Multi-dimensional Goal Refinement in Goal-Oriented Requirements Engineering

Wataru Inoue[1], Shinpei Hayashi[1], Haruhiko Kaiya[2] and Motoshi Saeki[1]

[1]*Department of Computer Science, Tokyo Institute of Technology, Ookayama 2–12–1, Meguro-ku, 152–8552, Tokyo, Japan*

[2]*Department of Information Sciences, Kanagawa University, Tsuchiya 2946, Hiratsuka, 259–1293, Kanagawa, Japan*

Keywords: Goal-Oriented Requirements Engineering, Concern, Multi-Dimensional Space.

Abstract: In this paper, we propose a multi-dimensional extension of goal graphs in goal-oriented requirements engineering in order to support the understanding the relations between goals, i.e., goal refinements. Goals specify multiple concerns such as functions, strategies, and non-functions, and they are refined into sub goals from mixed views of these concerns. This intermixture of concerns in goals makes it difficult for a requirements analyst to understand and maintain goal graphs. In our approach, a goal graph is put in a multi-dimensional space, a concern corresponds to a coordinate axis in this space, and goals are refined into sub goals referring to the coordinates. Thus, the meaning of a goal refinement is explicitly provided by means of the coordinates used for the refinement. By tracing and focusing on the coordinates of goals, requirements analysts can understand goal refinements and modify unsuitable ones. We have developed a supporting tool and made an exploratory experiment to evaluate the usefulness of our approach.

1 INTRODUCTION

In information systems development, it is necessary to clarify causality relationships between organizational goals and requirements to information systems, e.g., how and which organizational goals requirement to an information system resulted from. These causality relationships are very significant to verify if the information systems can satisfy real customers' needs. For example, Gulla analyzed system development projects whose delivery dates were delayed (Gulla, 2004) and found that they were re-enacted again. In these projects, the requirements related to the achievement of customers' strategy were not sufficiently analyzed, and as a result, the customers' needs were not satisfied. Thus, we need a technique to keep traceability links between various types of goals such as strategic goals and functional goals during a requirements elicitation phase.

Goal-oriented requirements engineering (GORE) is one of the promising approaches, and in this approach, elicited requirements and their relationships are represented with a graph, called goal graph (van Lamsweerde, 2009). We can reason about how a goal is derived by means of tracing the edges incoming to and outgoing from it. However, as elicited requirements are increasing more and more in a goal graph, it is more difficult to analyze them using the graph because the numbers of the goals and their relationships are larger. As a result, the structure of the goal graph is more complicated. A case study of analyzing a large-scale system was reported where goals exceeded more than 500 in a goal graph (van Lamsweerde, 2009). The complicated graph results in the difficulties of analyzing it, especially of finding specific goals and of tracing their relationships to the other goals such as their parent and/or sub goals. As a result, maintaining the goal graph can be error-prone tasks for human analysts. In addition to the larger number of goals and edges, goal refinements cause the difficulties in maintaining the goal graph.

A (parent) goal is refined into the sub goals that contribute to its achievement. More precisely, the achievement of the parent goal is a logical entailment of its sub goals' achievement from logical view (Mylopoulos et al., 1992; Giorgini et al., 2002; van Lamsweerde, 2001), i.e., if the achievement of the sub goals holds, that of the parent goal also holds. This rule on the logical entailment of goal achievement dominates the formal meaning of goal refinements. Although some techniques propose the patterns of goal refinements such as Divide-and-Conquer and Guard-Introduction (van Lamsweerde, 2009), most of them are based on logical entailment. For require-

Inoue W., Hayashi S., Kaiya H. and Saeki M..
Multi-dimensional Goal Refinement in Goal-Oriented Requirements Engineering.
DOI: 10.5220/0005499301850195
In *Proceedings of the 10th International Conference on Software Engineering and Applications* (ICSOFT-EA-2015), pages 185-195
ISBN: 978-989-758-114-4

ments analysts as practitioners, they refine goals into sub goals from various views not only from the view of logical entailment. In some cases, analysts may refine a goal into actions or tasks that can realize its achievement, and in other cases they may do the goal into the conditions or constraints that should hold for achieving it. The former case is based on operational view of goal refinements, i.e., a sequence of actions or tasks is necessary to be executed for the achievement of the goal, and the latter is on a logical view of the conditions or assumptions of the goal. The wide variety of goal refinement views causes the difficulties in analysts' constructing and/or maintaining a goal graph. To use GORE practically, it is necessary for analysts to understand goal refinements well (Munro et al., 2011).

The paper proposes a technique to classify goal refinement views and show them for analysts so as to understand the goal graph easily. The basic idea of our approach is the usage of the concept of a multi-dimensional space. In our approach, a goal graph is put in a multi-dimensional space, a concern of requirements analysts corresponds to a coordinate axis in this space, and goals are refined into sub goals referring to the coordinates. Thus, the meaning of goal refinements is explicitly provided by means of the coordinates used for the refinements. By tracing and focusing on the coordinates of goals, requirements analysts can understand goal refinements and furthermore correct unsuitable ones.

The rest of the paper is organized as follows. In Section 2, we show a motivating example and clarify where our problem to be solved is. Section 3 presents our approach and in Section 4, we show the supporting tool based on our approach. We assess our approach with an experiment and discuss its effectiveness based on the experimental results in Section 5. Sections 6 and 7 are for related work and concluding remarks.

