An Approach: SysML-based Automated Completeness Evaluation of the System Requirements Specification

Jovita Bankauskaite
Department of Information Systems
Kaunas University of Technology
Kaunas, Lithuania
jovita.bankauskaite@ktu.lt

Aurelijus Morkevicius
Department of Information Systems
Kaunas University of Technology
Kaunas, Lithuania
aurelijus.morkevicius@ktu.lt

***Abstract* — Model-Based Systems Engineering (MBSE) is systems engineering methodology that emphasizes the application of strict visual modeling principles. Models are created to deal with complexity, they allow to understand an area of interest or concern and provide unambiguous communication amongst interested sides. MBSE improves the quality of models of the system by providing the ability to evaluate it for completeness, correctness and consistency. MBSE is enabled by Systems Modeling Language (SysML) that supports the analysis, specification, design, verification, validation of complex systems and is used for modeling system requirements, behavior, structure, and parametrics. SysML is not a methodology, nor a method. In this case, it is necessary to choose a specific method in combination with system modeling language to comprehensively and accurately evaluate the completeness of system requirements specification (SRS). This opens up discussions of how to apply SysML provided infrastructure to evaluate the system requirements specification throughout the entire specifying process of SRS and achieve a high-quality of the SRS. In this paper, a new approach of how requirements specification, expressed with sufficient precision in SysML can be used for automated completeness evaluation.**

Keywords—SysML, MBSE Grid, Completeness Metrics, System Requirements Specification, Requirements Engineering, MBSE

I. Introduction

One of the main objectives of the Model Based Systems Engineering is the improved quality of early identification of requirements issues. In order to achieve this goal, the organization needs to implement appropriate practices for modeling qualitatively. Nowadays, MBSE is enabled by Systems Modeling Language. It is used for modeling complex systems such as submarines, trains, aircraft, spacecraft, etc. SysML is intended to create cohesive and consistent models of structure, behavior including their interconnections [1].

Requirements engineering widely recognized as a critical phase in MBSE which consists of two main processes: specification and management. Requirements management is an important part of the discipline that includes the planning, monitoring, analyzing, communicating, and managing of requirements. Poor requirements management is one of the biggest reasons why 47% of project fail to meet goals [2].

In order to avoid issues detection and correction in later stages of development, it is important to identify the issues of incompleteness in the early stages of requirements specification. The mistakes due to incompleteness, inconsistency, and ambiguity introduced at the stage of requirements engineering are difficult and more expensive to correct than those introduced in later stages of system development [3]. The uncertainty of the SRS completeness causes the risk of the requirements change during the development process [4]. Completeness and correctness (C&C) analysis of requirements specification aims to eliminate occurred issues.

In this paper, we focus on a subset of the C&C task – completeness analysis only. We understand the completeness of the SRS as atomic requirements coverage by atomic model elements. The question is how to utilize SysML provided infrastructure to successfully achieve a high quality of the requirements specification: what method to use in combination with SysML.

In this paper, we propose a new approach of how requirements specification that is expressed in SysML in combination with MBSE Grid method can be used for automated completeness evaluation of the system requirements specification.

The MBSE Grid method guides how to specify principal areas of the system model and how to manage different layers of abstraction [5]. The MBSE Grid is organized in a matrix view. Rows represent two main viewpoints: one to define the problem in order to understand it, other to provide one or several alternative solutions to solve it. Columns represent four main aspects of systems engineering (requirements, system structure, system behavior and parameters). Cells of the grid (Fig. 1) represent different views of model-based systems engineering [6]. Specified traceability among view specifications is a very important aspect of the MBSE Grid method. The method helps to organize and maintain the model.

