Metric-Based Evaluation of Multiagent Systems Models

Lidiane Damasceno
Computational Science Master Program
Universidade do Estado do Rio de Janeiro, Brazil
lidiane.arcoverde@gmail.com

Vera Maria B. Werneck
Informatics and Comp. Science Dept.
Universidade do Estado do Rio de Janeiro, Brazil
vera@ime.uerj.br

Marcelo Schots
Informatics and Comp. Science Dept.
Universidade do Estado do Rio de Janeiro, Brazil
schots@ime.uerj.br

ABSTRACT

The use of Multiagent Systems (MAS) has been increasing over the years due to their capacity of dealing with problems in a variety of domains. The modeling of such systems is not trivial: besides the knowledge and skills on agent-oriented software engineering and a basic understanding of the target domain to be modeled, it demands familiarity with agent-oriented modeling methodologies. This is not always the case, though, especially for newcomers in the field. This work proposes a set of guidelines in the form of a questionnaire for the evaluation of MAS models, aiming at supporting the verification of their quality. The questionnaire is built upon the results of a systematic mapping conducted to identify how MAS models have been evaluated and what metrics have been used. The proposed guidelines were evaluated through (i) a peer review by experts and (ii) its actual application by graduate students applying modeling methodologies in the context of Guardian Angel (GA), a patient-centered health system that automatically supports patients suffering from chronic diseases. Participants provided an overall positive feedback and proposed some improvements on the questionnaire, most of which were promptly incorporated.

CCS CONCEPTS

• **Computing methodologies → Multi-agent systems**; • Computing methodologies → Modeling methodologies • General and reference → Metrics

KEYWORDS

Multiagent systems, model evaluation, quality assurance, metrics

ACM Reference format:

L. Damasceno, V. Werneck and M. Schots. 2018. Metrics for Multiagent Systems Models. In *Proceedings of the 10th Workshop on Modelling in Software Engineering (MiSE 2018), Gothenburg, Sweden, May 2018*, 8 pages. https://doi.org/10.1145/123_4

MiSE'18, May 27, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5735-7/18/05…$15.00
https://doi.org/10.1145/3193954.3193960

1 INTRODUCTION

Agents are autonomous entities capable of making decisions to achieve individual or collective goals. They usually have interactive capacity for negotiation, cooperation, organization, task execution, and social behavior with individual roles. They can also compose systems, called Multiagent systems (MAS) [1].

The use of MAS modeling on several areas is increasingly growing due to its distributed nature and ability to solve complex problems. Examples of areas include energy consumption optimization [15], bee's society behavior [16], urban traffic [17], crowd behavior after disasters [18], among others. Besides using appropriate modeling methodologies, due to the MAS complexity it is also necessary to evaluate the produced models to ensure the quality of the results. This is especially important because of the impact of modeling in the quality of the next development stages.

For identifying what quality criteria are relevant in MAS models for their application in real situations, it is central to understand how they are evaluated in the several areas that have been using them, to check if there is a standardized evaluation.

Based on this motivation, this work proposes a set of verification guidelines and metrics for MAS models, with the following goals: (1) to support the quality assurance team in the assessment of the model quality, being used as a quality checklist, and (2) to serve as a guidance for MAS designers throughout the modeling process, highlighting quality aspects that are important for their models and showing which aspects can be improved. This proposal was elaborated from a systematic mapping conducted to identify kinds of evaluation of MAS models; besides, it is not limited to a specific domain or modeling methodology. The MAS model verification guidelines are inspired in the quality dimensions proposed in [2] and uses the GQM approach [3] to better structure the evaluation questionnaire.

This paper is organized as follows: Section 2 provides some background on the key topics of this work. Section 3 depicts an overview of the systematic mapping on MAS model metrics and the identified evaluation types. Section 4 presents our proposal for verifying the quality of MAS models, whose evaluation is shown in Section 5. Finally, Section 6 presents the final remarks.

2 BACKGROUND

2.1 Multiagent Systems Modeling

The agent-oriented modeling is a powerful and innovative approach for areas requiring robust software that can operate across multiple environments, evolve over time to handle changing requirements, be highly customizable to meet needs and be safe enough to protect personal data. Agent-based technology represents a concrete response to these new requirements [4].

With the increasing use of MAS modeling on several domains, there is a growing need of supporting methodologies for the development of these systems, helping the designer to understand the context of the MAS to be modeled, which is essential for the quality of the modeling activity. Henderson-Sellers and Giorgini [4] highlight some of the main methodologies for the development of agent-based systems; such methodologies support organizing agents so that they can perform their roles in a social environment.

2.2 Software Metrics

The measurement of software products is based on associating values to their attributes or processes, compared with established standards or thresholds for achieving a certain quality level. While one can obtain accurate measures in exact sciences, software measurement involves relative and indirect metrics, depending on patterns considered important for consistent results [5].