2 MOTIVATING EXAMPLE

In GORE, the customers' and users' abstract needs can be considered as goals to be achieved and the goals are decomposed or refined into more concrete sub goals. The achievement of the sub goals contributes to the achievement of their parent goals. We have two types of goal refinement; one is AND refinement, and the other is OR. In AND refinement, if all of the sub goals are achieved, their parent goal can be achieved, while in OR refinement, the achievement of at least one sub goal leads to the achievement of its parent goal. The meaning of refinement is "logical entailment" only.

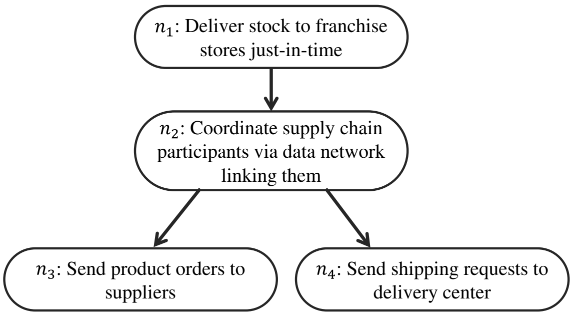


Figure 1: Part of a goal graph in B-SCP: Excerpt from (Bleistein et al., 2006).

Figure 1 illustrates a part of the goal graph excerpted from (Bleistein et al., 2006), which was the result of analyzing the business strategy of Seven Eleven Japan (Makino and Suzuki, 1997). In the figure, rounded boxes and arrows stand for goals and goal refinements, respectively. For example, the goal n_2 is refined into sub goals n_3 and n_4 in OR refinement. Considering the meaning of these goals, their goal refinement is not easy to understand underlying ideas on how these sub goals n_2, n_3, and n_4 had been derived from the root goal n_1. The goal n_1 specifies stock delivery *just-in-time*, and it includes both a functional aspect (stock delivery) and a strategic one (just-in-time)[1]. Its direct sub goal n_2 specifying the coordination of supply chain contributes to the achievement of the strategic aspect of n_1, just-in-timeliness of stock delivery, but less contributes to its functional aspect, i.e., stock delivery. Rather, n_3 and n_4, which are direct sub goals of n_2 not n_1, seem to directly contribute to the achievement to the stock delivery of n_1. It can be considered that the refinement of n_1 to n_2 is done based on a strategic aspect, and on the other hand, the refinement of n_2 to n_3 and n_4 is on functional aspect of n_1, not of n_2. Thus, n_3 and n_4 are not so suitable as the result of refining directly the goal n_2, rather it is better to consider that the goal n_1 is directly refined into them following a functional viewpoint. The point is that this example includes mixed views of a goal refinement, strategic view, and functional view in a plane goal graph. More concretely, goals derived from multiple views are intermixed in the same graph, and this intermixture leads the difficulty of understanding and furthermore maintaining the goal graph.

We call the aspects that a goal specifies and that a requirements analyst has an interest for his/her analysis *concerns*, and the analyst refines the goal into its sub goals from the view of these concerns. The goal

[1]Here, we regard the constraints of goals as strategic.

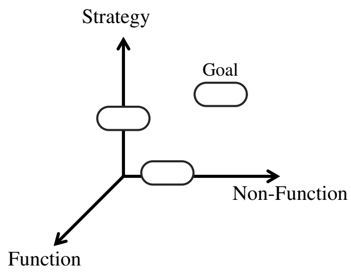


Figure 2: Goals in a multi-dimensional space.

refinements based on these concerns are more natural for human analysts rather than based on the view of logical entailment. A goal specifies more than one concern as shown in the goal n_1, and a requirements analysts try to refine it into sub goals from the view of its concerns. Suppose that the analyst refines the goal n_1 directly into n_3 and n_4 from the view of a functional concern and into n_2 from the view of a strategic concern. The resulting goal graph includes two types of goal refinement, and as a result, it is difficult to understand the meaning of the relationships between the parent goal n_1 and its sub goals (n_2, n_3, and n_4) because of the intermixture of goal refinement having different meaning. This paper addresses this problem, and the next section presents how to tackle it.

3 OUR APPROACH

3.1 Basic Idea

The problem that we address in this paper is the intermixture of multiple goal refinements based on the concerns related to information systems development. We adopt the idea of multi-dimensional spaces for addressing this problem. In this space, we can consider that a goal is put as a point, and a dimensional axis stands for the concern of a goal refinement. In our technique, we construct and maintain a goal graph whose goals are projected to the multi-dimensional space, and we concentrate on goal refinements for a single concern by tracing them along the corresponding coordinate axis. That is to say, each coordinate corresponds to a concern. The aim of our work is to develop the supporting technique to maintain goal graphs, especially to understand the relationships among goals through the meaning of goal refinements. Figure 2 illustrates the overview of this idea. This is a three-dimensional space, and it has three coordinate axes; function, strategy, and non-function. Suppose that a goal is refined into a sub goal from the view of a strategy concern. This sub goal is located in a vertical direction of its parent goal.

3.2 Coordinate of Goals

As mentioned above, since a goal in a goal graph is a point having multiple coordinates in a multi-dimensional space, and each coordinate corresponds to a concern, the coordinate of a goal means what concern the goal specifies. As shown in Figure 3(a), the goals of the motivating example in Figure 1 can have two coordinates: a function concern and a strategy one. Thus, this goal is projected to a two-dimensional space whose axes are Function and Strategy. Since n_1 specifies these two concerns, on one hand, we set its coordinate value to be $\langle function, strategy\rangle$. On the other hand, n_2 specifies only one concern, i.e., strategy, and so its value is $\langle -, strategy\rangle$. The value "–" denotes a null value, and it means that the goal does not specify the corresponding coordinate. The coordinates are similar to the attributes or semantic tags attached to goals (Tanabe et al., 2008; Saeki et al., 2009; Hayashi et al., 2012).

