Requirements Eng (2014) 19:89–104
DOI 10.1007/s00766-013-0180-9

ORIGINAL ARTICLE

Metrics for measuring the consistencies of requirements with objectives and constraints

JungWon Byun · SungYul Rhew · ManSoo Hwang · Vijayan Sugumara · SooYong Park · SooJin Park

Received: 29 January 2012 / Accepted: 9 August 2013 / Published online: 28 August 2013

Abstract Requirements engineering (RE) is essential to succeed in software projects. That is, it is important to elicit sound requirements for proper performing of RE. If some requirements are inconsistent with a given objectives and constraints, these requirements may hinder the performance of RE. For this reason, it is necessary to precheck the consistency of requirements. We propose the metrics for evaluating requirements by considering given objectives and constraints. Our metrics are validated by two sample scenarios: "Payroll and Roster Planning of Hospital" and "Noise Source Location System." At the result, we expect that these metrics would be helpful when selecting a set of consistent requirements.

Keywords Consistencies of requirements · Metrics of requirements · Objectives and constraints

J. Byun (✉) · S. Rhew · M. Hwang · V. Sugumara · S. Park · S. Park
Seoul, Korea
e-mail: jimi01@ssu.ac.kr

S. Rhew
e-mail: syrhew@ssu.ac.kr

M. Hwang
e-mail: mshwang@shc.ac.kr

V. Sugumara
e-mail: sugumara@oakland.edu

S. Park
e-mail: sypark@sogang.ac.kr

S. Park
e-mail: psjdream@gmail.com

1 Introduction

Various studies have argued that requirements engineering (RE) is critical to succeed in a software project. As discussed in the literature such as [1–3], many projects fail with the improper results of RE. One of the results is an inconsistency of requirements. Causing unnecessary efforts, it can interfere with the good performance of RE. Therefore, it is necessary to measure and validate a set of requirements.

The existing studies have examined the metrics to know whether one elicited requirement is good or not [4–6]. However, their metrics are difficult to apply when evaluating a requirement, because the needed information is not enough in this elicitation phase. So a requirement should be measured by using some useful information that is given from project initiatives to the elicitation phase. At the elicitation phase, a statements of work (SOW) and a software development plan (SDP) can be used to evaluate the requirements.

We propose metrics for measuring the consistency of requirements. The consistency of requirements means how much requirements consist with given objectives and constraints at the elicitation phase. To achieve it, we will describe the concept of our suggested metrics and then introduce how to make an expression of each metric and its interpreted result. These results will be helpful for selecting the consistent requirements. Also, we have performed empirical studies by using sample scenarios, based on existing study [7].

The remainder of this paper is organized as follows. Related work is discussed in the next section. Section 3 proposes the metrics for measuring the consistency of requirements. In Sect. 4, we describe the sample scenario to validate our metrics. Finally, the concluding section offers the summary of our study, limitations, and future work.

2 Related work

2.1 The consistency of requirements

IEEE 830-1998 standard [2] defines the characteristics of a good software requirements specification (SRS) such as consistent. According to this artifact, if each requirement does not agree with some higher-level document, then it would not be correct. Also, a requirement should be compared with any applicable superior specification, including the objectives and the constraints. While there is no exact tool or procedure that ensures this characteristic [2], some stakeholders can determine that each requirement correctly reflects the objectives and constraints.

Farry [8] introduced a principle of RE to make a successful product. He shows the types of requirements errors and their frequency of occurrence. One of the error types is the inconsistent requirements. These requirements may result in overused cost, slipped schedule, frustrated and overworked employees, lack of customer satisfaction, lost profitability, and limited careers.

Leite and Freeman [6] argued the importance of understanding objectives to validate requirements at the elicitation phase. Davis [4] suggested various constraints that can be used to select the requirements. Some requirements have a relevance to both the objectives and the constraints. They propose techniques that represent the relationship between requirements and objectives/constraints as a graph and select requirements with a specific criterion including schedule, cost, and so on. However, their studies have not dealt with the concrete metrics for measuring the relevance of requirements.

Karlsson and Ryan [5] presented the value-oriented prioritization (VOP) technique. In this study, they have assumed that each requirement has its value and its cost. So they have discussed the techniques for requirements prioritization and selection by considering their assumption. However, how to calculate the value and the cost of requirements is not covered in this study.

Various studies [9, 10] have introduced techniques to check and manage the consistency of requirements. Most studies have proposed techniques to validate the consistency between various artifacts and requirements. The difference between some existing studies and ours is that our study focuses on the consistency of each individual requirement with respect to the objectives and constraints, whereas other studies focus on the consistency of a set of requirements with respect to other artifacts. These techniques formalize the requirements and use the objectives and the constraints for the system as input/output factors to check consistency automatically, and requirements which have consistency with objectives and constraints are decided when selected input generates the expected output. Formalization of objectives, constraints, and requirements, however, needs a lot of effort. Furthermore, all of the objectives and constraints do not have equivalent value, and also, their degree of relationship is different.

Wilson et al. [11] proposed an automated tool, Automated Requirements Measurement (ARM). This tool evaluates the ASCII text written requirements specification to comply with a set of defined quality criteria. In this study, a consistent specification has no conflict among requirements. To identify the conflict, they introduce some criteria like constraints on our study. However, we consider not only constraints but also objectives.

Costello's work [12] contains various attempts on metrics for RE. In this article, each requirement should have internal consistency and be consistent with all the other requirements in the specification. However, the study does not suggest the specific metrics to measure the consistency.

As seen from the above discussion, several authors have discussed some criteria and techniques for requirements evaluation. Their studies laid the foundation for our study, and also, they gave us some inspiration to propose our metrics for requirements consistency (See Sect. 3 for the detailed information). It is worthwhile to assess its consistency and to select some requirements which are consistent with the given objectives and constraints at the elicitation phase. In Sect. 2.3, some definitions of terms are presented.

2.2 Metrics for measuring the quality of requirements

Measuring the quality of requirements is the key to achieving our goal of selecting requirements with high quality. To measure the requirements' quality, metrics for RE have been studied by various authors.

Boehm [13] have introduced the "Verification & Validation criteria for requirements," which are completeness, feasibility, testability, and consistency. He has introduced two types of consistency: internal consistency and external consistency. The former means that each individual requirement does not conflict with the others in the specification, and the later means that items in the specification should not conflict with the external specifications or entities. In this study, they proposed approaches to measure requirements quality such as reading, manual cross-referencing, checklist, manual model, and simple scenario. However, they have not realized specific technique and metric to measure the consistency yet.

Davis et al. [14] introduced some metrics to evaluate the quality of SRS. They exhibit a set of metrics that can be applied to SRS, and it includes unambiguity, completeness, and the consistency. Their study suggests two categories of the consistency: *Internal and External Consistency*. The "Internal Consistency" means that a subset of individual requirements is not stated in conflict. In other words, an *internal consistency* is an alignment among requirements. On the other hand, "External Consistency" means that requirements are not stated in conflict with other artifacts (e.g., system-level requirements specifications (RS), SOW, SDP, and white papers). That means an alignment between requirements and other documentations. Among them, our study covers external consistency with the artifacts before the elicitation phase, whereas others focus on after. Davis et al. have introduced a metric to measure the *external consistency* ($Q = n_{ec}/n_r$), where n_{ec} is the number of requirements which are consistent with all other documents and n_r is the total of requirements. The metric can measure a ratio that how many requirements in the specification all other documents consist with them. The study has introduced the metric of external consistency for SRS; on the other hand, our study suggests the metric of external consistency on each requirement.

The Laboratory for Quality Software has developed a model for the software product certification which is called the LSPCM (Laquso Software Software Product Certification Model) [15]. The LSPCM is used to analyze and certify not only the final software product but also intermediate deliverables such as requirement specification. This model includes quality criteria to validate requirements: complete, uniform, correct, and consistent. It consists in external consistency, automated consistency checks, and formally verified consistency. External consistency means that each ambiguous or unclear term from the requirements is contained in the glossary. In order to make requirements to possess the external consistency, the LSPCM recommends that the requirements should be in alignment with an environment description, object model, operation, and behavioral properties. However, this model does not describe an external consistency between each requirement and deliverables that are used as requirements' rationale. LSPCM has proposed 8 metrics for the quality of requirements. In these metric, there are two metrics for the consistency of requirements. One is "Percentage of requirements referring to objects containing unexpected information," and the other is "Percentage of requirements having terms/abbreviations/acronyms." These metrics mean that each requirement is consistent with either glossary or deliverables such as design specification.

2.3 Objectives, constraints, and requirements

In order to propose our metrics, it is important to define the term of *Objectives*, *Constraints*, and *Requirements*. When considering the related literature, these definitions may be summarized as Table 1.

As high-level documents, the objectives and constraints are produced before the elicitation phase; however, the requirements are elicited after that phase. Therefore, the relationship among them has to be analyzed to verify the consistency of elicited requirements from different sources. Figure 1 presents the source of requirements, objectives, constraints, and their relationship by considering the definitions of Table 1. When and where did these come from? As a definition in Table 1, the stakeholders including owners, sponsors, customers, and end users will be a source of objectives and constraints. Such stakeholders, *Business-Level Stakeholders*, may have business and operational knowledge, so that they influence the objectives, constraints, and requirements. Therefore, the acquired requirements have to be consistent with the objectives and constraints. The other stakeholders, *Technical-Level Stakeholders*, have some technical knowledge such as programming skills. They receive the objectives and constraints which are attained by *Business-Level Stakeholders*, and then, they influence on the requirements to be acquired.