Software engineers use metrics to build quality software and to achieve objectivity throughout the software development process. This involves choosing appropriate software metrics, collecting data and analyzed them according to established standards, to assess the process and its products and improve their quality [5].

To support defining appropriate metrics, the GQM (Goal/Question/Metric) approach was developed [3] [6]. This model helps identifying the appropriate metric for a software component or process. First, the measurement *goal* is defined. Then, *questions* are formulated, which are goal improvements with identification of uncertainty areas related to the goals, and finally *metrics* are identified to help answering the questions and confirming if the improvements of the desired goal were reached.

For achieving high maturity levels, it is necessary to comply with established requirements, assessed through well-defined and documented criteria. To achieve this goal, it is common to resort to software quality attribute checklists [19], whose results point out some improvements needed to the target of evaluation.

To better understand the role of metrics in the evaluation of MAS models, we performed a systematic literature mapping [20].

3 SYSTEMATIC MAPPING

3.1 Research Planning and Conduction

The systematic mapping was conducted based on the following primary (PQ) and secondary (SQ) research questions:

- *PQ*: What metrics are being proposed/implemented to measure MAS models?
- *SQ1*: When was the metric proposed/implemented?
- *SQ2*: In which domain the model, the modeling and/or the metrics were evaluated?
- *SQ3*: Were there tools for automating the model evaluation?
- *SQ4*: What artifact elements were target of the measurement?
- *SQ5*: Was there any addition of model information for applying the metric?
- *SQ6*: What were the critical points identified in the model evaluation?
- *SQ7*: Were there any metric applications to real scenarios?
- *SQ8*: Did using the metric lead to an improved model?
- *SQ9*: What was result of the evaluation?

We considered publications in the period between January 2010 and May 2017. The limit of focus to more recent articles is because we wanted to focus the evaluation on the methodologies that are currently being used, avoiding the risk of selecting technical evaluations that may be considered outdated. We understood that publications of the last seven years would be sufficient to identify the current metrics of agent-oriented models.

We developed the search terms in Portuguese (not shown in this paper) and English. The adopted inclusion and exclusion criteria are listed in Table 1. The final search string in English was: ("multi-agent" OR "multiagent" OR "agent-oriented" OR "agent oriented") AND ("metric" OR "measurement" OR "measure") AND ("model" OR "template" OR "modeling").

Table 1: Selection Criteria for the Systematic Mapping

Inclusion criteria	Exclusion criteria
IC1: The publication must be accessible through digital libraries.	**EC1:** A more recent or complete publication describing the same work eliminates the previous one.
IC2: The publication must be either in Portuguese or English.	**EC2:** The publication must not tackle MAS without mentioning model metrics.
IC3: The publication must have been published between 2010 and 2017 (inclusive).	**EC3:** The publication must not address only code metrics.
IC4: The publication must address metrics for MAS models.	**EC4:** The publication cannot be a thesis, dissertation or monograph.
IC5: The publication must be a primary study or at least present some kind of evaluation.	**EC5:** The publication cannot be a short paper or describe an incomplete or ongoing work.

The following steps were performed: (i) Filter by title and abstract; (ii) Removal of duplicate publication; (iii) Selection through screening read; (iv) Selection through full text reading. A total of 9689 publications were returned and filtered from the search sources used in the research, as described in Table 2. After applying the selection criteria, 62 publications were selected.

Table 2: Review Steps and Selection Results

Source	Search results	1st step	2nd step	3rd step	4th step
Google Scholar	1340	148	148	19	13
IEEExplore	640	145	144	91	43
ACM DL	7691	41	39	18	6
Citeseerx	18	1	1	1	0
Total	**9689**	**335**	**332**	**129**	**62**

3.2 Summary of Data Analysis and Results

3.2.1 Metrics proposed/implemented for evaluating MAS models (PQ). From the results, it was possible to notice that the evaluation of MAS models does not have a well-defined or standardized metric applicable for all or most models. Most kinds of evaluation (94%) are performed by simulating test scenarios of the proposed model for later comparison with known behavior results (of either some previous modeling or real situations). When evaluation metrics are used, they usually refer to the modeled context, not to the MAS model.

A major problem identified was the dependence of fixed input data for the chosen test scenarios. The model behavior varies according to the parameters, and divergences of result occur between actual and expected situations. Table 3 lists the kinds of MAS model evaluation and some evaluation metrics identified on the 62 selected publications.