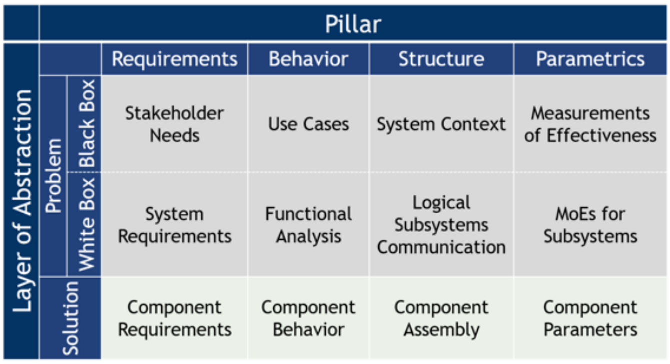

Fig. 1. MBSE Grid

This research is carried out using MagicDraw toolset, which supports SysML. It was chosen because of several published studies, e.g. [7], [8], [9], [10].

The rest of this paper is structured as follows: in section 2, the related works are analyzed; in section 3, the proposed approach for automated completeness evaluation of the requirements specification is presented; in section 4, evaluation of the proposed approach is described; in section 5, the achieved results, conclusions, and future work directions are indicated.

II. RELATED WORKS

A number of evaluation methods and techniques for requirements specification are currently used. Most of them are applied to the small area of the domain or a specific tool, e.g. [11], [12].

Several authors proposed methodologies for evaluation of completeness [13], [14], [12], [15], [16], [17] use formal techniques, e.g. mapping of model elements between successive levels of the refinement hierarchy in [13], service-based domain requirements completeness in [12]. [14] describes an ontology-based approach for completeness verification of a requirements specification. [15] describes an approach to support the quantitative assessment of goal-oriented and scenario-based requirements model completeness. [16] describes how the compositional properties of the formalism can be used to perform completeness analysis.

In [3] publication is proposed three metrics categories of the SRS Completeness: Formal Completeness, Semantic Completeness and Reference Completeness. Formal Completeness - counts the number of elements that are required by meta-classes and searches missing ones. Semantic Completeness –measures a missing semantic element count. Reference Completeness - evaluates the "trace" references leading to the "solutions" and all missing references that are required by the elements meta-classes.

Use of traceability relationships to evaluate the completeness or coverage of the requirements specification has been defined in [18], [19], [20], [21]. An approach in [18] calculates the coverage of a requirement as the degree to which the source code of (e.g. relevant to execute) a requirement is covered by tests. [19] paper describes a model-based testing process directed by structural coverage and functional requirements. The approach in [20] describes measuring the requirements coverage developed as an extension to the value-based approach supported by the TOSCA Testsuite™.

In conclusion, all the analyzed methods to evaluate the completeness of system requirements specification encounter several common issues: (i) unsupported completeness evaluation of particular stage of SRS, (ii) unsupported completeness evaluation during entire specifying process of SRS, (iii) unclear traceability relationships between requirements and design elements, (iv) are applied to the small area of the domain or a specific tool.

Overall, researches carried out in this area have very little proof that they been successfully applied in real-world industry. We are proposing a more generic approach, applicable to the majority of SysML modeling tools for different systems engineering domains. The proposed approach in combination with MBSE Grid will evaluate the completeness of particular stage or entire specifying process of requirements specification. This will ensure the high quality of each stage of SRS.

III. AN APPROACH FOR COMPLETENESS EVALUATION OF SYSTEM REQUIREMENTS SPECIFICATIONS

This section describes the proposed approach in detail.

The approach consists of the following metric groups to evaluate the completeness of SRS that is defined in accordance with the principles of MBSE Grid method:

A. Requirements Refinement Metrics

B. Requirements Satisfaction Metrics

C. Requirements Derivation Metric

D. Requirements Verification Metric

Completeness metrics are based on the determined traceability relationship between requirements and other model elements in the MBSE Grid. Only atomic model elements that are linked to the atomic requirements are included in the calculation of completeness metrics. The relation between atomic requirements and atomic model elements eliminates the ambiguities that may occur having relations between higher level elements.