3 Metrics for measuring the consistency of requirements

The requirements have to be consistent with the objectives and constraints. In this section, we propose metrics for measuring the consistency of requirements. In the requirements to construct the *Payroll and Roster Planning of Hospital System*, one of the elicited requirements is "The system must have built-in logical checks that make it possible to validate entries of single values against permissible values." In order to make a decision, whether the requirement is accepted or not, we have to measure the consistency of requirements. We focus on how much the requirements are consistent with the high-level documents, especially the objectives and constraints, according to the definition of consistency. It is important to measure the consistency of each requirement with respect to the objectives and constraints.

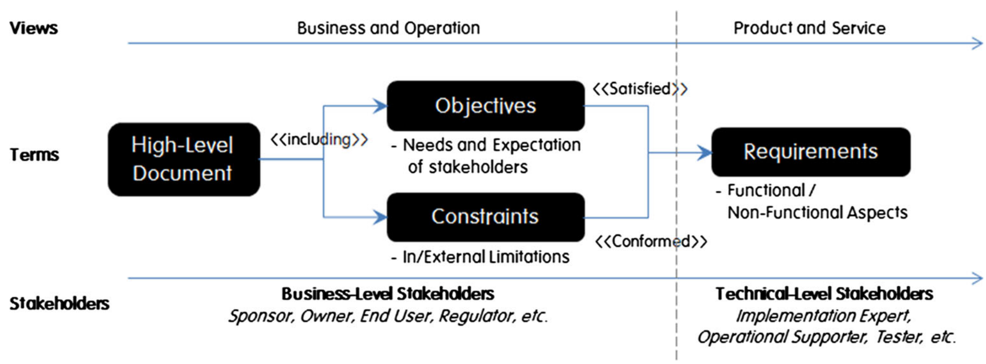
In this section, we will discuss the relationship of each requirement with the objectives and constraints. And then, we will propose consistency metrics that can be calculated from its relationship. Figure 2 shows our scope and approach to measure the consistency of requirements. The high-level documents such as the statements of work and software development plan are required for the objectives and constraints, because these documents have some objectives and constraints; likewise, the SRS is also needed to perform our approach, but they are not covered in this study.

We know the results of how much each requirement contributes to the objectives and/or the constraints when our metrics are applied. If we prioritize these requirements only,

Table 1 Definitions of objectives, constraints, and requirements

Terms	Definitions
Objective	A target that a person or organization seeks to meet [1]
	A statement expected to be accomplished in a relatively short time [3]
	Be elicited from stakeholders' needs and expectations [3]
	Example: To catch users' attentions, GUI design toward "Trendy and Youthful"
Constraint	Internally/externally imposed limitations that have to be or not to be allowed on the next phase [2]
	A limitation placed on the solution designed to the system [3]
	Be identified by stakeholders' operational/environmental characteristics [24]
	A kind of responsibility which is a statement of facts [25]
	Example: The types of products that must be provided by the system are "deposit," "insurance," and "loan"
Requirement	A condition and capability that must be met or possessed by a system to achieve objectives and constraints [26]
	A description of how the system should behave, or specifications of a system property or attribute [27]
	A characteristic of a solution that meets the objectives and constraints [3]
	Something the system must do, or a quality it must have [2]
	Example: the system should provide the "Search Box" to search some services by users

Fig. 1 The main terms and their relationship



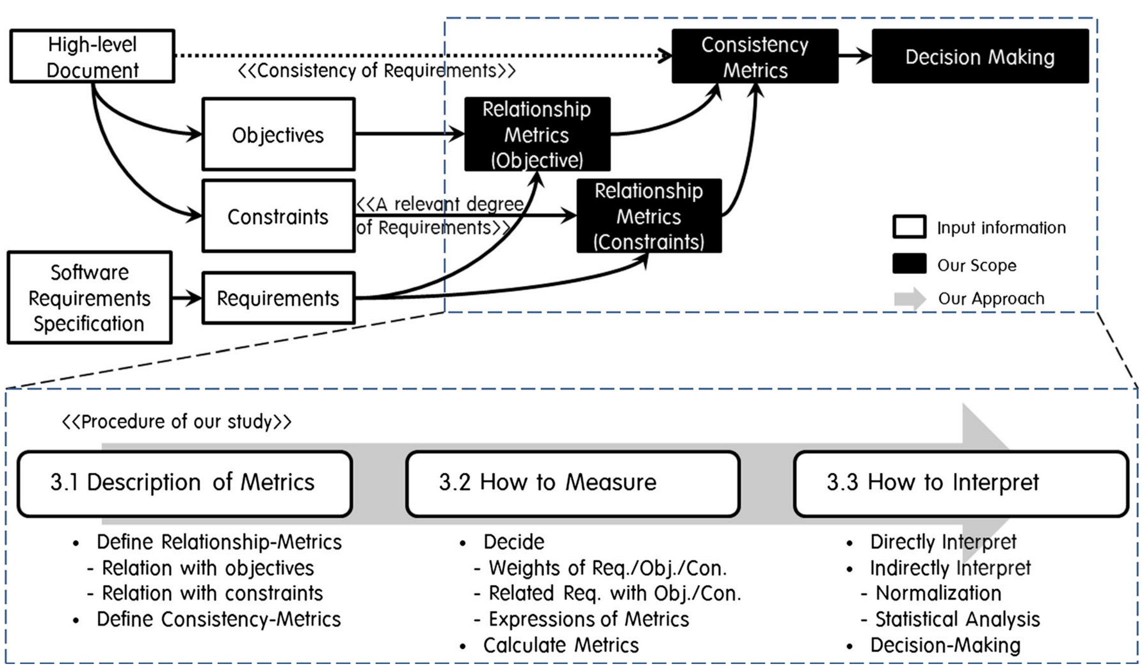


Fig. 2 The given information, scope, and procedure of our work

we will not interpret the results acquired by our metrics. However, if we select some requirements that are expected to meet the given objectives and constraints, we have to analyze these results. In other words, we have to interpret the consistency of each requirement when the criteria for selection are given. The detail information is argued in Sect. 3.3.

The following sections include the following: (a) description of our metrics, (b) how to calculate them, and (c) how to interpret the measured results. Table 2 represents the sample scenario that is discussed in Sect. 3. Since some requirements have been written by natural language and yet refined at elicitation phase, they might be abstract, ambiguous, and mixed with functional and non-functional requirements as shown in Table 2. Even if these requirements would be refined at next phase like analysis phase, such techniques are not covered in this paper.

3.1 The relationship metrics and consistency metrics

Various requirements which are acquired during a project would have their own values and costs, and the relationships among objectives and constraints on them could be used to measure such attributes. The relationships can be measured by the *Relationship-Metrics.* The metrics shows the relationship of requirements with either objectives or constraints; however, the consistency of requirements has to be measured considering both objectives and constraints. These metrics are called *Consistency-Metrics.* We propose six *relationship-metrics* and five *consistency-metrics*. Each metric is described in the following subsections.

3.1.1 The relationship between requirements and objectives

The value of each requirement can be evaluated by measuring how much it is relevant to objectives. We assume that the cardinality of the relationship between requirements and objectives is N:M, so that the value is able to be divided into two relationships. One of them is the relationship between one requirement and various objectives, and it is measured by a metric that is called the *Degree of Objective Contribution* (*DOC*). The other is the relationship between various requirements and one objective, and it is also measured by a metric that is called the *Degree of Objective Satisfaction* (*DOS*). Then, when can we apply these metrics? As shown in Table 2, for example, DOC measures how much R1 is related to O1, and DOS measures how much O1 is related to both R1 and R4.

DOC is a value about one requirement, and DOS is a value about various requirements. So we need another metric to combine DOC and DOS. The metric is called *Value Obtainable for each Requirement* (*VOR*) in this paper. This metric presents a value that summarizes the consistency of each requirement on given objectives at DOC and DOS. As above discussion, we propose three relationship metrics.

Degree of Objective Contribution (*DOC*) is a metric to measure a degree of the relationship between one requirement and various objectives.

DOC $= f_{\mathrm{DOC}}$ (*one requirement, various objectives, Relationship*)

Degree of Objective Satisfaction (*DOS*) is a metric to measure a degree of the relationship between various requirements and one objective.

DOS $= f_{\mathrm{DOS}}$ (*various requirements*, *one objective*, *Relationship*)

Value Obtainable for each Req. (*VOR*) is a metric to measure the value of each requirement by considering the relationship among objectives

VOR $= f_{\mathrm{VOR}}$ (*Objective Contribution Rate, Objective Satisfaction Rate, Relationship*)

The $f(x, y,$ *Relationship*) is a function that measures a relevant degree between x and y with *Relationship*. This function depends on the contexts of an organization and a project. The *Relationship* is given as the values of the matrix in Fig. 3. It is decided by those who have some influence on the objectives including stakeholders in Fig. 1.