Table 3: Kinds of Evaluation of MAS

Evaluation	Description
Openness, Efficiency and Trust	Openness is the MAS ability to introduce additional agents. The MAS efficiency is related to consistent data, ontology, and behavior cohesion. Trust is maintained at both the individual and the system level.
Test Scenarios	Test situations are created with the purpose of evaluating the model behavior, based on established initial configurations.
Test Scenarios and Results Comparison	The obtained results are compared with an ideal scenario or with a known behavior scenario to verify if the model is in accordance with the expected results.
Case Study	A complex problem or contextualized situation that aims at testing theories and understanding possible solutions.
Case Study and Results Comparison	Results are compared with results from the same Case Study considering other behavior conditions already known, to verify if the model is in accordance with expected results.
Rasch Analysis	Mathematical model to estimate the convenience probability of each item in a satisfaction survey questionnaire. This analysis depends on the difficulty of the item, the ability of the person responding the questionnaire, and considers the unidimensionality and independence in relation to the probability of correctness or error of the item.
Validation Coverage	It aims to establish a standard way to quantitatively measure and communicate the effectiveness of evaluation activities in a model simulation. Five aspects were identified for the metric application: (i) Identification of non-agent entities in the model; (ii) Agents' spatial environment; (iii) Agent lifecycle events; (iv) Agent state changes; and (v) Agent interactions.

3.2.2 Time of creation/implementation of the metric (SQ1). In recent years there has been an increasing number of MAS with some kind of model evaluation. This reinforces the relevance and need to investigate the evaluation of MAS models.

3.2.3 Domain area of the model, modeling and/or metric evaluated(s) (SQ2). It was observed that the MAS modeling predominance in the Energy sector. There are several factors justified on the publications for this use, for example: (i) automate the electrical systems monitoring in a decentralized way, (ii) self-recovery of the power distribution network, (iii) increase operational efficiency through the decentralized approach, (iv) time saving etc. A major factor for increasing the use of agents in electrical networks without human intervention is their autonomy and distributed nature to solve complex problems.

3.2.4 Artifacts for measurement (SQ4). Various elements used for evaluation were identified. Most elements are related to the modeling context, such as average time, performance, price variation etc. Only 23% used agents directly as elements target of the evaluation of MAS models.

3.2.5 Models with additional information for metric application (SQ5). 63% of selected publications did not add information to evaluate MAS models, such as mathematical formulas to evaluate model simulation results. This shows that most publications used qualitative or indirect evaluations, e.g., results comparison with known studies, level of satisfaction etc.

3.2.6 Critical points identified for the model evaluation (SQ6). The main critical points identified for the model evaluation (from a few mentioned in the publications) were the way chosen for the simulation with input parameters or their possibility of changing. Some publications also emphasized the need to increase the tests complexity to obtain a more accurate model assessment.

3.2.7 Applications in real scenarios (SQ7). Real data for model simulation and evaluation were widely used, although not in majority. By simulating a real scenario, one can verify the model result, and make possible adjustments. A simulation close to the reality is often not possible, even with real data, due to complexity factors that influence the result and may not be included in the model. Thus, several simulations with increasing complexity are required to achieve the expected result and improve the model.

3.2.8 Evolution after using the metric(s) (SQ8) and evaluation result (SQ9). Many publications do not cite evolutions from the identified weaknesses for improvements. Only 45% presented some evolution need after the evaluation. Most publications obtained a satisfactory result as an evaluation result. When performing tests of created models, often what is being published has been through several refinements to obtain suitable results. Few publications highlight weaknesses and unsatisfactory results.

From the conduction of the systematic mapping, it was possible to observe the increasing use of MAS modeling and the lack of evaluation standards and processes that guide the designer to carry out a model evaluation. The critical issue that hinders the establishment of standards is the diversity of areas and modeled problems. No publication related to the quality verification of MAS models was found. The full work can be found in [24].

4 QUALITY ASSESSMENT OF MAS MODELS

To help evaluating the quality of MAS models, including whether they reached the goals proposed at the beginning of the modeling (e.g., in an early requirements specification), this work proposes a set of verification guidelines based on quality dimensions.

Some approaches were evaluated with the purpose of defining quality attributes to be used as a basis for the quality dimensions of MAS models. Most approaches are too specific, not being easily adaptable for verifying the quality of MAS models. For instance, the quality factors listed in McCall et al. [8] propose verifications aimed at quantitative level factors that affect software quality, but it focuses on the operational level. The ISO/IEC 25010 standard [9], in turn, defines eight main product quality attributes: functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability and portability. Targeted to software quality, these attributes are less specific, and can be partially adapted to assess the quality of MAS models. Still, this approach lacks generality in the MAS context.

Seeking for a more "open" view of quality, we took inspiration on the quality dimensions proposed by Garvin [2] and used them for creating the quality dimensions of MAS models. Their broader definition of quality facilitates the adaptation to MAS models.

Our goal is not to define standards that will inform whether the generated model has enough quality; instead, we intend to provide both the quality assurance team and the designers with a perception of the points that deserve more attention in the model. We also do not intend to prioritize any dimension; the guidelines are a starting point for these roles to assess how important a given aspect is to their actual modeling context and domain.

4.1 Quality Dimensions of MAS Models

Table 4 presents the first attempt of adaptation/complementation of the Garvin's dimensions [2] from a MAS model viewpoint.