Our approach has three beneficial points to support maintaining goal graphs. The first one is that we can reason about the meaning of goal refinements and can understand how sub goals are derived. For example, the refinement from n_1 into n_2 has a meaning of the refinement of the strategy "just-in-time" because both of these goals have a strategy concern. If a requirements analyst tries to focus on a strategy concern, he/she can trace the goals having a coordinate "strategy" and their refinements. As a result, he/she can understand why and how the strategic goals were derived while constructing the goal graph. We can classify the meaning of goal refinements as follows, and this clarification is useful to reason about how goals specifying a certain concern are derived.

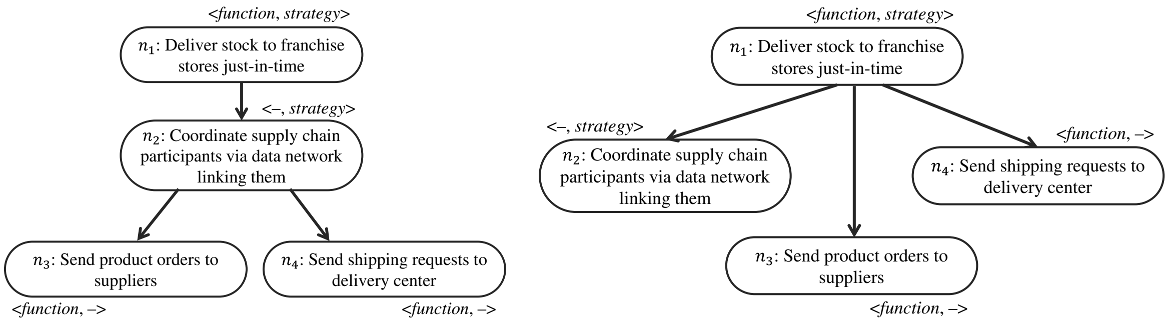
A parent: strategy, a sub goal: strategy
→ The parent goal is divided or refined into a more concrete strategy denoted by the sub goal.

A parent: strategy, a sub goal: function
→ The sub goal is a function or an operation to realize the strategy denoted by the parent goal.

A parent: function, a sub goal: strategy
→ The sub goal is a strategy to implement the function denoted by the parent goal.

A parent: function, a sub goal: function
→ The function denoted by the parent goal has the sub-function denoted by the sub goal.

In the first category of the above classification, if both a parent and a sub goal specify a strategic concern, the parent goal is divided or refined into the strategy denoted by the sub goal. By reading these meaning de-



(a) Coordinates of goals. (b) Modifying the goal refinement.

Figure 3: Coordinates of the motivating example.

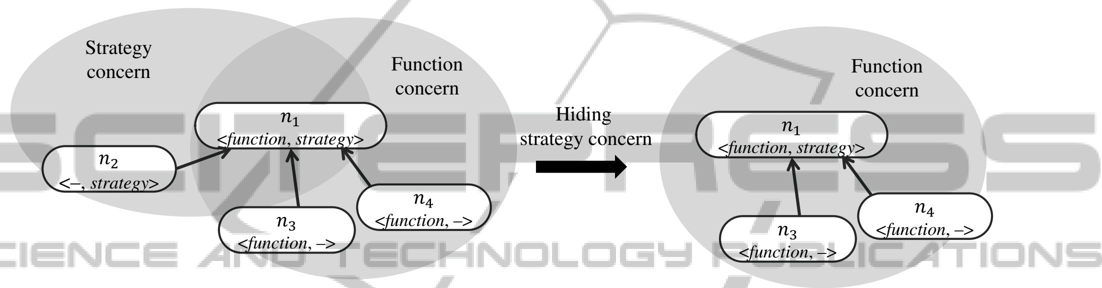


Figure 4: Focusing on a concern.

scriptions on tracing a sequence of goal refinements, a requirements analyst can reason about the derivation process of the related goals.

The second benefit is to support the detection of unsuitable goal refinements in a goal graph as shown in Figure 1. Suppose that we have the rule of goal refinement “A goal should be refined into sub goals from the view of at least one of the concerns that it specifies”. In a multi-dimensional space, this rule can be translated into “The coordinates that are a null value at a parent goal should be a null value at its sub goals”. Figure 3(a) violates this rule at the refinement from n_2 into n_3 and n_4, because the coordinate Function has a null at n_1 while its value is neither null at n_3 nor at n_4. Thus, we can obtain the modified version as shown in Figure 3(b) by letting n_3 and n_4 be direct sub goals of n_1. This approach allows us to automate detecting the occurrences of semantically unsuitable goal refinements. Note that this rule is an extreme but comprehensive example. We used it only for the explanation of the second benefit.

The third beneficial point is the mechanism to filter out the goals specifying irrelevant concerns, i.e., the goals that a requirements analyst has less interest in can be hidden. For a human analyst, it is difficult to analyze a goal graph recognizing many types of concerns at the same time, especially in the case when the graph is larger and more complicated. The analyst can select a small number of the concerns that he/she wants to focus on at first. Figure 4 illustrates hiding a strategy concern from the motivating example. As a result, the analyst can devote himself/herself to focusing on goals and their refinements from the viewpoint of a function concern.

We can summarize the beneficial points to improve the existing GORE approach by using our approach as follows:

1. understandability on why and how sub goals are derived by clarifying the meaning of goal refinements,
2. detectability of unsuitable goal refinements by considering coordinates of goals, and
3. easiness to analyze goal graphs by reducing their size by means of hiding the goals having less interesting concerns.