In this study, our function is based on multi-criteria decision-making (MCDM) [16], for more information, see Sect. 3.2. When using MCDM, a value of $f(x, y,$ *Relationship*) has to be more than 0. A requirement that has a high value would have a more positive effect on the objectives than one that has a lower value, since the value depends on weights of requirements, related objectives, and the degrees of relationship among them. See a more detailed interpretation of a value in Sect. 3.3.

Figure 3 shows the concepts to consider the relationship between various requirements and various objectives. The relationship of certain requirements such as R2 may be related to various objectives. It is measured by a metric "DOC." This metric means a degree of how much this requirement has an influence on various objectives. Likewise, the relationship of specific objective such as Obj1 may be related to various requirements. It is measured by a metric "DOS." This metric is a degree of how much various requirements have an influence on this objective. Finally, the relationship between various requirements and various objectives can be measured by using the results from both DOC and DOS. The relationship presents a value that will be obtained for each requirement by considering objectives, and its value is measured by a metric "VOR."

3.1.2 The relationship between requirements and constraints

In this paper, the *constraints* will be composed of organizational rules and regulations, governmental legal and

Table 2 The requirements/objectives/constraints (Payroll and Roster Planning of Hospital system)

ID	Requirements	Related obj.	Related con.
R1	The system must be able to handle all collective agreements and contracts under the National Association of Local Authorities	O1	C2, C4
R2	The system must allow registration of the taxes, bank, and pension of employee data	O2, O3	C1, C2
R3	The system must have built-in logical checks that make it possible to validate entries of single values against permissible values	O2	C3
R4	The system must be able to calculate the financial consequences of a given duty roster	O1, O3	C1, C4

ID	Objectives	Related req.
O1	Salary calculation and payment	R1, R4
O2	Pension calculation and payment, plus imaginary pension	R2, R3
O3	Personnel Management	R2, R4

ID	Constraint	Related req.
C1	The parts of the system where the County's functional requirements correspond to a well-defined market standard are only described briefly	R2, R4
C2	Where the functionality is expected to deviate from the expected market standard, the functional description goes into detail. This means the level of detail varies	R1, R2
C3	The system is designed to cater for ease of use as far as possible	R3
C4	All procedures/lists/pictures that are necessary for support, maintenance, operation, and check of databases and files are taken for granted and consequently not mentioned separately	R1, R4

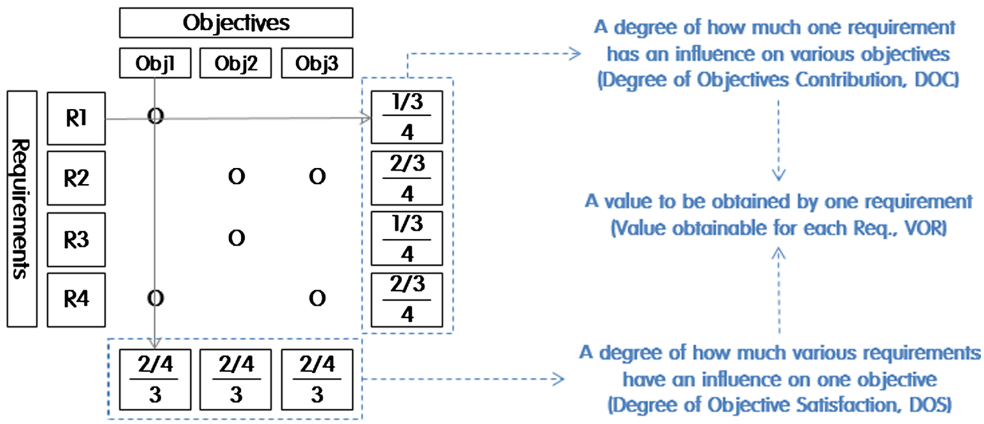


Fig. 3 The relationship between requirements and objectives

restrictions, budget and schedule of a project, and assumptions related to development and operational phases.

The cost of each requirement can be evaluated by measuring how much it is relevant to constraints. We assume that the relationship between requirements and constraints is N:M, so that the cost is able to be divided into two relationships. One of them is the relationship between one requirement and various constraints, and it is measured by a metric that is called the “Degree of Constraint Conformance (DCC).” The other is the relationship between various requirements and one constraint, and it is measured by a metric that is called the “Degree of Constraint Impact (DCI).” Then, when can we apply these metric? As shown in Table 2, for example, DCC measures how much R1 is related to both C2 and C4, and DCI measures how much C2 is related to both R1 and R2.

DCC is a value about one requirement, and DCI is a value about various requirements. So we need another metric to combine DCC and DCI. The metric is called the *Cost Demandable for each Requirement* (*CDR*) in this paper. This metric presents a value that summarizes the consistency of each requirement on given objectives at DCC and DCI. As above discussion, we propose three relationship metrics.

Degree of Constraints Conformance (*DCC*) is a metric to measure the relationship between one requirement and various constraints.

$\text{DCC} = f_{\text{DCC}}$ (*one requirements, various constraints, Relationship*)

Degree of Constraints Impact (*DCI*) is a metric to measure the relationship between various requirements and one constraint.

$DCI = f_{DCI}$ (*various requirements, one constraint, Relationship*)

Cost Demandable for each Requirement (*CDR*) is a metric to measure the cost of each requirement by considering the relationship among constraints.

$CDR = f_{CDR}$ (*Constraints Conformance Rate, Constraints Impact Rate, Relationship*)

The *Relationship* in a function of the proposed metrics is given as the value of the matrix as shown in Fig. 4. The value is decided by those who could have some influence on the constraints including the stakeholders in Fig. 1. When using MCDM, a value of $f(x, y, Relationship)$ has to be more than 0. The requirement that has a higher value would have a more influence on the given constraints than one that has a lower value, since the value depends on weights of requirements, related objectives, and the

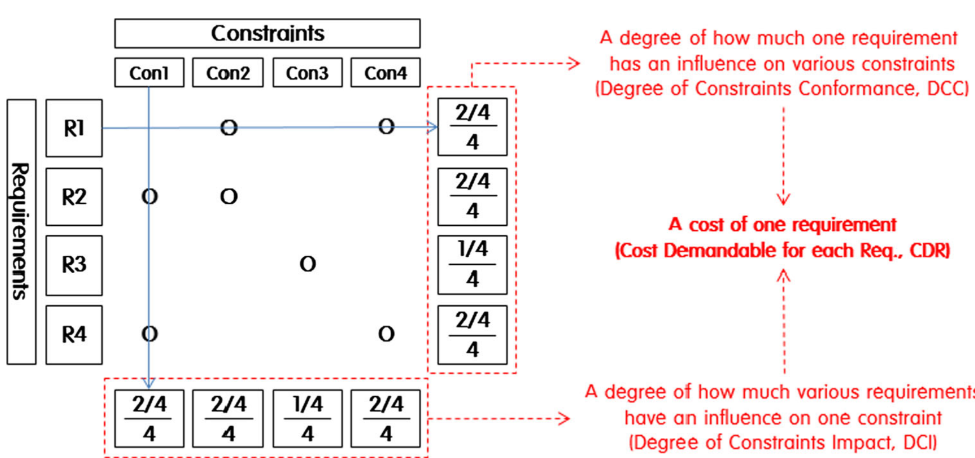


Fig. 4 The relationship between requirements and constraints

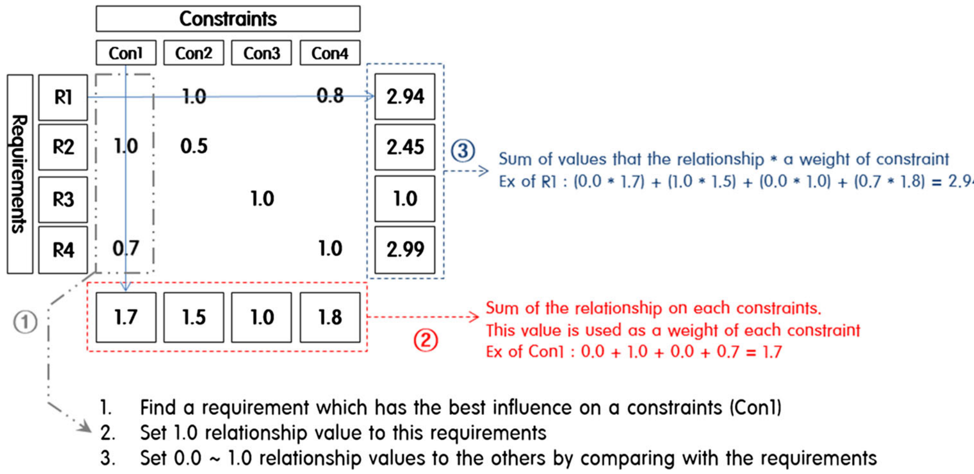


Fig. 5 An approach for deciding our relationship metrics

degrees of relationship among them. See a more detailed interpretation of a value in Sect. 3.3.