Table 4: Quality Dimensions (adapted from [2])

Dimension	Description
Model Quality	The model must correctly represent the problem domain and have the main concepts.
Performance Quality	Model coverage of all features, functions and results contained in the problem domain. Suitable relationship of the agents to ensure the overall goals of modeling.
Resources Quality	Model agents must adequately represent/manage the problem domain resources. All relevant features must be represented in the model.
Reliability	The model must represent/manage the resources, capabilities and functionality without the occurrence of errors. Changes in the problem domain parameters, as long as they are correct, should not influence the quality of the model representation.
Modeling Quality	The MAS must be modeled according to the best practices of model standards. The methodology used should be adequate to problem domain model.
Durability	Corrections/modifications in MAS generated in the modeling must be performed without generating undesirable side effects in the model. The reliability of the model should increase over time with these corrections/modifications.
Maintainability	The correction time of MAS generated in modeling should be reasonable (not too long). All the documentation and information needed for correction of defects must be available to the system support team.
Evaluation Quality	The results obtained in the evaluation of the model must be in accordance with the expected results for the problem domain. The tests conducted to assess the model must be sufficient to ensure the integrity.
Agents Quality	All the relevant roles and responsibilities of the problem domain must be represented in the model. The skills and tasks of these agents should be sufficient for the problem domain.

4.2 GQM (Goal/Question/Metric) Application

4.2.1 Goal. The goal definition template from GQM was used in the definition of goals for quality assessment and applied for each of the dimensions, as shown in Table 5.

Table 5: GQM-Based Goals Definition

Object of study	Purpose	Focus	Viewpoint	Context factor
Model Quality	evaluate concepts	coverage	designer	problem domain modeling
Performance Quality	evaluate resources, functions and results	coverage of resources, functions and results	designer	problem domain modeling
Resources Quality	verify if agents are represented/ managed adequately	resources	designer	problem domain modeling
Reliability	verify if the agents are represented/ managed without the occurrence of errors	resources, capabilities, and functionalities	designer	problem domain modeling
Modeling Quality	check if the modeling is according to best practices	methodology	quality assurance	software process
Durability	check the generated MAS	flexibility of modification without generating unwanted side effects	designer	problem domain modeling
Maintainability	check the generated MAS	possibility of correction in an acceptable period of time	designer	problem domain modeling
Evaluation Quality	check the compliance of the model evaluation results	avoidance of unexpected results	designer	problem domain modeling
Agents Quality	check the representation coverage	determination of roles and responsibilities	designer	problem domain modeling

4.2.2 Questions. Table 6 presents the separated questions by dimension and divided by components with their quality corresponding assessment metrics. The questions should be answered by evaluating each metric related to the component.

4.2.3 Metrics. After answering the questions by applying the component-related quality metric, the quality team/designer can assess the satisfaction level of the evaluated model. Table 7 shows the description of each metric identified. These metrics and their definitions were extracted from [9], [10] and [11]. Fig. 1, in turn, illustrates (as an example) the Completeness metric and its relationship with the goals and questions components.

Table 6: Questions Quality Dimensions

	Question	Component	Metric
Model quality	Q1: In general, does the model represents the problem domain correctly?	Representation	Correctness
			Compliance
	Q2: Does the model have the main problem domain concepts?	Concepts	Completeness
			Conciseness
			Context Coverage
	Q3: Has the model met expectations and initial goals?	Expectations	Adequacy
			Compliance
		Goals	Adequacy
			Compliance
Performance quality	Q1: Does the model contain all the features related to the problem domain?	Resources	Completeness
	Q2: Does the model contain all the functions related to the problem domain?	Functions	Completeness
	Q3: Does the model contain all the results related to the problem domain?	Results	Completeness
	Q4: Do the agents have a suitable relationship in order to ensure the modeling goals?	Agents relationship	Effectiveness
			Compliance
	Q5: Do the agents have suitable communication in order to ensure the modeling goals?	Agents Communication	Effectiveness
			Compliance
Resources quality	Q1: Does the model provide agents that adequately represent the problem domain resources?	Resource representation	Adequacy
	Q2: Does the model provide agents that properly manage the problem domain resources?	Resource Attributes Management	Adequacy
Reliability	Q1: Do model agents represent features without errors?	Resource Representation	Correctness
	Q2: Do model agents manage resources without errors?	Resource Management	Correctness
	Q3: Do model agents represent capabilities without errors?	Capacity Representation	Correctness
	Q4: Do model agents manage capabilities without errors?	Capacity Management	Correctness
	Q5: Do the model agents represent the functionalities without the occurrence of errors?	Functionalities representation	Correctness
	Q6: Do model agents manage functionalities without errors?	Functionalities Management	Correctness
	Q7: Is the parameters change of the problem domain safe for not affecting the model representation quality?	Changing of Parameters	Correctness
			Effectiveness
	Q8: Is the model safe enough to protect stakeholders' data?	Data Protection	Security
Evaluation quality	Q1: Are the model evaluation results aligned with the problem domain expected results?	Evaluation Results	Compliance
	Q2: Have the tests performed to evaluate the model been sufficient to ensure its integrity?	Tests	Completeness
		Model	Integrity
	Q3: Have all model components been tested?	Model Components	Completeness
	Q4: Are the evaluation results within the common range?	Results within the Common Range	Accuracy
Agents quality	Q1: Are all the roles of the problem domain represented in the model?	Roles	Completeness
	Q2: Are all the responsibilities of roles of the problem domain represented in the model?	Role Responsibilities	Completeness
	Q3: Are agent skills sufficient for the problem domain?	Agent Skills	Context coverage
	Q4: Are the tasks performed by the agents sufficient for the problem domain?	Agents Tasks	Context coverage
	Q5: Do the agents have a suitable perception of the environment, in order to react to the stimuli to achieve the design goals?	Environment Perception	Compliance
			Efficiency
		Reactive Agents	Compliance
			Efficiency
	Q6: Are agents proactive at the time needed to achieve design goals?	Proactive agents	Compliance
			Efficiency
Durability	Q1: Can the generated MAS model be modified without generating unwanted side effects to the model?	Multiagent System	Modifiability
	Q2: Will the changes make reliability increase over time?	Change	Maturity
Maintainability	Q1: Can the generated MAS model be fixed in an acceptable period of time?	Multiagent System	Efficiency
			Modifiability
	Q2: Is the information/ documentation required for defect correction available to the system support team?	Documentation	Availability