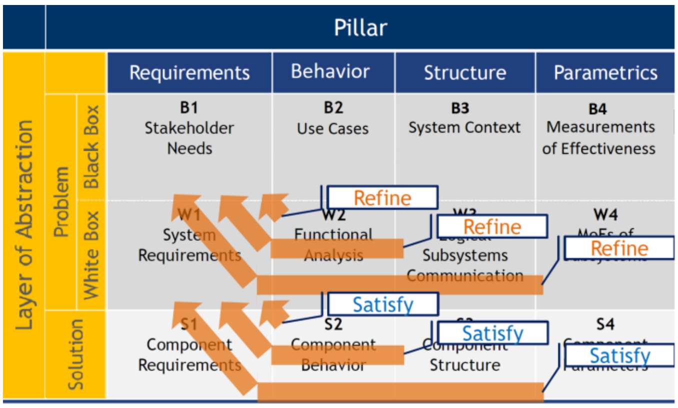

Fig. 2. MBSE Grid Traceability

In order to obtain the more precise evaluation results of requirements specification completeness, metrics are categorized by three aspects of the system engineering: Behavior, Structure, and Parameters. Each elements group of the system aspects coverages the specific requirement category:

- functional requirements are covered by behavior elements. Atomic behavior element - call behavior action which has assigned behavior that does not have owned elements of call behavior actions. Call behavior action has to represent the behavior of the system;
- physical requirements are covered by structure elements. Atomic structure element - block which does not have owned Part Property or Part Property which has assigned Type that not have owned elements of Part Property;
- interface requirements are covered by Proxy Port;
- performance requirements are covered by Value Property.

The proposed method concerns the completeness evaluation of the system requirements specifications. An approach is implemented in the MagicDraw modeling tool.

The subsections below describe in detail each completeness metric of requirements specification.

A. Requirements Refinement Metrics

Metric group of requirements refinement evaluates the completeness of White Box stage of Problem layer in MBSE Grid. The metric group evaluates the refinement of stakeholder needs by model elements which are specified at white box layer.

Requirements refinement metric group consists of the following metrics:

- *Functional Requirements Refinement by Behavior Elements Metric*

This metric evaluates the refinement of functional requirements by behavior elements. This evaluation represents the completeness of Functional Analysis . Below is provided the metric formula.

$$RR_{FR} = \frac{FR_{SN}}{FR} \times 100\% \quad (1)$$

RR_{FR} – functional requirements refinement by behavior elements metric

FR_{SN} – quantity of atomic functional requirements of stakeholder needs refined by atomic behavior element.

FR – quantity of atomic functional requirements of stakeholder needs

- *Physical Requirements Refinement by Structure Elements metric*

This metric evaluates the refinement of physical requirements by structure elements. This evaluation represents the completeness of Logical Subsystem Communication. Below is provided the metric formula.

$$RR_{PhR} = \frac{PhR_{SN}}{PhR} \times 100\% \quad (2)$$

RR_{PhR} – physical requirements refinement by structure elements metric

PhR_{SN} – quantity of atomic physical requirements of stakeholder needs refined by atomic structure element.

PhR – quantity of atomic physical requirements of stakeholder needs

- *Interface Requirements Refinement by Proxy Ports Elements metric*

This metric evaluates the refinement of interface requirements by proxy ports. This evaluation represents the completeness of Logical Subsystem Communication. Below is provided the metric formula.

$$RR_{IR} = \frac{IR_{SN}}{IR} \times 100\% \quad (3)$$

RR_{IR} – interface requirements refinement by proxy ports elements metric

IR_{SN} – quantity of atomic interface requirements of stakeholder needs refined by atomic proxy ports.

IR – quantity of atomic interface requirements of stakeholder needs

- *Performance Requirements Refinement by Parameters Elements metric*

This metric evaluates the refinement of performance requirements by parameters. This evaluation represents the completeness of Measurements of Effectiveness. Below is provided the metric formula.

$$RR_{PR} = \frac{PR_{SN}}{PR} \times 100\% \quad (4)$$

RR_{PR} – performance requirements refinement by parameters elements metric

PR_{SN} – quantity of atomic performance requirements of stakeholder needs refined by parameters element.

PR – quantity of atomic performance requirements of stakeholder needs

B. Requirements Satisfaction Metrics

Metric group of requirements satisfaction evaluates the completeness of Solution layer in MBSE Grid. The metric group evaluates the system requirements satisfaction by atomic model elements which are specified at solution layer.