Three metrics (DCC, DCI, and CDR) are proposed above. Additionally, we suggest an approach to decide the relationship metrics when some weights on requirements and constraints are not set. In this context, Fig. 5 shows how to decide the relationship and calculate our metrics. A constraint such as Con1 is related to various requirements. Assume that a constraint Con1 is related to requirements R2 and R4, and the relationship with R2 is stronger than R4. This approach says that a relationship with R2 is set to 1.0 and the other (R4) is relatively set from 0.0 to 1.0 by considering R2. And then, the relative consistency of each requirement is calculated considering weights of constraints and relevant degrees of relationships. This approach has the trait that an expert's intuition can be used when a weight should be set, and not only objectives but also constraints can be measured by using relationship metrics and this approach.

3.1.3 *The consistency of requirements confirmed by the objectives and the constraints*

The consistency of requirements has to be measured by considering both the objectives and the constraints. However, the relationship metrics measure either objectives or constraints of each requirement only. Therefore, we propose *Consistency-Metrics* for combinations of the objective-related and constraints-related metrics. Figure 6 shows the data flow diagram to calculate five consistency metrics

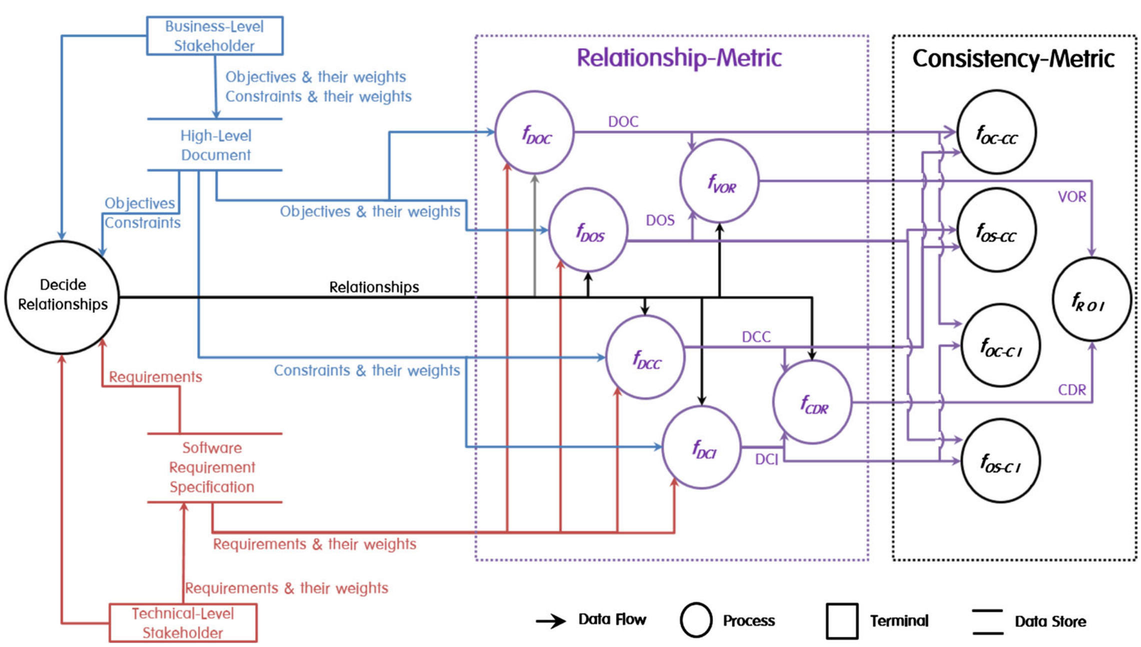


Fig. 6 The proposed consistency metrics

and relationship metrics. The titles and functions of metrics depend on the context of a project or organization. Our consistency metrics are considered with the relationship of each requirement among both objectives and constraints. For more information on how to interpret the value of consistency metrics, see Sect. 3.3.

As shown in Fig. 6, the relationship metrics are applied for either objectives or constraints only. For example, f_{DOC} uses the weights of requirement and objectives, not constraints. But the consistency of each requirement has to consider the relationship between both objectives and constraints and each requirement. So we propose metrics to consider the weight of all of them. These metrics are called "Consistency Metrics." For example, $f_{\mathrm{OC\text{-}CC}}$ is applied for the weights of requirements, objectives, and constraints as shown in Fig. 6.

OS-CC is a metric to measure the consistency of each requirement that is related to one objective and various constraints. *OS-CC = f(DOS, DCC)*

OS-CI is a metric to measure the consistency of each requirement that is related to one objective and one constraint. *OS-CI = f(DOS, DCI)*

OC-CC is a metric to measure the consistency of each requirement that is related to various objectives and various constraints. *OC-CC = f(DOC, DCC)*

OC-CI is a metric to measure the consistency of each requirement that is related to various objectives and one constraint. *OC-CI = f(DOC, DCI)*

Pseudo-ROI for each Req. (*ROI*) is a metric to measure the consistency of each requirement using its value and cost. *ROI = f(VOR, CDR)*

We propose five consistency metrics. Each one of the consistency metrics has a unique meaning; for example, "Pseudo-ROI for each Requirement" is a metric that is based on both a value and a cost of each requirement. The value is considered when a requirement is related to the objectives, and the cost to be needed is considered when a requirement is related to the constraints. The value is measured by VOR, and the cost is measured by CDR. Once again, we propose the consistency of requirements which is a combined VOR and CDR by Pseudo-ROI for each requirement.

3.2 How to calculate the consistency metrics

In the previous section, the weight of requirements, objectives, and constraints is needed to calculate the metrics, but we do not cover how to assign the weights to them, and the weights might depend on various factors such as the viewpoint of stakeholders. The weights are decided by

Table 3 Example to calculate metrics (Payroll and Roster Planning of Hospital system)

	Objective	O1	O2	O3
Requirement	Weight	0.57	0.31	0.12
R1	0.26	1.00	0.00	0.00
R2	0.22	0.00	1.00	1.00
R3	0.22	0.00	1.00	0.00
R4	0.30	1.00	0.00	1.00

the stakeholders and can affect or be affected by developing and managing the requirements, objectives, and constraints, and they can be calculated by using some decision techniques such as the analytic hierarchy process [17]. Please refer to the techniques for details.

For the example of "Payroll and Roster Planning of Hospital system," let us assume that the weights of requirements are [0.26 0.22 0.22 0.30] and the weights of objectives are [0.57 0.31 0.12], as shown in Table 3.

The weight of the *n*th requirement is denoted as a scalar ${}_{\mathrm{W}}S_{\mathrm{R}n}$, and the weights of all requirements are denoted as a vector ${}_{\mathrm{W}}V_{\mathrm{R}}$. In the relationship matrix, a relationship between the *n*th requirement and *m*th objective is denoted as a scalar ${}_{\mathrm{Rel}}S_{\mathrm{R}n-\mathrm{O}m}$; the relationships between the *n*th requirement and all objectives are denoted as a vector ${}_{\mathrm{Rel}}V_{\mathrm{R}n-O}$; and the relationships between all requirements and *m*th objective are denoted as a vector ${}_{\mathrm{Rel}}V_{\mathrm{R-O}m}$. In the following equations, the acronyms "*R*, *O*, *C*, *S*, *V*, *W* and *Rel*" mean the word "*requirement*, *objective*, *constraint*, *scalar*, *vector*, *weight* and *relationship.*" For instance, ${}_{\mathrm{W}}S_{\mathrm{R}n}$ is a weight (scalar) of *n*th requirement, and ${}_{\mathrm{W}}V_{\mathrm{O}}$ is a vector of the weights of all objectives. So ${}_{\mathrm{W}}S_{\mathrm{R3}}$ is 0.22, ${}_{\mathrm{W}}V_{\mathrm{R}}$ is [0.26 0.22 0.22 0.30], ${}_{\mathrm{Rel}}S_{\mathrm{R3-O2}}$ is 1.00, and ${}_{\mathrm{Rel}}V_{\mathrm{R3-O}}$ is [0.00 1.00 0.00].

The concept of MCDM [16] is used to derive the equation for *f*(*x*, *y*, *Relationship*). These techniques can be applied when multiple criteria have to be considered at once. By applying the concept, we can derive the equations for DOC and DOS as follows:

$$\begin{aligned}{}_{\mathrm{DOC}}S_{\mathrm{R}n} &= {}_{\mathrm{DOC}}f_{\mathrm{R}n}({}_{\mathrm{W}}S_{\mathrm{R}n}, {}_{\mathrm{W}}V_{\mathrm{O}}, {}_{\mathrm{Rel}}V_{\mathrm{R}n-\mathrm{O}}) \\ &= {}_{\mathrm{W}}S_{\mathrm{R}n} * ({}_{\mathrm{W}}V_{\mathrm{O}} \circ {}_{\mathrm{Rel}}V_{\mathrm{R}n-\mathrm{O}})\end{aligned}$$

where ${}_{\mathrm{DOC}}S_{\mathrm{R}n}$ is a value of DOC of *n*th requirement, ${}_{\mathrm{W}}S_{\mathrm{R}n}$ is a weight of *n*th requirement, ${}_{\mathrm{W}}V_{\mathrm{O}}$ is a vector of the weights of all objectives, ${}_{\mathrm{Rel}}V_{\mathrm{R}n-\mathrm{O}}$ is a vector of the relationships between *n*th requirement and all objectives, and "'∘' is the inner product of vectors."

$$\begin{aligned}{}_{\mathrm{DOS}}S_{\mathrm{O}n} &= {}_{\mathrm{DOS}}f_{\mathrm{O}n}({}_{\mathrm{W}}V_{\mathrm{R}}, {}_{\mathrm{W}}S_{\mathrm{O}n}, {}_{\mathrm{Rel}}V_{\mathrm{O}n-\mathrm{R}}) \\ &= {}_{\mathrm{W}}S_{\mathrm{O}n} * ({}_{\mathrm{W}}V_{\mathrm{R}} \circ {}_{\mathrm{Rel}}V_{\mathrm{O}n-\mathrm{R}})\end{aligned}$$

where ${}_{\mathrm{DOS}}S_{\mathrm{O}n}$ is a value of DOS of *n*th objective, ${}_{\mathrm{W}}V_{\mathrm{R}}$ is a vector of the weight of all requirements, ${}_{\mathrm{W}}S_{\mathrm{O}n}$ is a weight of nth objectives, and ${}_{\mathrm{Rel}}V_{\mathrm{O}n\text{–}\mathrm{R}}$ is a vector of the relationships between all requirements and nth objective.