Table 7: Metrics Description (based on [9], [10] and [11])

Metric	Description
Accuracy	Ability to provide, with the required precision degree, correct results or effects as agreed upon.
Adequacy	Ability to provide functions that meet stated and implied needs when used under specified conditions.
Context coverage	Ability to be used effectively, efficiently, free from risks, both in specific contexts of use and in other contexts.
Completeness	Ability to lack nothing required.
Conciseness	Ability to use what is strictly necessary.
Compliance	Ability to comply with standards, conventions, regulations, or similar requirements.
Correctness	Ability to be free of (known) errors or faults.
Availability	Ability to be operational and accessible when needed.
Effectiveness	Ability to achieve specified goals with accuracy and completeness, within a specified context of use.
Efficiency	Ability to present appropriate performance, relative to the amount of resources used, under specified conditions.
Integrity	Ability to be complete, i.e., not missing any of its parts.
Maturity	Ability to meet reliability needs in normal operation.
Modifiability	Ability to be effectively and efficiently modified without presenting defects or degrading the product quality.
Security	Ability to protect information and data, providing people, products, or systems with proper data access degrees, with corresponding authorization types and levels.

4.2.4 Level of Satisfaction. To ease answering according to the level of satisfaction for each metric related to each component, we used an colored scale for each question: DN (Does not address), MR (Addresses with many restrictions), PA (Partially addresses), A (Addresses), and O (Overcomes). The designer can then work on the metrics/components/questions with low satisfaction levels.

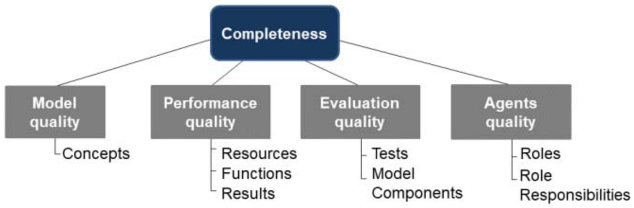

Figure 1: Completeness Metric

5 EVALUATION

We sent the MAS quality verification guidelines for expert evaluation and for the application by graduate students. We prepared an Excel worksheet containing the quality assessment questionnaire. This worksheet consists of four tabs. The first one contains instructions for filling out the questionnaire (for designers) and for reviewing (for experts); the second one contains the quality assessment questionnaire, with the dimensions and metrics descriptions (shown in Table 6); the third one presents the detailed result achieved for each metric after filling out the questionnaire, and the fourth one presents the final results of the dimensions, metrics, and the satisfaction levels.

5.1 Expert Evaluation

A worksheet with the verification model was sent to eighteen experts (in domains listed in Table 8), five of which returned with their evaluation pointing out several suggestions for improving the questionnaire. The main changes are: *a*) We made explicit that the checklist should be applied after finishing the design and before implementation. Dimensions and questions that verified the implemented MAS were excluded. *b*) We changed and detailed the description of questions for a better understanding. *c*) The response unit became the metric, rather than the question component; *d*) We included some suggested metrics. *e*) We changed the name and description of some dimensions.

Table 8: Expert Profiles

Expert	Areas of expertise
Expert 1	MAS development, modeling, software processes, quality, requirements engineering, software transparency and non-functional requirements.
Expert 2	Software engineering, software process improvement, software quality, software processes, process performance analysis, knowledge management, project management and requirements engineering. Practical experience with services and software process improvement.
Expert 3	Requirements engineering, software engineering, MAS development, software quality, goal-oriented modeling, knowledge-based systems and medical systems.
Expert 4	Software engineering, working mainly on software systems scalability, requirements engineering, sustainability in systems engineering, goal-oriented modeling and software quality.
Expert 5	Virtual reality, artificial intelligence, software engineering, computer science education and MAS.