3.3 Applying Our Approach

In this subsection, we mention how to apply our approach to GORE. There are two alternatives to apply it. The first alternative is its application to the construction of a goal graph specifying no unsuitable goal refinements. A requirements analyst decides what concerns he/she has an interest and then starts constructing a goal graph. While constructing the graph, the analyst can find and add new concerns that he/she must consider. He/she considers what concerns that a

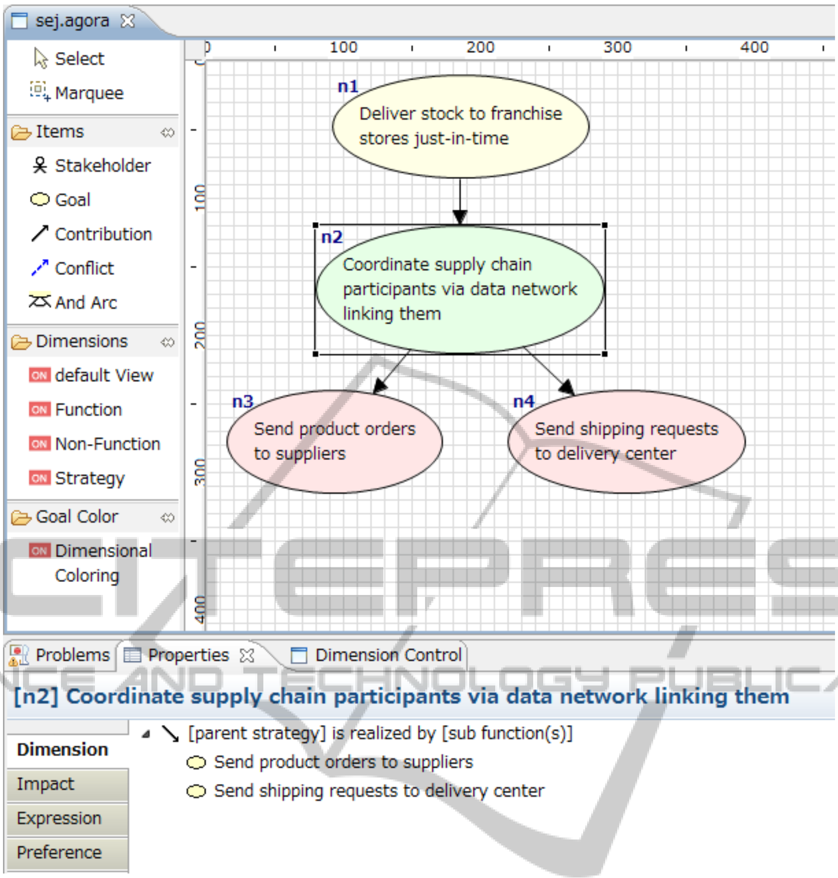


Figure 5: Screenshot of the supporting tool.

goal represents and attaches the suitable corresponding concerns to it. The attached coordinates play a role of the guideline on goal refinements. Suppose that a requirements analyst has a goal n_1 shown in Figure 3 and that he/she focuses on a function concern and a strategy one. The analyst recognizes that n_1 has both of function and strategy concerns and for each concern, the analyst tries to decompose n_1 into sub goals. As a result, as shown in Figure 3(b), the analyst refines n_1 into n_3 and n_4 from the viewpoint of a function concern, while into n_2 from a strategy view.

The second alternative is the application of our approach to already constructed goal graphs. A requirement analyst has constructed a goal graph by using the existing technique such as KAOS, the strategic rationale model in *i**, or AGORA, and then he/she identifies what concerns each goal in the graph has. After finishing the attachment of corresponding coordinates to the goals, the analyst checks the suitability of goal refinements and improves them by changing sub goals or adding new sub goals if any unsuitability exists. To understand the goal graph for maintaining it, the analyst can trace the goal graph along a goal refinement of a certain concern. In addition, the analyst can remove goals having the other concerns from his/her view. These supports result from the three beneficial points mentioned in the previous subsection.

4 SUPPORTING TOOL

We have implemented a tool for supporting our approach by using an AGORA tool that had been developed before by the authors' group (Saeki et al., 2009). Figure 5 illustrates the screenshot for editing a goal graph. The example goal graph shown in this screenshot is the same as the one shown in Figure 3(a). The following is a list of the functions of this tool.

1. **Getting a coordinate definition file:** Before starting and/or while constructing a goal graph, a requirements analyst can define new coordinates. The tool can obtain a file having information on the coordinates that the analyst (also the tool user) newly defines in XML form. The definition of a coordinate includes the name of the coordinate,

new goal refinement rules in which the coordinate participate, and semantic information on the goal refinements related to the coordinate. The refinement rules are represented in a matrix form whose entries are suitable goal refinements, and are used to detect the occurrences of unsuitable goal refinements. The semantic information is written in text and presents the explanations of the goal refinements as shown in the previous section. It is shown to an analyst and useful for him/her to understand how the corresponding sub goals have been derived.

2. **Attaching coordinates to a goal and displaying them:** An analyst selects a goal that he/she tries to attach coordinates, i.e., concerns, by clicking it with a pointing device and chooses suitable coordinates from a pop-up menu. In the screen of the tool, coordinates are differentiated by coloring a goal. For example, a function, a strategy, and a non-function coordinate are respectively assigned to red, green, and blue, and goals having these coordinates are colored following this color assignment. Like the goal n_1 having function and strategy concerns, it is colored with yellow, which is the mixture of red and green.

