F. SODA—the principles. In: Rosenhead J, Mingers J, editors. Rational analysis for a problematic world revisited. 2nd ed, Chichester: Wiley Ltd.; 2001. p. 21–42.

[6] Ellspermann, Susan J., Gerald W. Evans, and Min Basadur. "The impact of training on the formulation of ill-structured problems." *Omega* 35.2 (2007): 221-236.

[7] Howard, Thomas J., Stephen J. Culley, and Elies Dekoninck. "Describing the creative design process by the integration of engineering design and cognitive psychology literature." Design studies 29.2 (2008): 160-180.

[8] Jonassen, David. "Supporting problem solving in PBL." Interdisciplinary Journal of Problem-based Learning 5.2 (2011): 8.

[9] Kunene, K. N. (2005). A Field Experiment on the Effect of Task Decomposition on the Quality of Decisions in a Group Support Environment. *Int. J. Comput. Syst. Signal*, *6*(2), 13-25.

[10] Madhuri Mavinkurve, " Development and assessment of engineering design competencies" (Doctoral thesis), 2015.

[11] Narciss, S. (2013). Designing and Evaluating Tutoring Feedback Strategies for digital learning environments on the basis of the Interactive Tutoring Feedback Model. *Digital Education Review*, (23), 7-26.

[12] Norese MF. Macrame: a problem formulation and model structuring assistant in multiactorial contexts. European Journal of Operational Research 1995;84(1):25–34.

[13] Pugh, S, "Total design e integrated methods for successful product engineering",Addison-Wesley Publishers Ltd, Strathclyde, 1991.

[14] Runco, Mark A., and Selcuk Acar. "Divergent thinking as an indicator of creative potential." Creativity Research Journal 24.1 (2012): 66-75.

[15] Sahana Murthy, Sridhar Iyer, and Madhuri Mavinkurve., "Pedagogical Framework for Developing Thinking Skills Using Smart Learning Environments In 'Learning, Design, and Technology", An International Compendium of Theory, Research, Practice, and Policy'; Michael J Spector, Barbara B Lockee and Marcus D. Childress (eds). Springer International Publishing Switzerland, 2016.

[16] Saaty, Thomas L. "Decision making with the analytic hierarchy process." International journal of services sciences 1.1 (2008): 83-98.

[17] Schon, D. A. (1987). Educating the Reflective Practitioner. Toward a New Design for Teaching and Learning in the Professions. The Jossey-Bass Higher Education Series. Jossey-Bass Publishers, 350 Sansome Street, San Francisco, CA 94104.

[18] S. Pugh (1981) Concept selection: a method that works. In: Hubka, V. (ed.), Review of design methodology. Proceedings international conference on engineering design, March 1981, Rome. Zürich: Heurista, 1981, blz. 497 – 506.

[19] Volkema, Roger J. "Problem formulation in planning and design." *Management science* 29.6 (1983): 639-652.

[20] Xun Ge., "Designing Learning Technologies to Support Self-Regulation During Ill-Structured Problem-Solving Processes." International Handbook of Metacognition and Learning Technologies, Springer New York, (2013). 213-228.