The resulting verification model is presented in Tables 9 and 10. Fig. 2 illustrates the new relationships of the Completeness metric with the goals and questions components.

Table 9: Improved Version of the Quality Dimensions

Dimension	Description
Model Expectations	The model must correctly represent the problem domain and meet the initial expectations and needs.
Resources Quality	Model agents must adequately represent/manage the problem domain capabilities. All relevant features must be represented in the model.
Reliability	The resources, skills/capabilities and goals represented in the model must be organized in a way that allows the system recovery in the occurrence of errors. The model must be flexible enough to allow changes in system parameters, ensuring the problem domain goals.
Methodology Adequacy	The MAS modeling must comply with the standards proposed in the methodology. The methodology and the tool used should be adequate to the problem domain model.
Maintainability	The agents' responsibilities must be modularized to facilitate their later maintenance and avoid unwanted MAS side effects. All the necessary documentation and information for any defect corrections must be available.
Agents Quality	All the relevant roles and responsibilities of the problem domain must be represented in the model. The agents' skills and tasks must be necessary and sufficient, and there must be a suitable relationship between agents, in order to ensure the modeling goals.

Table 10: Improved Version of the Verification Model

	Question	Component	Metric
Model Expectations	Q1: Do the model elements and interactions in the model meet the needs of the problem domain?	Problem Domain Needs	Adequacy
			Compliance
	Q2: Does the model contain all relevant goals (necessary and sufficient) for the problem domain?	Goals	Completeness
			Conciseness
	Q3: Do the model elements and interactions meet the identified expectations for the user?	User Expectations	Adequacy
			Compliance
Resources Quality	Q1: Does the model present all relevant resources (necessary and sufficient) for the domain problem?	Resource Definition	Completeness
			Conciseness
	Q2: Does the model present agents that properly represent or manage the problem domain resources?	Resources Representation/ Management	Adequacy
			Completeness
Reliability	Q1: Are the resources represented in the model organized in order to allow the system recovery in an eventual occurrence of error?	Resource Representation	Correctness
	Q2: Are the competences/capacities represented in the model organized in order to allow the system recovery in an eventual occurrence of error?	Competences/ Capacities Representation	Correctness
	Q3: Are the goals represented in the model organized in order to allow the system recovery in an eventual occurrence of error?	Goals Representation	Correctness

	Question	Component	Metric
	Q4: Is the model flexible enough to allow changes in the system parameters, ensuring the meeting of goals of the problem domain?	Changing of Parameters	Correctness Effectiveness
	Q5: Does the model contain goals related to information security, allowing privacy and protection of the stakeholders' data?	Data Privacy/ Data Protection	Security
Methodology suitability	Q1: Does the MAS model comply with the methodology standards?	Methodology	Compliance
	Q2: Is the MAS methodology proper for modeling the problem domain?	Methodology	Adequacy
	Q3: Did the tool used for modeling represent the model elements adequately?	Tool	Adequacy
Agent quality	Q1: Are all the relevant roles (necessary and sufficient) of the problem domain in the model?	Roles	Completeness Conciseness
	Q2: Are all relevant roles' responsibilities (necessary and sufficient) of the problem domain represented in the model?	Role Responsibilities	Completeness Conciseness
	Q3: Are the agents' skills and abilities necessary and sufficient to meet the problem domain goals?	Agent Skills/Abilities	Completeness Conciseness
	Q4: Are the tasks/actions performed by the agents necessary and sufficient to meet the problem domain goals?	Agent Tasks/Actions	Completeness Conciseness
	Q5: Are agents able to perceive the environment and its information?	Environment Perception	Efficiency
	Q6: Are agents able to react to an event or context situation perceived in the environment?	Reactive Agents	Efficiency
	Q7: Are agents able to detect the need to act before being asked to do so, based on their goals and environmental information?	Proactive Agents	Efficiency
	Q8: Do the agents have a suitable relationship in order to ensure the modeling goals?	Agent Relationships	Effectiveness
	Q9: Do the agents have suitable communication in order to ensure the modeling goals?	Agent Communication	Effectiveness
Maintainability	Q1: Are the agents' responsibilities modularized in the model, to facilitate further maintenance and avoid unwanted side effects?	Agent Responsibilities	Accuracy
		Multiagent System	Efficiency Modifiability
	Q2: Have the model information and documentation required for eventual defects correction been prepared and made accessible for further maintenance?	Documentation	Availability