The DOC of R1 and DOS of O1 are calculated as below:

$$\begin{aligned}{}_{\mathrm{DOC}}S_{\mathrm{R1}} &= {}_{\mathrm{DOC}}f_{\mathrm{R1}}(0.26\ 0.57\ 0.31\ 0.12], [1\ 0\ 0]) \\ &= 0.26 * (0.57 * 1 + 0.31 * 0 + 0.12 * 0) \\ &= 0.15\end{aligned}$$

$$\begin{aligned}{}_{\mathrm{DOS}}S_{\mathrm{O1}} &= {}_{\mathrm{DOS}}f_{\mathrm{O1}}([0.26\ 0.22\ 0.22\ 0.30], 0.57, [1\ 0\ 0\ 1]) \\ &= 0.57 * (0.26 * 1 + 0.22 * 0 + 0.22 * 0 + 0.30 * 1) \\ &= 0.32\end{aligned}$$

The objective O1 is related to two requirements R1 and R4 as Table 3. The DOS of each requirement can be calculated by expression ${}_{\mathrm{DOS}}S_{\mathrm{R1}} = 0.32/2 = 0.16$. If some requirements have been related to either the certain objective or constraint (such as DOS and DOC), we are able to get a relevant degree of each requirement by using its relationship such as ${}_{\mathrm{Rel}}V_{\mathrm{O\text{–}R}}$ and ${}_{\mathrm{Rel}}V_{\mathrm{C\text{–}R}}$. It is shown in the following equation:

$$\begin{aligned}{}_{\mathrm{DOS}}S_{\mathrm{R}n} &= {}_{\mathrm{DOS}}f_{\mathrm{R}n}\left({}_{\mathrm{DOS}}V_{\mathrm{O}}, {}_{\mathrm{Rel}}V_{\mathrm{R}n-\mathrm{O}}, \sum \left({}_{\mathrm{Rel}}\mathrm{V}_{\mathrm{O}-\mathrm{R}n}\right)\right) \\ &= \sum_i \left({}_{\mathrm{DOS}}S_{\mathrm{O}i} * {}_{\mathrm{Rel}}S_{\mathrm{R}n-\mathrm{O}i} \Big/ \sum_j \left({}_{\mathrm{Rel}}S_{\mathrm{R}j-\mathrm{O}i}\right)\right),\end{aligned}$$

where $i = \{0, \ldots, I\}$, $I =$ the number of Objectives.

If (1) the weights of all objectives ${}_{\mathrm{DOS}}V_{\mathrm{O}}$ are [0.32 0.14 0.06], (2) the relationships between the certain requirement R2 and all objectives ${}_{\mathrm{Rel}}V_{\mathrm{R2\text{–}O}}$ are [0.00 1.00 1.00], and (3) the vector sum about relationships of each objectives $\sum({}_{\mathrm{Rel}}V_{\mathrm{O\text{–}R}})$ are [2 2 2], then the DOS of R2 is calculated as follows:

$$\begin{aligned}{}_{\mathrm{DOS}}\mathrm{S}_{\mathrm{R2}} &= {}_{\mathrm{OMR}}f_{\mathrm{R2}}([0.32\ 0.14\ 0.06], [0.00\ 1.00\ 1.00], [2\ 2\ 2]) \\ &= (0.32 * 0/2) + (0.14 * 1/2) + (0.06 * 1/2) = 0.10\end{aligned}$$

In the same way, the DOCs and DOSs of all requirements are calculated as follows:

$${}_{\mathrm{DOC}}V_{\mathrm{R}} = [0.15\ 0.09\ 0.07\ 0.21]$$
$${}_{\mathrm{DOS}}V_{\mathrm{R}} = [0.16\ 0.10\ 0.07\ 0.19]$$

If the equations of the relationship metrics for constraints are similar to the equations of relationship metrics for objectives, the DCCs and DCIs are calculated as follows:

$${}_{\mathrm{DCC}}V_{\mathrm{R}} = [0.13\ 0.13\ 0.04\ 0.15]$$
$${}_{\mathrm{DCI}}V_{\mathrm{R}} = [0.13\ 0.15\ 0.04\ 0.14]$$

Using the result of relationship metrics, now we can calculate the consistency metrics. The equations are of the form $f(x,\ y)$; they depend on the contexts of the organization. Let us assume that the equation of OS-CC is given as follow:

$${}_{\mathrm{OS-CC}}S_{\mathrm{R}n} = {}_{\mathrm{OS-CC}}f_{\mathrm{R}n}\left({}_{\mathrm{DOS}}S_{\mathrm{R}n}, {}_{\mathrm{DCC}}S_{\mathrm{R}n}\right) = {}_{\mathrm{DOS}}S_{\mathrm{R}n} / {}_{\mathrm{DCC}}S_{\mathrm{R}n}$$

Now, we calculate R1's OS-CC as follows:

$${}_{\mathrm{OS-CC}}S_{\mathrm{R1}} = {}_{\mathrm{OS-CC}}f_{\mathrm{R1}}(0.16,\ 0.13) = 0.16/0.13 = 1.23$$

In the same way, the OS-CCs of all requirements are also gained as follow:

$${}_{\mathrm{OS-CC}}V_{\mathrm{R}} = [1.23\ 0.75\ 1.55\ 1.27].$$

3.3 How to interpret the consistency-metrics

As above discussion, our metrics present a relevant degree of how much each requirement effects the given objectives and/or constraints. If we just prioritize these requirements, we do not need to interpret the acquired results. However, if we select some requirements to met objectives and constraints, we have to analyze these results. In short, we have to interpret the consistency of each requirement when the criteria for selection are given.

In this paper, we have assumed that requirements are selected with consideration of their consistency. The consistency can be measured as a quantitative value when using our metrics. However, our metrics may be difficult to be understood, because they are calculated by a complicated expression. In order to interpret our metrics, we describe techniques to (1) transform a value of metrics into another value and (2) select requirements by using these values.

First, we have to analyze both the value that has to be transformed and the transformed value, because their meanings may differ. If their meanings are different, we have to adjust these values.

Second, we select a statistical technique for changing a value of metrics. When the sample size is 30 or more, the sample distribution follows a normal distribution [18]. In this case, the values of metrics can be transformed into the probabilities, which involve a normal distribution. If the size is below 30, however, we have to consider other available techniques such as the fuzzy theory [19].

Finally, we select the requirements by using the transformed values. The criteria for selection are settled by considering the values before making a decision. These criteria depend on the contexts of an organization and project. The selected requirements have to meet the criteria.

For instance, we change the OS-CC values to the selection probabilities and select consistent requirements to apply the probabilities. The OS-CC values of all requirements ${}_{\mathrm{OS\text{-}CC}}V_{\mathrm{R}}$ are [1.23 0.75 1.55 1.27]. The higher value of OS-CC means that consistency would be high with positive meaning. And then, a high value of requirements selection probability has a good possibility to be selected. The way of transforming values is used to imply a transformation technique. The OS-CC values are changed by using the Gaussian function.

Table 4 Interpreting the values of our metrics using statistical analysis

Requirements	R1	R2	R3	R4
OS-CC	1.23	0.75	1.55	1.27
Probability (%)	54	9	85	58

Mean of samples 1.20, and this value is 50 % chance of selection
Standard deviation of samples 0.33

$$\text{Gaussian function}, f(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(\frac{(x-\mu)^2}{2\sigma^2}\right),$$

where μ is mean of samples and σ is standard deviation of samples

The function needs the *mean* (μ) and *standard deviation* (σ) of OS-CC values. The average of ${}_{\text{OS-CC}}V_{\text{R}}$ is 1.20, and their standard deviation is 0.33. If we change OS-CC value of the requirement R1 (1.23), its probability is calculated as follows:

$f(1.23; 1.20, 0.33) = 0.54$

The selection probability of R1 is 54 % chance. Table 4 shows the probabilities of requirements based on OS-CC values.