Requirements satisfaction metric group consists of the following metrics:

- *Functional Requirements Satisfaction by Behavior Elements metric*

This metric evaluates the satisfaction of functional requirements by behavior elements. This evaluation represents the completeness of Component Behavior. Below is provided the metric formula.

$$RS_{FR} = \frac{FR_S}{FR} \times 100\% \quad (5)$$

RS_{FR} – functional requirements satisfaction by behavior elements metric

FR_S – quantity of atomic functional requirements of system satisfied by atomic behavior element.

FR – quantity of atomic functional requirements of the system

- *Physical Requirements Satisfaction by Structure Elements metric*

This metric evaluates the satisfaction of physical requirements by structure elements. This evaluation represents the completeness of Component Structure. Below is provided the metric formula.

$$RS_{PhR} = \frac{PhR_S}{PhR} \times 100\% \quad (6)$$

RS_{PhR} – physical requirements satisfaction by structure elements metric.

PhR_S – quantity of atomic structure requirements of system satisfied by atomic structure element.

PhR – quantity of atomic physical requirements of the system.

- *Interface Requirements Satisfaction by Proxy Elements metric*

This metric evaluates the satisfaction of interface requirements by proxy ports. This evaluation represents the completeness of Component Structure. Below is provided the metric formula.

$$RS_{IR} = \frac{IR_S}{IR} \times 100\% \quad (7)$$

RS_{IR} - interface requirements satisfaction by proxy ports elements metric

IR_S – quantity of atomic interface requirements of system satisfied by proxy ports element.

IR – quantity of atomic interface requirements of the system.

- *Performance Requirements Satisfaction by Parameters Elements metric*

This metric evaluates the satisfaction of performance requirements by parameters. This evaluation represents the completeness of Component measurements of effectiveness. Below is provided the metric formula.

$$RS_{PR} = \frac{PR_S}{PR} \times 100\% \quad (8)$$

RS_{PR} – performance requirements satisfaction by parameter elements metric

PR_S – quantity of atomic performance requirements of system satisfied by parameter element.

PR – quantity of atomic performance requirements of the system.

C. System Requirements Derivation metric

This metric evaluates the system requirements derivation from stakeholder needs. This evaluation represents the completeness of System Requirements. Below is provided the metric formula.

$$RD_{SR} = \frac{SR_D}{SR} \times 100\% \quad (9)$$

RD_{SR} – system requirements derivation from stakeholder needs metric

SR_D – quantity of atomic system requirements derived from stakeholder needs.

SR – quantity of atomic system requirements.

D. System Requirements Verification metrics

This metric evaluates the verification of system requirements by test cases. Below is the formula for evaluating the system requirements verification.

$$RV_{SR} = \frac{SR_V}{SR} \times 100\% \quad (10)$$

RV_{SR} – system requirements verification by test cases metric

SR_V – quantity of atomic system requirements verified by test cases.

SR – quantity of atomic system requirements.

E. Satisfaction evaluation of System Requirements

This subsection describes in the detail the principles of the system requirements satisfaction evaluation by elements that are defined in Component Structure (S3) stage.

The figure below (Fig. 3) represents the system requirements satisfaction by structure and proxy ports elements.

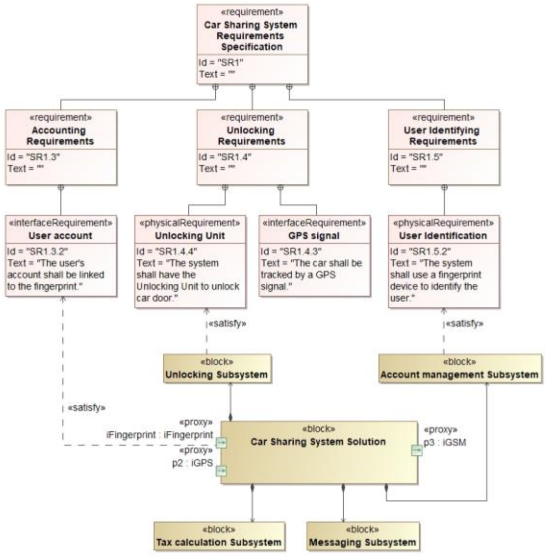

Fig. 3. Satisfaction of system requirements

In the figure below (Fig. 4), is provided the Satisfy Requirement Matrix, which visualizes the satisfy dependencies between system requirements and model element.