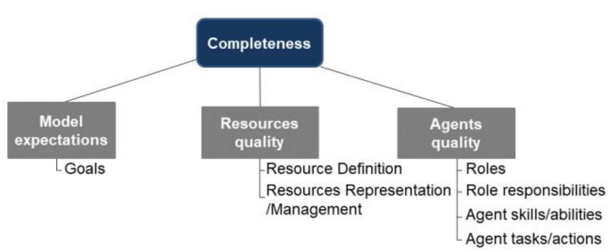

Figure 2: Completeness Metric After Model Evolution

5.2 Application

The proposed guidelines were applied by four graduate students at the Master Course level. These students had a MAS discipline, during which they conducted a modeling exercise, for a better assimilation of the subject. The exercise consisted of modeling a MAS in the context of the Guardian Angel (GA) project [12], a patient-centered health system for automatic support for chronic diseases that has also been used for evaluating agent-oriented methodologies [13] [14]. In this discipline, the methodologies applied were Prometheus [22], Ingenias [21], and MaSE [23].

Students answered the quality assessment questionnaire after the GA modeling, based on the models they produced. The average time reported by them to complete the questionnaire was about 30 minutes. They gave their opinion about their models and the results of the evaluation. Positive responses include: "the questionnaire is well organized and objective"; "I found it very comprehensive, addressing several aspects (dimensions)"; "I think the questionnaire is cohesive and potentially useful".

They also reported some difficulties, most of which related to understanding some technical terms used in the previous version of the questions. Although some changes have already been made to the questionnaire (as mentioned previously), we created a glossary of terms and definitions used, to support its completion.

We also improved the scale of possible answers (acronyms and descriptions) to ease the identification of levels of satisfaction: NA (The model does not address this metric at all); MR (The model addresses this metric partially, with many restrictions); FR (The model addresses this metric partially, with few restrictions); TA (The model addresses this metric totally); and OE (The model overcomes expectations for this metric) (if applicable). Fig. 3 presents an example of result (after improving the questionnaire).

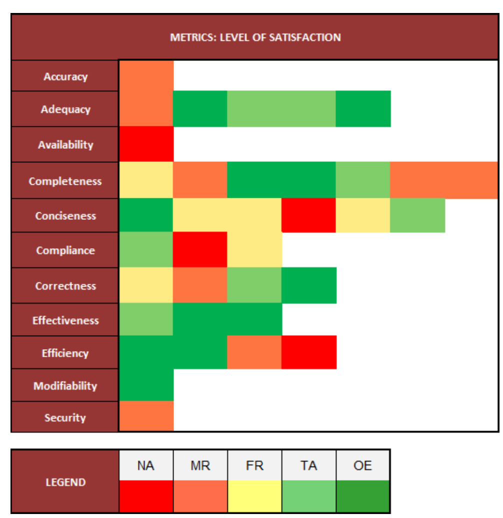

Figure 3: Example of Evaluation Result

6 FINAL REMARKS

The research presented in this paper investigated knowledge about the evaluation forms that have been used in MAS models. There is no evaluation standard, since each publication carries out the most convenient evaluations according to the context of the modeling, which allows the evaluation to be open, flexible, and also subject to errors. In software engineering, standards and processes help ensuring suitable results, but MAS have several particularities in relation to object-oriented approaches that make them unique, requiring a more targeted evaluation.

The systematic mapping conducted provided an overview of the area, allowing the opportunity to propose an evaluation process of MAS models. This work made it possible to understand their evaluations and verify the need for standardization. The identified kinds of evaluations also helped to shape the needs and difficulties in their execution. We proposed guidelines for the quality evaluation of MAS model to support the model evaluation process, helping the quality assurance team with a quality checklist and serving as a guidance for MAS designers throughout the modeling process. With these guidelines, the model can be evaluated considering the importance of a particular aspect, providing a perception of points that require more attention.

The proposal was evaluated through a peer review process executed by experts (in different areas of expertise, as listed in Table 8) and through the application of the checklist by graduate students who designed models of the Guardian Angel project in the Prometheus [22], Ingenias [21] and MaSE [23] methodologies. The new version of the model was sent for expert evaluation, which recognized the improvement with positive feedback. Some of them also suggested some minor improvements to the new version, which are under preparation.

As future work, more specific checklists can be developed for handling particularities of the different MAS modeling methodologies, making a more targeted quality check. Besides, we intend to ask students and practitioners to evaluate multiple models with and without the questionnaire, comparing their results. This would allow to assess if the questionnaire supports revealing hidden knowledge, correlating the questionnaire results with unguided human judgment. We also intend to develop a system to replace the Excel worksheet in the questionnaire application, providing improvements in terms of user interaction.

ACKNOWLEDGMENTS

We would like to thank the experts who gently providing feedback on this work and the graduate students who applied the questionnaire, for their time and contributions to this research.

REFERENCES

[1] Michael J. Wooldridge. 2002. An Introduction to Multiagent Systems. England: John Wiley & Sons Ltd, 348p.

[2] David Garvin. 1987. Competing on the Eight Dimensions of Quality. Harvard Business Review, 101-109.