The criteria below are developed to select requirements, and they are assigned by considering the meaning of the values such as OS-CC and probability. We have made two criteria as follows:

- **Criterion 1**: The average weight of all requirements, in order to consider the opinions of stakeholders those who can assign the weights. In this case, the weights of requirements are [0.26 0.22 0.22 0.30]. So the average of probabilities is 0.25 (25 %).
- **Criterion 2**: The median weight of all requirements, in order to consider a meaning of the weights. In this case, the range of weights is from 0 to 1. So the median of weights is 0.50 (50 %).

In Table 5, *Accept* means that the requirements are consistent with objectives and constraints above a median of the weights (50 %). On the other hand, *Refinement* means that the requirements are below average of stakeholders' weights. However, it does not mean that the requirements are discarded. These requirements will be interpreted by additional tasks such as another communication with stakeholders [20] and carefully considered in term of their relationship among the objectives and the constraints. It means that they have to be refined in order to be delivered into the next phase in RE.

Table 5 Interpreting the quantities of requirements' probabilities

Requirements	R1	R2	R3	R4
Probabilities (%)	54	9	85	58
Results	Accept	Refinement	Accept	Accept

Selection criteria 1: 25 %, selection criteria 2: 50 %

Accept: 50 % $\leq$ probability, consideration: 25 % $\leq$ probability $<$ 50 %, refinement: probability $<$ 25 %

4 Sample scenario for proposed metrics

In this section, we describe a sample scenario to validate our metrics. The target system is the *Noise Source Location System* (*NSL system*) [21]. Its goal is to measure the sound field around an object. The example of NSL system is published as Fig. 7, and it has been used to various studies. In the given requirements specification of the system, we have identified 8 objectives, 10 constraints, 37 requirements, and their weights. The requirements weight ranges 1–5. The relationship is marked 0 (no related) or 1 (related). Table 6 presents the objectives, constraints, and requirements of NSL system.

4.1 The calculation of consistency-metrics

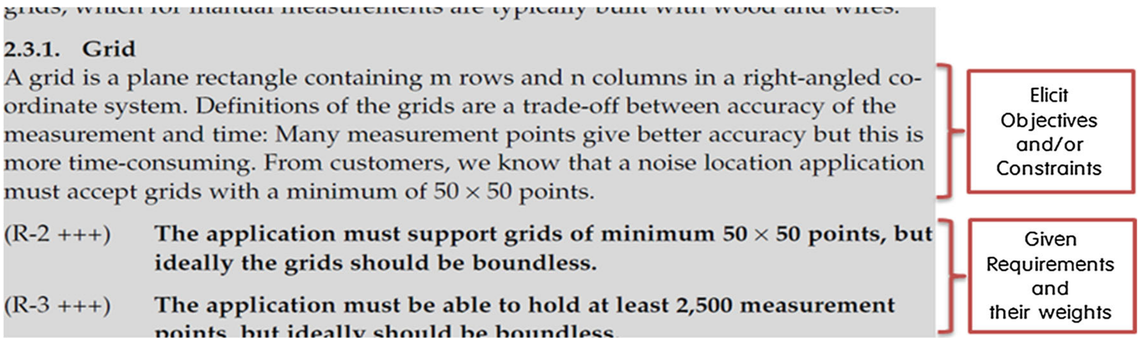
By using the given weights, we are able to calculate the consistency metrics of R1–R37. The equations of the metrics are made by applying a concept of the MCDM, as discussed in Sect. 3. The calculated results are shown as Table 7. For example, in the NSL system, the weight of objective O3 is ${}_{\text{W}}S_{\text{O3}} = 2$, the weight of requirement R2 is ${}_{\text{W}}S_{\text{R2}} = 5$, and the relationship between them is ${}_{\text{Rel}}S_{\text{R2–O3}} = 1$. As calculated results, the DOS of R2 is ${}_{\text{DOS}}S_{\text{R2}} = 10$, the DOC of O3 is ${}_{\text{DOC}}S_{\text{O3}} = 7.50$, and the PV of R2 is ${}_{\text{PV}}S_{\text{R2}} = 75$. Based on the two relationship metrics (PV and PC), we can measure the metric "ROI of each requirement" which is one of our consistency metrics; for example, the ROI of R2 is ${}_{\text{ROI}}S_{\text{R2}} = 0.18$.

4.2 The interpretation of consistency metrics

The measured results from a metric have to be interpreted to select the consistent requirements. If the size of the specified requirements on the specification is above 30, then the results can be transformed into the selection probabilities by using Gaussian function. The average of ROIs is 0.74, and the standard deviation is 0.93. Table 8 shows the probabilities of requirements. For example, the ROI of R2, ${}_{\text{ROI}}S_{\text{R2}} = 0.18$, is changed to the selection probability of R2, $f(0.18; 0.74, 0.93) = 0.28$ (28 %).

For selecting the proper requirements, two criteria are set as follows:

- **Criterion 1**: Probability based on ROI $>$ 0.21 (21 %). This value is a probability when the value of ROI is 0.


2.3.1. Grid

A grid is a plane rectangle containing m rows and n columns in a right-angled co-ordinate system. Definitions of the grids are a trade-off between accuracy of the measurement and time: Many measurement points give better accuracy but this is more time-consuming. From customers, we know that a noise location application must accept grids with a minimum of 50 × 50 points.

(R-2 +++) **The application must support grids of minimum 50 × 50 points, but ideally the grids should be boundless.**

(R-3 +++) **The application must be able to hold at least 2,500 measurement points, but ideally should be boundless.**

Fig. 7 Example to elicit requirements/objectives/constraints of NSL system

Table 6 The sample scenario of Noise Source Location (NSL) system

ID	Weight	Description
(a) *Objectives*		
O1	5	The surface is done by setting up containing the measurement positions
O2	5	The system analyzes positions to sound intensity vector or pressure levels
O3	2	The system displays the results on a picture of the surface or a fraction of the surface
…	…	…
O8	3	The system must be able to display all types of results. The most commonly used data type is 1/3 octave. However, a user may want the high resolution of 1/12 and 1/12 octave or narrow band.
(b) *Constraints*		
C1	5	The sound spectra are measured at different position around a device in which the noise sources are located on
C2	4	A device has values of some properties such as exact places, environments, etc.
C3	4	English is the most common language of our potential customers
…	…	…
C10	4	The recording sequence is very important, but at the same time difficult to communicate
(c) *Requirements*		
R1	3	The language of this system will be in American English
R2	5	The system must support grids of minimum 50 × 50 points, but ideally, the grids should be boundless
R3	5	The system must be able to hold at least 2,500 measurement points, but ideally should be boundless
R4	5	It must be possible to construct a surface consisting of at least 50 grids
…	…	…
R16	4	During the measurement, the system must show the latest measured spectra
…	…	…
R37	5	It must be possible to A-weigh or un-A-weigh the measured spectra. This must always be done on all spectra

A requirement will be above 21 % ROI only if it is related to both objectives and constraints.

- **Criterion 2**: Probability based on ROI $\geq$ 0.40 (40 %). This value is a probability based on the meaning of the weight. The range of the weight is from 1 to 5, and a requirement that has weight >2 is important to NSL system.

In order to make a decision, three categories are set by using these criteria; *Accepted, To Be Considered,* and *To Be Refined.* Let us expand on each of these categories a little: (1) we make a decision as *Accepted* if the requirements are satisfied with both criteria. These requirements will be delivered to the subsequent phase; (2) the category *To Be Considered* includes the requirements that met criterion 1 only. Since these requirements are not consistently enough to make decisions, other tasks or metrics might be needed to determine the suitableness of requirements; and (3) some requirements may be not satisfied by any criteria. They are assigned to the category *To Be Refined*. They are not related to any objectives and/or constraints. In this case, their ROI is 0 and their probability is below 21 %. So they have to be refined by considering

Table 7 Example to calculate our metrics

	Req.	R1	R2	R3	R4	…	R16	…	R37	
(a) *Relationship between requirements and objectives*										
Obj.	W	3	5	5	5	…	4	…	5	DOC
O1	5	0	0	1	1	…	0	…	1	22.50
O2	5	0	0	0	0	…	1	…	1	21.36
O3	2	0	1	1	0	…	1	…	0	7.50
…	…	…	…	…	…	…	…	…	…	…
O8	3	0	0	0	0	…	1	…	0	12.82
DOS		0	10	35	25	…	56	…	90	
PV		0	75	525	563	…	813	…	1749	
(b) *Relationship between requirements and constraints*										
	Req.	R1	R2	R3	R4	…	R16	…	R37	
Con.	W	3	5	5	5	…	4	…	5	DCC
C1	5	0	0	1	0	…	0	…	0	21.25
C2	4	0	0	0	0	…	0	…	1	17.00
C3	4	1	0	0	0	…	0	…	0	14.00
…	…	…	…	…	…	…	…	…	…	…
C10	4	0	0	0	0	…	0	…	0	16.00
DCI		12	30	35	30	…	0	…	40	
PC		168	413	512	413	…	0	…	453	
(c) *ROI of requirements considering PV and PC*										
ROI		0	0.18	1.03	1.36	…	0	…	3.86	

Table 8 The result of interpreting ROI using statistical analysis

Requirements	R1	R2	R3	R4	…	R16	…	R37
The number of related objectives	0	1	2	1	…	4	…	4
The number of related constraints	1	2	2	2	…	0	…	3
ROI	0	0.18	1.03	1.36	…	0	…	3.86
Probability (%)	21	28	62	75	…	21	…	99

Sample average 0.74, this value means 50 % chance of selection

Sample standard deviation 0.93

the objectives and constraints. Table 9 represents the decision of 37 requirements based on their probabilities.