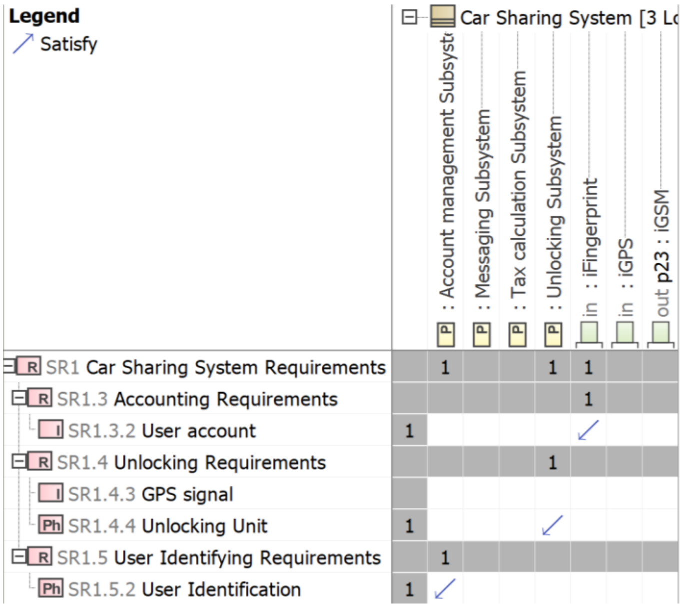

Fig. 4. Satisfy Requirement Matrix

The evaluation process of the system requirements satisfaction by structure elements and proxy ports:

1. The quantity of atomic physical requirements of the system is calculated.
2. The quantity of atomic physical requirements of the system that are satisfied by atomic structure elements is calculated. Only those physical requirements which are satisfied by a block that does not have owned Part Property or are satisfied by a Part Property that has assigned a Type that does not have owned Part Property are included in the calculation.
3. The metric "Physical Requirements Satisfaction by Structure Elements" (6) is calculated using before calculated quantities at first and second step.
4. The quantity of atomic interface requirements of the system is calculated.
5. The quantity of atomic interface requirements of a system that are satisfied by a Proxy Port is calculated.
6. The metric "Interface Requirements Satisfaction by Proxy Elements" (7) is calculated using before calculated quantities at fourth and fifth step.

Following is the evaluation result of systems requirements satisfaction according to Fig. 3 and Fig. 4.

$PhR_S = 2$

$PhR = 2$

$$RS_{PhR} = \frac{2}{2} \times 100\% = 100\% \quad (11)$$

This indicates that 100% of the physical system requirements are satisfied by structure elements that are defined at Component Structure (S2).

$IR_S = 1$

$IR = 2$

$$RS_{IR} = \frac{1}{2} \times 100\% = 50\% \quad (12)$$

This indicates that 50% of the interface system requirements are satisfied by proxy ports that are defined at Component Structure (S2).

IV. CASE STUDY

This section describes the case study of the proposed approach. This is a case study of a commercial project to evaluate the completeness of the system requirements specification.

The commercial project is based on SysML and is modeled in the MagicDraw toolset. The SRS has been modeled according to the principles of MBSE grid.

The completeness of system requirements specification has been evaluated over the whole period of requirements specification process. In the beginning, the manager determined that the coverage of each completeness metric should be at least 90 % in order to move to the next stage of the specification. After each metric calculation, the manager has been analyzed the metric data and made appropriate decision to ensure a high quality of the SRS.

Fig. 5 provides the calculated table of Requirements Refinement in the MagicDraw tool. In order to effectively analyze the metrics data, charts were generated based on calculated metrics data. The charts help the manager to quickly compare data and monitor the progress of SRS completeness.