3. **Hiding goals:** When an analyst selects the concerns that he/she likes or does not like to focus on, the tool displays the goals specifying the selected concerns or not specifying them.

4. **Showing semantic information of a goal refinement:** See the Properties tab in the bottom screen of Figure 5. This tab is for the area where the meaning of the selected goal refinements is displayed in a textual form. More precisely, when an analyst points a goal, the tool displays the meaning of its refinement to sub goals and their contents in a tree form. In the figure, the analyst points the goal n_2, and the tool shows the meaning of the refinements to n_3 and n_4 "[parent strategy] is realized by [sub function(s)]". This means that strategy n_2 is realized by sub function(s) n_3, etc. The analyst considers if the strategy "Coordinate supply chain participants via data network linking them" (n_2) can be realized by "Send product orders to suppliers" (n_3) or not. Perhaps, he/she may be doubtful for this refinement and consider that it is not sufficient from the view of realizing a strategy by functions. In this way, considering the meaning of goal refinements, the analyst may also find an unsuitable goal refinement.

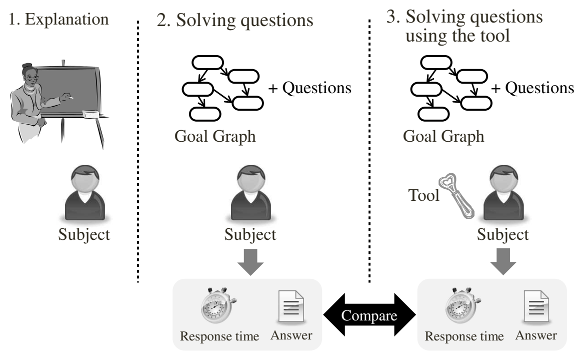


Figure 6: Experiment.

5 EXPLORATORY EXPERIMENT

As mentioned in Section 3, our approach and tool have three beneficial points:

- understandability on why and how sub goals are derived by clarifying the meaning of goal decompositions,
- detectability of unsuitable goal decompositions by checking coordinates of goals, and
- easiness to analyze goal graphs by reducing their size by means of hiding the goals having less interesting concerns.

In this section, we made an experiment to assess if these beneficial points are obtained or not. The main results of this experiment are shown in Table 1, which is explained later.

5.1 Procedure

The aim of our experiment is to confirm that our approach including the supporting tool can have the above three beneficial points. To do, we made a comparative experiment with two settings; one is the situation that our subjects analyze goal graphs using the existing technique, and the other is that they analyze the goal graphs using the supporting tool. Their processes and outcomes are compared. Figure 6 illustrates this experimental procedure. Our subjects were provided with goal graphs and questions about them, and they answered the questions by analyzing the goal graphs. The subjects were grouped into two sets; one is for the subjects who use the existing goal graph editor such as (Saeki et al., 2009), and the other is for the subjects who use our tool mentioned in Section 4. Subjects made answers to the several questions related to sample goal graphs that might include unsuitable goal decompositions. We observed their answering processes and their resulting answers and extracted the following information.

1. The numbers of wrong answers and missing answers: We checked if the answers of a subject using our approach included wrong answers and missing ones less than those of a subject not using our approach.
2. The time spent in making answers (response time): We measured the time when a subject spent in making answers. This is for checking if our subjects could reduce their efforts or not.

We made questions to observe and evaluate the following items;

Item (a) our subjects can understand goal decompositions,

Item (b) our subjects can detect unsuitable goal decompositions, and

Item (c) our subjects can focus on goals having a certain concern.

We designed three types of questions to obtain the information on the above items from our subjects as follows.

- **Question Type 1 (QT 1): Selecting goals having the specific roles on deriving the given goals.** Given two goals that are connected in a goal graph, a subject is required to identify the goals having specific roles on deriving these given two goals. Since the given two goals, letting a descendant goal and its ancestor be n_1 and n_2 respectively, are connected, one goal n_1 is transitively derived from the other goal n_2. There are goals on the path between n_1 and n_2 that more contribute to the derivation of n_1 from n_2 based on a certain concern. The subjects were asked to look for such goals as these. To find them correctly, the subjects should understand the derivation process from n_2 to n_1, i.e., how and why n_1 is derived from n_2 from a view of a certain concern. The example of the question of this type is "Goal n_1 can contribute to the achievement Goal n_2. Which can goals on the path between n_1 and n_2 play a role of Function for their achievement?".
- **Question Type 2 (QT 2): Identify goals decompositions to be modified.** We gave goal graphs of lower quality that had been artificially generated and asked our subjects to find the goal decompositions to be modified. To generate the goal graphs of lower quality, we picked up original goal graphs whose quality was high, and deteriorate them by reconnecting some of sub goals to wrong parent goals so that we got goal graphs having unsuitable goal refinements. See Figure 7. Given the produced unsuitable goal graphs, a subject is required to identify unsuitable goal refinements.

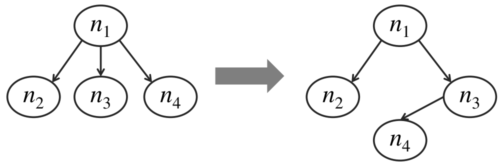


Figure 7: Modifying edges in a goal.

- **Question Type 3 (QT 3): Identifying insufficient derivation of sub goals from a certain concern.** Given a certain coordinate and a goal having it, a subject is required to check if there exist the goals that contribute to the given goal from the view of the other relevant coordinates, or not. The example of the question of this type is "Where are the goals having the concern Non-Function whose their achievement as Function are not considered, if any?" In a completed goal graph, the goals representing non-functional requirements should be finally refined into sub goals that denote the functions to implement these non-functional requirements. If these sub goals do not exist yet in a goal graph, the goals of the non-functional requirements should be analyzed and refined further. This type of questions is for the subjects to select the goals necessary to be analyzed and refined further.