[3] Victor R. Basili and Hans D. Rombach. 1988. The TAME project: towards improvement-oriented software environments. IEEE Trans. on Software Engineering, 758–773.

[4] Brian Henderson-Sellers and Paolo Giorgini. 2005. Agent-oriented methodologies. USA: Idea Group Publishing, 413 p.

[5] Ian F. Sommerville. 2011. Software Engineering: Addison Wesley, 529 p.

[6] Rini V. Solingen, Victor R. Basili, Gianluigi Caldiera and Hans D. Rombach. 2002. Goal question metric (GQM) approach. Encyclopedia of software engineering.

[7] V. Basili, G. Caldiera, H. Rombach. 1994. Goal Question Metric Paradigm. Encyclopedia of Software Engineering, v. 1, John J. Marciniak, Ed. John Wiley & Sons, pp. 528-532.

[8] Jim A. McCall, Paul K. Richards and Cene F. Walters. 1977. Factors in Software Quality, three volumes, NTIS AD-049-014,015,055.

[9] ISO/IEC 25010: 2011: Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models.

[10] ISO Standard 9126: Software Engineering – Product Quality, parts 1, 2 and 3", International Organization for Standarization, Geneve, 2001 (part 1), 2003 (parts 2 and 3).

[11] PUC-Rio. 2018. Transparency Catalog (Catálogo Transparência) (in Portuguese). Available at: <http://transparencia.inf.puc-rio.br/wiki/index.php/Cat%C3%A1logo_Transpar %C3%AAncia>. Accessed in: 11 Jan. 2018.

[12] Peter Szolovits, Jon Doyle, William J. Long, Isaac Kohane and Stephen G. Pauker. 1994. Guardian Angel: Patient-Centered Health Information Systems. Technical Report MIT/LCS/TR-604. Available at: <http://groups.csail.mit.edu/medg/projects/ga/manifesto/GAtr.html>. Accessed in 26 Jan 2018.

[13] Felipe Cordeiro, Vera Maria B. Werneck, Neide dos Santos and Luiz M. Cysneiros. 2016. MAS Ontology: Ontology for Multiagent Systems. In *18th International Conference on Enterprise Information Systems (ICEIS 2016)*, pp. 536-543.

[14] Eric Yu and Luiz M. Cysneiros. 2002. Agent-oriented methodologies-towards a challenge exemplar. In 4 Intl. Bi-Conference Workshop on AOIS, Toronto. v. 151.

[15] Leo Raju, Kaviya Appaswamy, Janani Vengatraman and Antony Amalraj Morais. 2016. Advanced Energy Management in Virtual Power Plant using Multi Agent System. 3rd International Conference on Electrical Energy Systems, p. 133-138.

[16] José Eduardo M. de Figueiredo, Vera M. B. Werneck, Rosa M. E. M. da Costa. 2013. A 3-D Serious Game to Simulate a Living of a Beehive. Virtual, Augmented and Mixed Reality. Systems and Applications (VAMR 2013), pp. 363-371.

[17] Shouren Wang, Zhongjian Dai and Jing He. 2013. A Multi-Agent Simulation of Traffic System under Mixed Traffic Condition. Third International Conference on Instrumentation, Measurement, Computer, Communication and Control, China, p. 1646-1649.

[18] Florin Leon, Gabriela M. Atanasiu and Georgiana Bunea. 2015. Critical Assessment of Road Network Resilience Using an Agent-Based Evacuation Model. Proceedings of the 7th Balkan Conference on Informatics Conference, Romenia, n. 21.

[19] Roger S. Pressman. 2011. Software Engineering: A Practitioner's Approach, 7. Ed.

[20] K. Petersen, Sairam Vakkalanka and Ludwik Kuzniarz. 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. Information and Software Technology, 64 (2015), pp. 1-18.

[21] J. Pavón, J. J. Gómez-Sanz, R. Fuentes-Fernández. 2005. The INGENIAS methodology and tools. In: Henderson-Sellers, B., Giorgini, P. (eds) Agent-Oriented Methodologies, Chapter IX, pp. 236–276. Idea Group Publishing, Hershey.

[22] Lin Padgham and Michael Winikoff. 2005. Developing intelligent agent systems: A practical guide. Vol. 13. John Wiley & Sons.

[23] S. A DeLoach and M. Kumar. 2005. Multi-agent Systems Engineering: An Overview and Case Study. In: Brian Henderson-Sellers and Paolo Giorgini (eds.), Agent-Oriented Methodologies, USA: IDEA Group Publishing, pp 317-340.

[24] L. D. M. Arcoverde. 2018. Guidelines for Evaluation Multiagent System Models: A Metric-Based Proposal (Diretrizes para Avaliação de Modelos de Sistemas Multiagentes: Uma Proposta Baseada em Métricas) (in Portuguese). M.Sc. Thesis, Rio de Janeiro State University, Rio de Janeiro, Brazil.