Through two sample scenarios, we believe that our metrics can provide the following benefits. First, the consistency of each requirement can be measured by using our metrics. Our metrics can be used to show the relevant degrees among them, even though we do not know the weights of requirements, objectives, constraints, and their relationships. Second, each requirement has a specific value. This value is not derived by comparing with other requirements, but it is assigned by considering the relationship of requirements with the objectives and the constraints. Finally, our metrics and techniques can be flexibly changed. Some weights may be decided by other techniques, replacing AHP [22]. Some equations may be shaped by using the "Meaning Distance Analysis" in knowledge engineering. Some values can be interpreted by "Fuzzy Theory" [23], not statistical theory. The selection criteria may reflect the stakeholders' opinions.

4.3 Comparison between our approach and the others, and benefit of our metrics

To do a better explanation of our metric and to provide a better justification, we demonstrate the problem solving when the relationship between the requirements and objectives/constraints is not considered by using the suggested metrics. And then, we compare the existing studies with ours to identify the differences between them. We describe the problems that could be happened in situations as follow: (1) a decision-making with the weights of requirements; (2) a decision-making with the relationship of requirements; (3) a decision-making with the metrics for RE, and then, we show that the problems could be possibly solved when applying our study.

Figure 8 shows the results when weights of requirements are used only. This decision-making method includes various techniques (e.g., five-way priority scheme [4]) by many authors and is widely used. Each requirement has a weight from 1 to 5, and most of 37 requirements have weight 4 above. These facts may mean that most requirements have a higher importance among them. That is, it is impossible to select requirements based on their importance (weight) only.

Figure 9 is the results from relationship between requirements and 8 objectives or 10 constraints. Similar to automated consistency measurement approaches [9, 10], this figure shows the total amount of objectives and constraints that are consistent with requirements. Most of the 37 requirements have a counted number of relationships, and the range of number is from 1 up to 7. Among these requirements, some requirements are related to either objectives or constraints only. For instance, a requirement R17 is related to some objectives, but is not related to any constraints. R17 is an important requirement to achieve the related objectives, but is not analyzed as a view of constraints yet. On the other hand, a requirement R01 is related to some constraints only, and this requirement has nothing to do with objectives. Is it truly reasonable to select these requirements by using the number of relationships that they have?

Figure 10 presents the results from the time when our metrics are just applied. This metric can be swapped with existing techniques for scoring of requirements (e.g., value-oriented prioritization). In this case, the values of each requirement are considered by weight and relationship. If a requirement has a number of relationships with the objectives and the constraints and also has a highly weighted value, then its value will be high. But, unlike Fig. 9, if a requirement has no relationship with either objectives or

Table 9 The result of decision-making about requirements

Result (number/total)	Requirements (probability)
Accept (19/37)	R3 (62 %), R4 (75 %), R8 (99 %), R9 (58 %), R10 (46 %), R12 (56 %), R14 (61 %) R15 (71 %), R24 (72 %), R26 (62 %), R27 (65 %), R28 (62 %), R29 (69 %), R31 (58 %) R32 (58 %), R33 (72 %), R34 (72 %), R35 (56 %), R37 (99 %)
To be considered (8/37)	R2 (28 %), R5 (29 %), R6 (23 %), R7 (23 %), R11 (24 %), R13 (25 %), R25 (29 %) R30 (38 %) (*the needed cost for requirements is higher than their earned value*)
To be refined (10/37)	R1 (21 %, *consider the relationship with objectives*) R16–R22 (21 %, *consider the relationship with constraints*) R23 (21 %, *consider the relationship with objectives*) R36 (21 %, *consider the relationship with constraints*)