These types of questions correspond to the observation Items (a), (b), and (c) mentioned above. For example, to answer QT 1 correctly, a subject should understand the meaning of goal decompositions between the given two goals, e.g., the goal n_1 and n_2 in the example question. We prepared 3, 1, and 2 questions for QT 1, 2, and 3, respectively. Some of the questions correspond to multiple solutions. For example, the question of QT 2 for each graph correspond to 8 solution goals.

In our experiments, we use three coordinates: function, non-function, and strategy. Note that we did not use goal decomposition rules, i.e., any coordinate combination was allowable in this example, and we made the textual explanations of the meaning of 9 (= 3 coordinates $\times$ 3) types of goal decompositions. The textual explanations of the meaning of decomposing a strategy concern into a function concern are displayed to our subjects on the screen of the tool, as shown in Section 4. The reason we did not use the rules is as follows. It is obvious that our tool can detect the violation of the rules in an instant, and the comparison of correct answers and answering time is not so meaningful. Rather, we would like to show our semantic information of goal decompositions can contribute to detecting unsuitable goal decompositions as well.

Table 1: Average of correct and wrong answers.

		G_{SEJ}			G_{CRE}		
		Correct	Wrong	Missing	Correct	Wrong	Missing
QT 1	No support	2.5	2.5	0.5	1	3.5	2
	Our approach	2.5	1	0.5	3	4.5	0
QT 2	No support	3	0	5	3	0	5
	Our approach	3.5	0	4.5	3.5	0	4.5
QT 3	No support	2	1	3	1	2	2
	Our approach	5	1	0	3	3	0

Table 2: Details of used goal graphs.

	G_{SEJ}	G_{CRE}
# goals	50	50
# functions	26	22
# non-functions	11	15
# strategies	15	16
# edges	61	60

Table 3: Goal graphs that subjects used.

	No support	Our Approach
Subject *A*	G_{CRE}	G_{SEJ}
Subject *B*	G_{CRE}	G_{SEJ}
Subject *C*	G_{SEJ}	G_{CRE}
Subject *D*	G_{SEJ}	G_{CRE}

We used two goal graphs G_{SEJ} and G_{CRE}. G_{SEJ} is for the strategies of Seven Eleven in Japan (Makino and Suzuki, 1997), and G_{CRE} is for an information system where university students apply for class credits. Table 2 shows their size such as the numbers of goals, edges and the occurrences of coordinates. For example, the graph G_{SEJ} had 50 goals and 61 edges, and function coordinate was attached to 26 goals out of the 50 ones.

We had four student subjects of the computer science department who had learned requirements engineering. All of them had experiences in developing information systems, but were not expertized to requirements analysis techniques. Thus, as shown in Figure 6, we gave the subjects a short lecture of GORE and the existing goal graph editor, which had no support of our approach and our tool in 20 min. They tried out these tools in this lecture. Table 3 shows the assignment of the goal graphs G_{SEJ} and G_{CRE} to the subjects. For example, Subject *A* analyzed G_{CRE} with the existing goal graph editor and G_{SEJ} with our tool, i.e., he/she proceeded the step 2 with G_{CRE} and the step 3 with G_{SEJ}. We use the term "No support" to express the group of the subjects who did not use our tool but the existing goal graph editor.

Table 4: Average time spent in answering.

		G_{SEJ}	G_{CRE}
QT 1	No support	2'48"	1'44"
	Our approach	2'39"	2'16"
QT 2	No support	10'15"	10'34"
	Our approach	10'47"	10'44"
QT 3	No support	3'14"	4'44"
	Our approach	3'31"	3'59"

5.2 Results

Table 1 shows the results of the subjects' answers. The numerals of the table represent the average numbers of correct answers, wrong answers, and missing answers over the four subjects. In the case where a subject did not find a correct answer, we count it as a missing answer. Suppose that a goal n is one of the correct answers. If a subject answered as n, we count this answer as a correct one in the table. On the other hand, if he/she did not, we count it as a missing answer, but if he/she did the different goal n', it is a wrong answer. In the example of QT 1 and the goal G_{SEJ}, each subject answered three questions, and two subjects of four used our approach. The total number of *real* correct answers to these questions was just 3. Subject *A* answered the three questions correctly, i.e., he/she could find all three goals that were correct answers to the questions. However, Subject *B* listed four goals as his/her answers, and two of the four were correct. He/she identified two wrong goals and missed one *real* correct goal. For the two subjects using our approach, 2.5 in average were correct one, and they selected one wrong answer and missed 0.5 correct answers in average.

Table 4 shows the average time spent on subjects' answering the questions. In the example of the graph G_{SEJ} and QT 1, the average time of the subjects was 2 min 48 sec.

5.3 Discussion

According to the experimental results, we can consider that our approach allows to prevent requirements analysts from missing the goals necessary to understand goal decompositions from the viewpoint of a certain concern. In QT 1 to the graph G_{CRE} of Table 1, our approach was successful in reducing missing answers at two and increasing correct answers. However, for G_{SEJ} we could not observe the same result, so we need more experiments and investigations further. In addition, we could not observe considerable differences on wrong answers between no support and our approach, so we cannot conclude that our approach reduced misunderstanding of goal decompositions.