#	Date	Users Functional Requirements	Refined By Behavior	Refined By Behavior Percentage	Users Physical Requirements	Refined By Design	Refined By Design Percentage
1	2017.11.13	58	38	65.52	33	0	0.00
2	2017.11.30	58	53	91.38	33	0	0.00
3	2017.12.08	58	53	91.38	33	13	56.52
4	2017.12.15	58	53	91.38	33	19	86.96
5	2017.12.20	58	53	91.38	33	22	95.65
6	2017.12.22	58	53	91.38	33	22	95.65

Fig. 5. Metric Table of Requirements Refinement Evaluation

Below is provided a detailed completeness analysis of each metric group. Calculated metric data is provided in the charts.

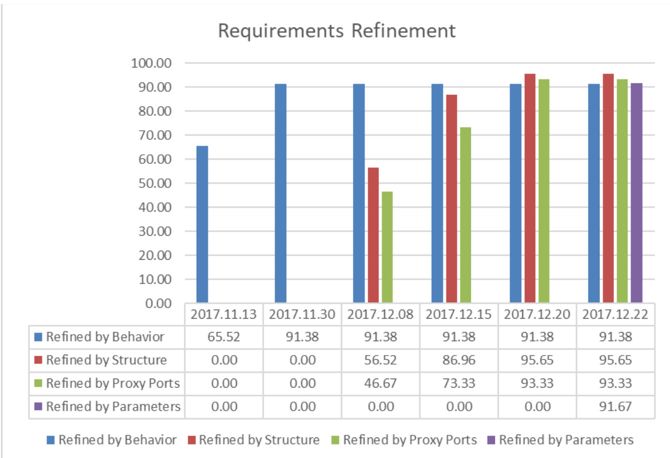

Fig. 6. Requirements Refinement Chart

Fig. 6 provides the requirements refinement analysis chart. Requirements refinement metrics have been calculated over the entire period specifying the problem layer of requirements specification. First of all, functional requirements of stakeholder have been refined by behavior elements. Results of first metric calculation showed that only 65.52% of functional requirements were refined by behavior elements. The decision was taken to continue the specification of requirements refinement. At second metric calculation reaching the 91.38% of functional requirements refinement has been started another stage of the specification, refinement of physical and interface requirements. The specification of physical and interface requirement refinement was continued until the refinement reached over the 90%. When all metrics of refinement reached over 90%, it was decided that the refinement of stakeholder requirements is sufficient.

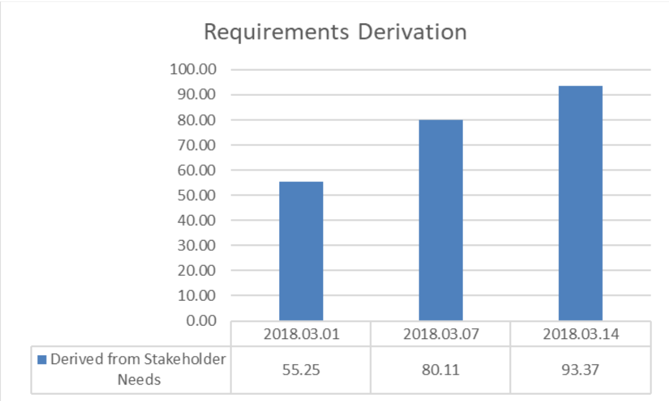

Fig. 7. Requirements Derivation Chart

Fig. 7 provides the analysis chart of system requirements derivation from Stakeholder Needs. Reaching 93.37% of derivation at the third calculation of metric was decided that the system requirements derivation from stakeholder needs is sufficient.