Accept: 40 % ≤ probability

To be considered: 21 % < probability < 40 %

To be refined: probability ≤ 21 %

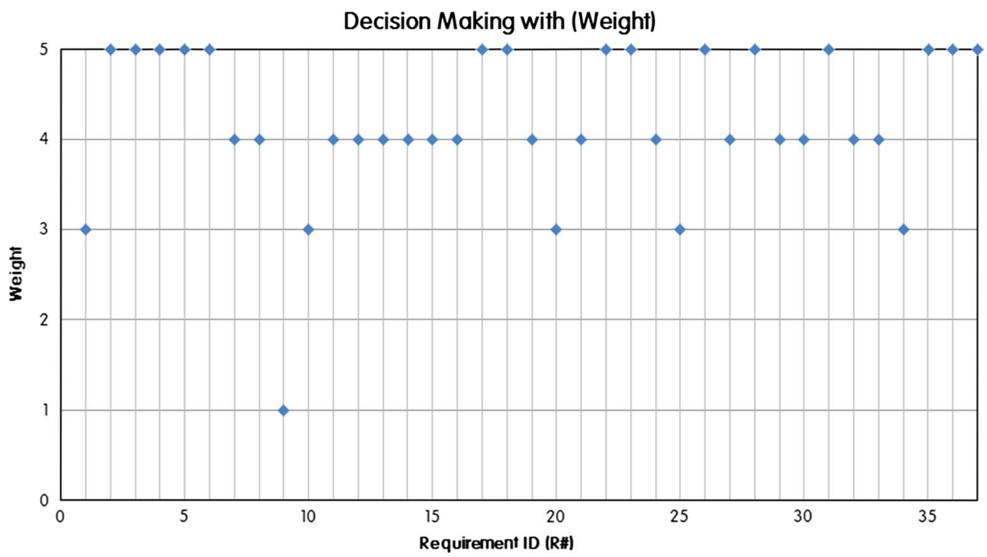


Fig. 8 A result of decision-making with weights of requirements

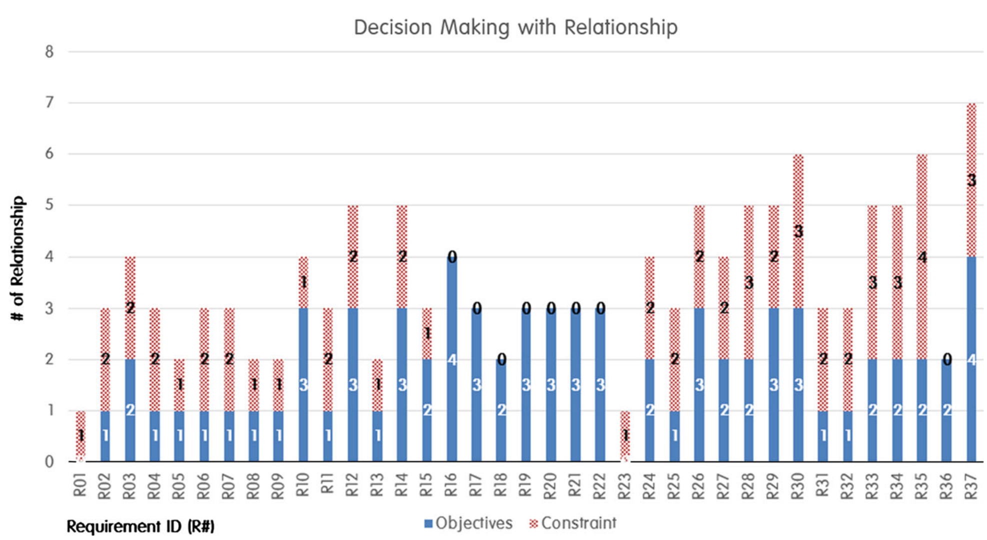


Fig. 9 A result of decision-making with relationship of requirements

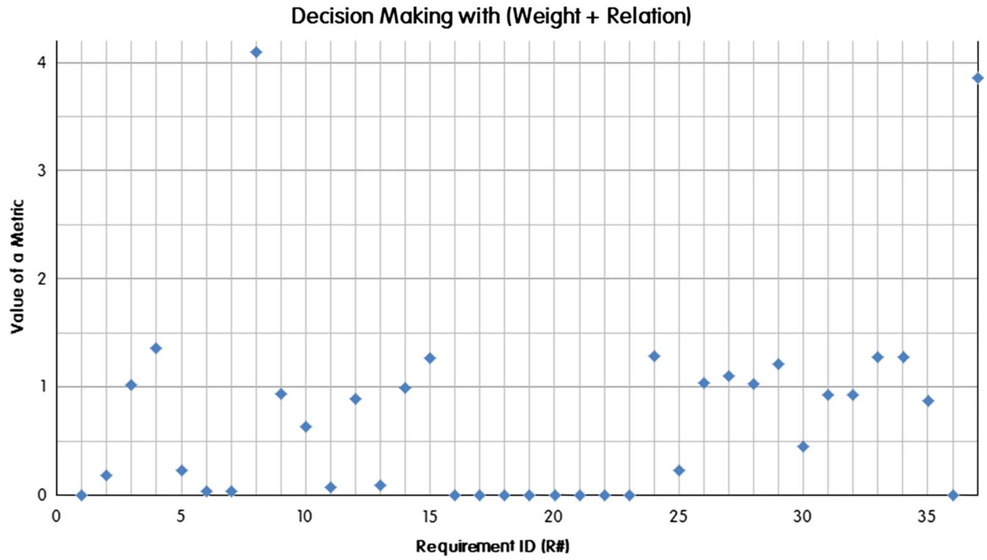


Fig. 10 A result of decision-making with metric

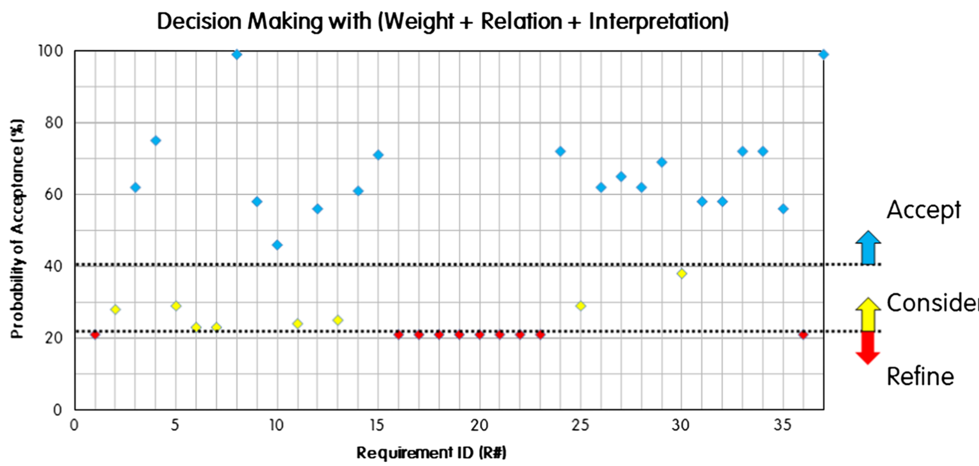


Fig. 11 A result of decision-making with weights, relationship, and interpretation

constraints, its value will be low. Depending on the view of the interpreter, this result could be intuitively categorized into three parts: (1) bigger than 3.5 (includes two requirements); (2) between 3.5 and 0.5 (includes 17 requirements); (3) upper 0.5 (rest of 19 requirements).

Now, some questions could be thrown like what does the each value of 3.5 and 0.5 means? Why should the categories be classified as based on these values? Therefore, we need the criteria and a way to interpret these categorized requirements.

Figure 11 shows the result of our metrics and its interpretation. This graph is similar in appearance to Fig. 10. However, the value of each requirement has a percentage of acceptance from 0 to 100, and the requirements are classified into three groups. We can provide the classification rationale as criteria meaning, which is discussed in Sect. 4.2.

Figure 11 can be more easily interpreted than Fig. 10 as follows: 40 % of acceptance in Fig. 11 is a value of our metric 0.5 in Fig. 10; and 21 % is a value of 0.0. Note that Fig. 11 has a pattern which is similar to Fig. 10 has.

When using the result of comparison, our metrics may have the following benefits.

- Our metrics can be used to validate the requirements by using high-level documents, and then, we can select the consistent requirements. The consistency values of each requirement and the criteria of an organization or project can be used to classify a list of requirements. We can classify 37 requirements into 3 groups (*Accept*, *To be Considered*, and *To be Refined*) when using them.
- Our metrics are able to help each requirement have its own quantitative value. We can use various techniques, such as AHP, KANO [20], and statistical analysis, to justify and to quantify the values of requirements.
- Our metrics can be used to present a relationship of requirements as a graphical way. Our interpretation provides the graph of requirement's consistency to make a decision for selection. Each requirement can be traceable to objectives and constraints by this relationship.

5 Conclusion

We have proposed metrics for the consistency of requirements. Our metrics describe the relevant degrees of requirements by considering the given objectives and constraints. Our metrics consist of six relationship metrics and five consistency metrics. The relationship metric presents a relevant degree of a requirement with either objectives or constraints, whereas the consistency metric presents a relevant degree of a requirement with both objectives and constraints. By using sample scenarios, we have shown that our metrics can be quantitatively calculated as a value, and then, we have described the benefits when our metrics are applied.

In this study, we have shown that our solution can solve the existing problems of requirements selection with weights only, relationships only, and metrics only. Consequently, our metrics and their interpretation have the following benefits: presenting a quantitative value that each requirement has by measuring the consistency of requirements among the objectives and the constraints; communicating with consideration of their distribution.

We have found several questions and limitations on our study. First, the consistency of requirements is important at other phases as well as elicitation phase. Likewise, even though we have focused on the given objectives and constraints, there exist a number of factors that also need to considered. Second, we have applied a few sample scenarios, so that we can validate the availability of our metrics; however, we cannot analyze the effectiveness of our metrics. At last, the requirements analyst needs to communicate with other stakeholders to get a more information for applying our metrics. We know that the above assumption is the ideal situation where the information can be acquired with communication only and is not practical. Therefore, we should apply our metrics in more practical projects.

As a future work, we have to apply our approach to various projects in order to demonstrate the effectiveness of our metrics. So it is important to validate the feasibility of our metrics. Also, we will consider other factors to measure the consistency of requirements. Finally, CASE tools have to be developed to measure the consistency easily and efficiently.

References

1. Karl W (2003) Software requirements. Microsoft Press, Redmond
2. Robertson S (2006) Mastering the requirements process. Addison Wesley, Reading
3. IIBA (2009) A guide of business analysis body of knowledge. International Institute of Business Analysis, Toronto, Ontaria, Canada
4. Davis AM (2005) Just enough requirements management. Dorset House Publishing, New York
5. Karlsson J, Ryan K (1997) A cost-value approach for prioritizing requirements. IEEE Softw 14:67–74
6. Leite JCSP, Freeman PA (1991) Requirements validation through viewpoint resolution. IEEE Trans Softw Eng 17:1253–1269
7. Byun J, Rhew S, Hwang M (2012) A study on techniques for measuring requirement suitability considering project objectives and constraints. In: Lee G et al (eds) Convergence and hybrid information technology, ICHIT 2012, CCIS 310. Springer, Berlin, pp 449–457
8. Farry KA (2001) Customer-centered products: creating successful products through smart requirements management. Amacom, New York
9. Heitmeyer CL, Jeffords RD, Labaw BG (1996) Automated consistency checking of requirements specifications. ACM Trans Softw Eng Methodol 5(3):231–261
10. Lee G, Sacks R, Eastman CM (2002, November) Dynamic information consistency checking in the requirements analysis phase of data modeling. In: European conference for process and product modeling, Slovenia, pp 285–291
11. Wilson WM, Rosenberg LH, Hyatt LE (1997) Automated analysis of requirement specifications. In: Proceedings of the 19th international conference on software engineering ICSE 97, ACM Press, pp 161–171
12. Costello RJ, Liu DB (1995) Metrics for requirements engineering. J Syst Softw 29(1):39–63
13. Boehm BW (1984) Verifying and validating software requirements and design specifications. IEEE Softw 1:75–88. IEEE. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1695100
14. Davis A, Overmyer S, Jordan K, Caruso J, Dandashi F, Dinh A, Kincaid G et al (1993) Identifying and measuring quality in a software requirements specification. In: Proceedings first international software metrics symposium, IEEE Computer Society Press, pp 141–152
15. Serebrenik A, Mishra A, Delissen T, Klabbers M (2010) Requirements certification for offshoring using LSPCM. In: 2010 seventh international conference on the quality of information and communications technology QUATIC. IEEE. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5655261
16. Higgins A, Hajkowicz S, Bui E (2008) A multi-objective model for environmental investment decision making. Comput Oper Res 35:253–266
17. Saaty TL (1980) Analytic hierarchy process. Decis Anal 50:579–606
18. Johnson O (2004) Information theory and central limit theorem. Imperial Collage Press, London
19. Novak V (2005) Are fuzzy sets a reasonable tool for modeling vague phenomena? Fuzzy Sets Syst 156:341–348
20. Byun JW, Kim JH, Rhew SY, Hwang MS (2010) A technique for classifying requirement/stakeholder and generating information for negotiation using Kano model and statistical technique. J KIISE 37:161–169
21. Lauesen S (2002) Software requirements: styles and techniques. Addison-Wesley, Harlow
22. Saaty TL (1988) What is the analytic hierarchy process?. Springer, Berlin, pp 109–121
23. Zadeh LA (1965) Fuzzy sets. Inf Control 8(3):338–353
24. Cox J, Goldratt EM (1986) The goal: a process of ongoing improvement. North River Press, Croton-on-Hudson
25. IEEE (2009) IEEE Std 100-2009, the authoritative dictionary of IEEE standards terms. IEEE Computer Society, Los Alamitos
26. IEEE (1990) IEEE Std 610.12-1990. IEEE standard glossary of software engineering terminology. IEEE Computer Society, Los Alamitos
27. Kotonya G, Sommerville I (1997) Requirement engineering—processes and techniques. Wiley, New York