As for detectability of unsuitable goal decompositions, we could not obtain positive results. consider that our approach might be useful to reduce the efforts of our subjects because the time spent in finding the unsuitable goal decompositions was decreasing. See the column QT 2 of Table 4. Our subjects using the tool could reduce their average times of G_{SEJ} and G_{CRE} at only 32 and 10 sec respectively, and we could not conclude that the difference is enough large. Also, in QT 1 and QT 3, we could not obtain the evidence of the benefits of our approach. Rather, in these cases the time spent in making answers was increasing. The manipulation time of our tool, e.g., the manipulation time for retrieving the goals having specific coordinates, might cause additional time. As a goal graph to be analyzed is larger and more complicated, we might obtain clearer results.

Let us turn back to Table 1 and see the column of QT 3. For the subjects using our tool, the number of missing correct answers was reduced at 3 in G_{SEJ} and 2 in G_{CRE} respectively, and the correct answers were increasing. Thus, we can conclude that our approach reduced missing the goals necessary to be analyzed.

To sum up, our approach has possibilities of preventing analysts from missing goal decompositions to be recognized and goals to be analyzed. Regarding the reduction of the efforts in detecting unsuitable goal decompositions in a goal graph, the new usages of the tool might be the obstacle to detecting such decompositions quickly.

5.4 Threats to Validity

5.4.1 External Validity

External validity is related to generalization of our experimental results. We used students as subjects of the experiment, and in this sense, it may be doubtful whether the experimental results can hold in practitioners. However, our subjects had been involved in practical projects and had experiences in system development. Although they were not practitioners of requirements analysis or GORE, they took lectures related to requirements engineering. Thus, although experiments for practitioners may be necessary, we do not consider that experimental results extremely different. In addition, the two goal graphs and the concerns that we used are limited. The domain of the goal graphs was business application. We need to make experiments using goal graphs in different problem domains and different concerns. In particular, we consider concerns more than three: Function, Strategy, and Non-Function. Interrelated concerns and the larger number of them may result in more positive effects of our approach.

5.4.2 Internal Validity

It may be a problem that the questions that we made really reflect on the items to be evaluated. For example, we made a question like "Where are the goals having the concern Non-Function whose their achievement as Function are not considered?" to check if the subjects could identify insufficient goal decompositions or not. Suppose that we have a goal having a Non-Function concern and that it should be achieved. The Non-Function concern should be implemented by functions in an information system, so the goal graph should include some goals having a Function concern that can contribute to the Non-Function goal. If this kind of a Function goal does not exist, the goal decomposition can be insufficient, and we should add sub goals having Function concern to achieve the Non-Function goal. To identify these insufficient decompositions, the subjects should understand the meaning of goal decompositions in the goal graph well. Thus, the question can contribute to evaluating Item (c) (our subjects can understand goal decompositions) mentioned in Section 5.1. In this way, we designed the questions carefully, but we do not consider that the used questions can cover all the aspects of the items to be evaluated. Thus, we should consider wide varieties of questions that cover more aspects of the evaluation items in further experiments.

5.4.3 Construct Validity

Our experiment was based on a comparative way, i.e., we have two experiments where only one factor is different from them. This factor is the usage of the supporting tool for our approach, and the other factors such as the characteristics of subjects should be the same in the two experiments. Construct validity guarantees that nothing but the difference of this factor causes the experimental results. We chose the subjects who had the same experiences and skills as prudently as possible. We used two tools having editing functions of goal graphs. One is the AGORA editor (Saeki et al., 2009) as the tool having no functions of our approach. The other is the tool supporting our approach and is an extended version of AGORA editor. That is to say, two tools have the same user interface and the functions except for the functions related to coordinates. Thus, we do not consider there are any effects of the experimental results caused by the differences on the tools.

6 RELATED WORK

The idea of modeling intermixture of concerns with a multi-dimensional space is not new. Tarr et al. proposed a technique to manage software documents by separating them from the multiple viewpoints based on concerns (Tarr et al., 1999). The hierarchical structure of a document along a concern corresponds to a coordinate axis of the space. Our approach is the extension of this basic idea to goal graphs. Moreira et al. developed a technique of handling multiple concerns to analyze cross-cutting functional requirements (Moreira et al., 2005). In their approach, concerns are separated into coordinates in a multi-dimensional space and the relationships among the concerns are defined. These relationships are helpful to understand cross-cutting functional requirements. Their aim and concrete approach are similar to ours. However, it applied to XML documents, not goal graphs. Rather, they tried to support the selection of software architectures to implement the functional requirements. Giorgini et al. introduced the concept of *dimension* into Tropos method in order to model data warehouse systems (Giorgini et al., 2005). In their method, a dimension is attached to a goal, and it looks like a check item that is necessary to confirm the achievement of the goal. Thus, the definition and usage of a dimension concept are different from ours. Rather, they set up two perspectives for modeling: organizational perspective and decisional one, and these perspectives are conceptually closer to the idea of our coordinates. *i** approach adopts different notations of graph nodes, e.g., Hard Goal, Soft Goal, Task, and Resource (Yu, 1997). It may be useful for requirements analysts to understand the meaning of goal refinements, e.g., they can understand that the refinement from a soft goal to tasks via Means-End links expresses the implementation of the soft goal. However, in this approach varieties of notation are limited, and the goals derived from multiple concerns cannot be handled. In addition, it is difficult for human analysts to understand at a glance the underlying various intents of goal refinement. In *i** and NFR, it is also possible to tag goals, e.g., via contribution links to soft goals. However, our approach can use varieties of semantic tags as coordinates, and furthermore requirements analysts can individually define them.