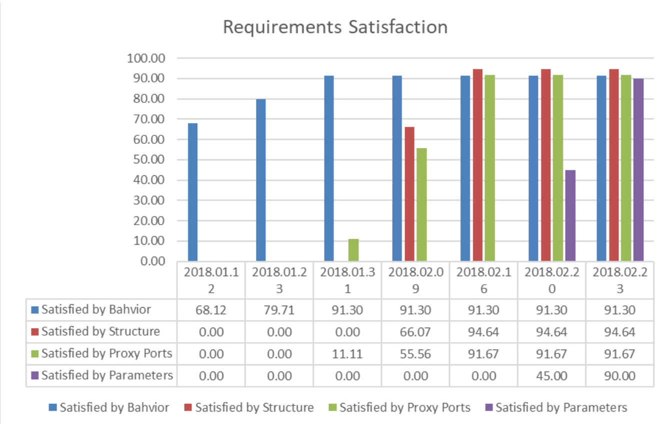

Fig. 8. Requirements Satisfaction Chart

Fig. 8 provides the requirements satisfaction analysis chart. Requirements satisfaction metrics have been calculated over the entire period specifying the solution layer of requirements specification. First, functional requirements of the system have been satisfied by behavior elements. Results of first metric calculation showed that only 68.12% of functional requirements were satisfied by behavior elements. The decision was taken to continue the specification of requirements satisfaction by behavior element. At third metric calculation the satisfaction by behavior reached 91.30% and has been started another stage of the specification, the satisfaction of physical and interface system requirements. The specification of physical and interface requirement refinement was continued until the satisfaction level reached over the 90%. When all metrics of satisfaction reached over 90%, it was decided that the satisfaction of system requirements is sufficient.

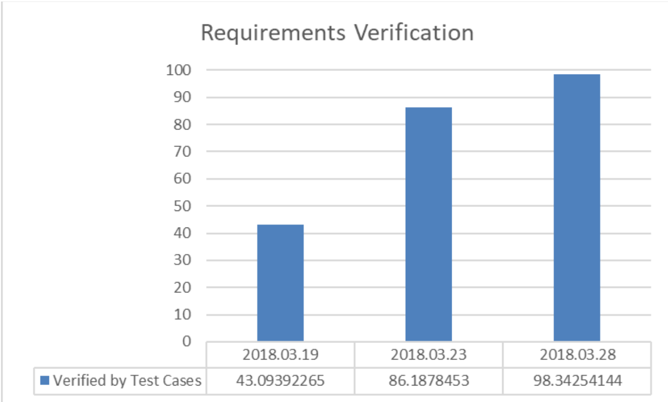

Fig. 9. Requirements Verification Chart

Fig. 9 provides the analysis chart of system requirements verification by test cases. The first calculation of metric showed that only 43.09% of system requirements were verified. The decision was taken to continue the specification of requirements verification. At third metric calculation reaching the 98.38% of system requirements verification was decided that the verification level is sufficient.

V. Conclusion and Future Works

In this paper, we have analyzed the methods for completeness evaluation of system requirements specification. The analysis disclosed that there are many different methods for evaluating the completeness of the SRS, but none of them are appropriate for use in the evaluation of the SRS during the entire period of specifying process. Also, most of the methods cannot be used in combination with systems modeling techniques, such as SysML, in practice. We have identified the need for a more generic approach, applicable to the majority of SysML modeling tools for different systems engineering domains.

In this paper, we have proposed a new approach of how requirements specification, expressed with sufficient precision in SysML, can be used for automated completeness evaluation. The approach is composed of metric groups that are defined on the basis of the principles of MBSE Grid method: Requirements Refinement Metrics, Requirements Satisfaction Metrics, Requirements Derivation Metric, Requirements Verification Metric.

The proposed approach has been implemented in the MagicDraw CASE tool and the case study of commercial project based on the principles of SysML and MBSE Grid has been demonstrated. The analysis of case study disclosed that computing a particular metric group determines the completion of a certain stage of SRS. Completeness calculation reduces the risk of issues detection and correction in the late stage of development and ensures a high quality of requirements specification.

Currently, the approach is oriented to automated completeness evaluation of system requirements specification. We plan to expand this approach in the near future, to more accurately evaluate the completeness and consistency of the system requirements specification.

VI. REFERENCES

[1] S. Friedenthal, A. Moore and R. Steiner, A practical guide to SysML, San Francisco: Morgan Kaufmann, 2014.

[2] P. M. Institute, "PMI's Pulse of the profession®: Requirements management-A core competency for project and program success," Newtown Square, 2014.

[3] Lian Yu, Shuang Su, Shan Luo, Yu Su, "Completeness and Consistency Analysis on Requirements of Distributed," in *2nd IFIP/IEEE International Symposium on Theoretical Aspects of Software Engineering*, 2008.

[4] E. Knauss and C. E. Boustani, "Assessing the quality of software requirements," in *2008 16th IEEE International Requirements*, Catalunya, 2008.

[5] D. Mazeika, A. Morkevicius and A. Aleksandraviciene, "MBSE driven approach for defining problem domain," in *1th System of Systems Engineering Conference*, Kongsberg, 2016.

[6] A. Morkevicius, A. Aleksandraviciene, D. Mazeika, L. Bisikirskiene, Z. Strolia, "MBSE Grid: A Simplified SysML-Based Approach for Modeling Complex Systems," in *27th Annual INCOSE International Symposium*, 2017.

[7] R. Cloutier, M. Bone, "Compilation of SysML RFI- Final Report," 20 02 2010. [Online]. Available: https://www.nomagic.com/mbse/images/whitepapers/omg_rfi_final_report_02_20_2010-1.pdf.

[8] S. C. Spangelo, "Applying Model Based Systems Engineering (MBSE) to a Standard CubeSat," in *Aerospace Conference IEEE*, 2012.

[9] C. Delp, D. Lam, E. Fosse, Cin-Young Lee., "Model based document and report generation for systems engineering," in *Aerospace Conference*, 2013.

[10] P. Godart, J. Gross, R. Mukherjee, W. Ubellacker, "Generating real-time robotics control software from SysML," in *IEEE Aerospace Conference*, 2017.

[11] L. Mangeruca, O. Ferrante and A. Ferrari, "Formalization and completeness of evolving requirements using Contracts," in *013 8th IEEE International Symposium on Industrial Embedded Systems (SIES)*, Porto, 2013.

[12] W. Liu, Chengwan He and Kui Zhang, "Service-based domain requirements completeness analysis," in *2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications (PACIIA)*, Wuhan, 2009.

[13] N. Deb, N. Chaki and A. Ghose, "Using i* model towards ontology integration and completeness checking in enterprise systems requirement hierarchy," in *2015 IEEE International Model-Driven Requirements Engineering Workshop (MoDRE)*, Ottawa, 2015.

[14] T. Avdeenko and N. Pustovalova, "The ontology-based approach to support the completeness and consistency of the requirements specification," in *2015 International Siberian Conference on Control and Communications (SIBCON)*, Omsk, 2015.

[15] C. Gralha, "Evaluation of Requirements Models," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, Beijing, 2016.

[16] M. P. E. Heimdahl, N. G. Leveson, "Completeness and Consistency Analysis of State-Based Requirements," in *17th International Conference on Software Engineering*, Seattle, 1995.

[17] D. Alrajeh, J. Kramer, A. van Lamsweerde, A. Russo and S. Uchitel, "Generating obstacle conditions for requirements completeness," in *2012 34th International Conference on Software Engineering (ICSE)*, Zurich, 2012.

[18] R. Mordinyi, S. Biffl, "Exploring Traceability Links via Issues for Detailed Requirements Coverage Reports," in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, Lisbon, 2017.

[19] Y. Sun, G. Memmi and S. Vignes, "Model-Based Testing Directed by Structural Coverage and Functional Requirements," in *2016 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, Vienna, 2016.

[20] R. Ramler, T. Kopetzky and W. Platz, "Value-Based Coverage Measurement in Requirements-Based Testing: Lessons Learned from an Approach Implemented in the TOSCA Testsuite," in *2012 38th Euromicro Conference on Software Engineering and Advanced Applications*, Cesme, 2012.

[21] P. Rempel and P. Mäder, "Preventing Defects: The Impact of Requirements Traceability Completeness on Software Quality," *IEEE Transactions on Software Engineering*, vol. 43, pp. 777-797, 2017.