Similarity to the proposed approach, our previous work (Tanabe et al., 2008; Hayashi et al., 2012) can handle semantic information to goals. Although this approach aims to support the change impact analysis of a goal graph, the proposed approach aims to obtain the goal refinement of higher quality.

7 CONCLUSION

This paper addresses the problem of the intermixture of goals specifying various concerns in a goal graph so as to make it difficult to understand the goal graph, especially goal refinements. Our approach is to adopt the idea of a multi-dimensional space, and we consider a concern as a coordinate. Goals are refined along the directions of coordinate axes. We provide the meaning of a goal refinement based on a coordinate axis, i.e., the combinations of the coordinates of a parent goal and those of a sub goal. Furthermore, we developed a supporting tool, where a requirements analyst can define his/her interesting concerns as coordinates. The tool also has functions for displaying the meaning of goal refinements, of retrieving goals having a specific concern, of hiding the goals having irrelevant concerns, etc. These functions are embedded to the existing goal graph editor AGORA as a plug-in. Although our experimental results did not show positive observations for all beneficial points that we expected, some of them were observed.

Our future work can be listed up as follows:

More Case Studies. As future work, we have to make more experiments using various problem domains, more concerns, and practitioners.

Formalism. The proposed multi-dimensional space is defined in an informal way in this paper. We can clearly define the target models by providing the extended meta model of our goal graph.

Improving the Usability of the Tool. When the number of coordinates would increase, the usability of the technique might decrease because the effects of hiding and showing goals become large. A further technique to visualize and analyze a goal graph on *many*-dimensional space should be considered in the future.

Using Different Types of Concerns. Using new types of concerns can be used in our multi-dimensional technique. For example, some quality attributes of goals, such as performance or security (Tanabe et al., 2008), can also be regarded as the concerns put on the proposed multi-dimensional space. Moreover, we are also able to regard the view from a stakeholder as a specific concern.

Handling the Type of Refinements. Although we adopted the three dimensions in this paper as an example, our approach can handle with various types of goal refinements such as causality relationship. Its reason is that an axis denotes an atomic meaning of goal relationship on refinement, and requirements analysts can define the axes that they will use. Furthermore, by combining axes, we can express complex meaning of goal relationships. It is also one of the future work to illustrate this issues and to show an additional effectiveness of our approach.

ACKNOWLEDGEMENTS

This work was partly supported by JSPS Grants-in-Aid for Scientific Research (#15K00088 and #15K00109).

REFERENCES

Bleistein, S. J., Cox, K., Verner, J., and Phalp, K. T. (2006). B-SCP: A requirements analysis framework for validating strategic alignment of organizational it based on strategy, context, and process. *Information and Software Technology*, 48(9):846–868.

Giorgini, P., Mylopoulos, J., Nicchiarelli, E., and Sebastiani, R. (2002). Reasoning with goal models. In *Proc. 21st International Conference on Conceptual Modeling* (*ER'02*), volume 2503 of *LNCS*, pages 167–181.

Giorgini, P., Rizzi, S., and Garzetti, M. (2005). Goal-oriented requirements analysis for data warehouse design. In *Proc. 8th ACM International Workshop on Data Warehousing and OLAP* (*DOLAP'05*), pages 47–56.

Gulla, J. A. (2004). Understanding requirements in enterprise systems projects. In *Proc. 12th IEEE International Conference on Requirements Engineering* (*RE'04*), pages 176–185.

Hayashi, S., Tanabe, D., Kaiya, H., and Saeki, M. (2012). Impact analysis on an attributed goal graph. *IEICE Transactions on Information and Systems*, E95-D(4):1012–1020.

Makino, N. and Suzuki, T. (1997). Convenience stores and the information revolution. *Japan Echo*, 24(1):44–49.

Moreira, A., Rashid, A., and Araujo, J. (2005). Multi-dimensional separation of concerns in requirements engineering. In *Proc. 13th IEEE International Conference on Requirements Engineering* (*ICSE'05*), pages 285–296.

Munro, S., Liaskos, S., and Aranda, J. (2011). The mysteries of goal decomposition. In *Proc. 5th International i* Workshop* (*iStar'11*), pages 49–54.

Mylopoulos, J., Chung, L., and Nixon, B. (1992). Representing and using non-functional requirements: A process-oriented approach. *IEEE Transactions on Software Engineering*, 6(4):489–497.

Saeki, M., Hayashi, S., and Kaiya, H. (2009). A tool for attributed goal-oriented requirements analysis. In *Proc. 24th IEEE/ACM International Conference on Automated Software Engineering* (*ASE'09*), pages 670–672.

Tanabe, D., Uno, K., Akemine, K., Yoshikawa, T., Kaiya, H., and Saeki, M. (2008). Supporting requirements change management in goal oriented analysis. In *Proc. 16th IEEE International Requirements Engineering Conference* (*RE'08*), pages 3–12.

Tarr, P., Ossher, H., Harrison, W., and Jr., S. M. S. (1999). N degrees of separation: Multi-dimensional separation of concerns. In *Proc. 10th International Conference on Software Engineering* (*ICSE'99*), pages 107–119.

van Lamsweerde, A. (2001). Goal-oriented requirements engineering: A guided tour. In *Proc. 5th IEEE International Symposium on Requirements Engineering* (*RE'01*), pages 249–263.

van Lamsweerde, A. (2009). *Requirements Engineering: From System Goals to UML Models to Software Specifications*. Wiley.

Yu, E. (1997). Towards modeling and reasoning support for early-phase requirements engineering. In *Proc. 3rd IEEE International Symposium on Requirements Engineering* (*RE'97*), pages 226–235.