Evaluation of Multi-Agent Systems: Proposal and Validation of a Metric Plan

Pierpaolo Di Bitonto, Maria Laterza, Teresa Roselli, and Veronica Rossano

Department of Computer Science, University of Bari
Via Orabona 4, 70125 Bari, Italy
{dibitonto,marialaterza,roselli,rossano}@di.uniba.it

Abstract. In the MAS evaluation research field there are still few works devoted to evaluating systems' efficacy, and none of these aimed to measure the adequacy of the MAS in terms of rationality, autonomy, reactivity and environment adaptability. A reliable evaluation method should be general enough to estimate the success of the multi-agent paradigm in different domains, measuring the performances of each single agent and then of the entire MAS. Moreover, it should be able to relate these measures to the environment complexity, that embodies the complexity of the problem solved by the MAS. In this paper a method for evaluating static multi-agent systems is presented and its validation described. The main novelties of the method are that it allows the MAS to be evaluated in the context of the environment in which it will operate, and its adequacy to the environment to be judged from the viewpoints of both the designer, wishful to measure the quality of the designed MAS, and the evaluator, wishful to verify the adequacy of several MASs in a specific context. A validation of the method is described, carried out by evaluating two MASs: the GeCo-Automotive system and a Multi-Agent Tourism Recommender system.

Keywords: Multi-agent system, Goal-Question-Metric, MAS evaluation.

1 Introduction

In the last few years the growing employment of Multi-Agent Systems (MASs) in several domains, including logistics, networking, automation, simulation and robotics, has provided the impetus for much research into new tools and methodologies for their design and implementation. Although researchers in the MAS field have proposed a huge number of solutions, there are still few works addressing valid methods for evaluating MASs.

MAS evaluation is a complex process that should take into account several dimensions, considering a MAS not only as an aggregation of single agents, but also as a system in which the agents must interact in order to solve problems.

A reliable evaluation method should be general enough to estimate the success of the multi-agent paradigm in different domains, measuring the performances of each agent and of the entire MAS, and should be able to relate these measures to the

N.T. Nguyen (Ed.): Transactions on CCI VII, LNCS 7270, pp. 198–221, 2012.

environment complexity. In fact, the environment complexity embodies the problem that needs to be solved by the MAS.

Against this background, the aim of our research is to define a method for evaluating static multi-agent systems consisting of a metric plan based on the Goal-Question-Metric paradigm [1] and a set of guidelines to interpret the results of the GQM application.

The metric plan allows measurement of the complexity of the environment where the agent acts, as well as the level of rationality, autonomy, reactivity, and adaptability to the environment exhibited by the MAS. The paper presents a MAS evaluation method that is an evolution of the first proposal described in [2]. The newest version is more detailed than the previous one, because it splits MAS characteristics and environment complexity into different evaluation dimensions. By comparing each MAS dimension with the relative environment complexity, the evaluator can gain a more accurate evaluation of the adequacy of the MAS to the environment.

One of the main novelties of the proposed method is that it merges two different approaches, namely intra-agent and inter-agent, to the analysis of multi-agent systems. The intra-agent approach analyses the MAS agent as an individual system, highlighting the internal structure, the beliefs, the goals, and the perceptions related to its environment. The inter-agent approach considers each single MAS agent as a part of a society and analyses its interaction with the other agents of the system and its environment. Moreover, a strong point of the proposed method is that it can be used for two evaluation purposes: on one hand, to estimate the adequacy of the MAS from the designer point of view, allowing this specialist to check whether the implemented MAS is adequate to cope with the problem constraints, and on the other, to estimate the adequacy of a MAS from the evaluator point of view, verifying which MAS, among a set of similar systems, is the best suited to solve a specific problem. In order to validate the method's independence of the problem domain and to investigate the efficacy of its application according to the different evaluation goals (designer vs. evaluator), two MASs have been evaluated: GeCo-Automotive [3] and a Multi-Agent Recommender for Tourism [4]. The former aims at developing an ICT environment to manage small-medium sized company knowledge about automotive spare parts; this MAS is evaluated from the designer point of view, since the authors are the designers of the system. On the other hand, the Multi-Agent Recommender for Tourism [4], that aims to promote tourism in Argentina, is evaluated from the evaluator point of view in order to measure the adequacy of this system to a specific context in which the evaluator wishes to use it. It is important to notice that, the Multi-Agent Recommender for Tourism, developed by Casali et al. [4], is evaluated only to show how the defined GQM could be applied in order to verify if a MAS is adequate to a specific environment problem.

The paper is organized as follows: section 2 discusses some related works about the evaluation of multi-agent systems; section 3 presents the defined metric plan, using the GQM paradigm; section 4 describes the guidelines to interpret the metric plan measures, relating these measures to the environment complexity; sections 5 and 6 describe the application of the defined method to the two MASs. Finally, some conclusions and future research directions are outlined.

2 Related Works

Analysis of the literature shows that several approaches have been proposed to evaluate MAS quality. The first proposed approaches stem directly from the field of software engineering because the agent-based paradigm was originally considered as an evolution of object-oriented programming (the agents are often implemented using object-oriented programming languages). In this perspective, the aspects evaluated are only related to the software quality of each single agent. Higher level characteristics such as MAS organizational models or interaction among agents are still not considered. The most popular metrics collected in the suite of metrics for O.O. design are those proposed by Chidamber and Kemerer [5]. These metrics, based on object-oriented programming key concepts (class, method, inheritance, coupling), measure the software quality in terms of coupling between object classes, depth of the inheritance tree and so forth [6]. For example, the metric coupling between object classes measures how many classes each class is coupled with. It allows estimation of both the reusability and the software code efficiency, because a high coupling value between classes will mean that there is low modularity and reusability. These metrics seem too low-level to be meaningful for agent-based systems [7], but the next, higher level approaches allow estimation only of some aspects of MAS such as the architecture and communication among agents, considered mainly as distributed systems. For example, in [8], Król and Zelmozer focus attention on the structural performance of multi-agent platforms. In particular, they consider only Java RMI implementations and define metrics such as the connection cost metric, serving to predict how well different implementations are suited to various network configurations and environments. In [9] the intent is to propose a set of metrics for measuring the communication among MAS agents in order to detect reasons for an unbalanced communication. But, as emphasized in [10], the current trend in the MAS evaluation field should go beyond the hardware and software implementations. For this reason, the authors propose an approach that captures the messages exchanged by the application agents and extracts useful information serving to draw a communication graph. On the basis of this graph they calculate the value of metrics such as the degree of communication, the number of agents involved in communication, the network mean traffic, and so on. Following this trend, the newest research works have aimed at defining metrics for measuring higher level characteristics of MASs. In their recent work [11], Lass et al. survey existing metrics employed to estimate MASs, provide an evaluation framework for applying them and use this framework to compare the performances of some distributed algorithms. They classify the metrics as environment/host metrics and system metrics. The first ones describe the MAS environment (i.e. the physical world in the case of a robot, or users, services and other MAS agents in the case of a software agent) and allow the environment complexity to be measured. The second ones measure macroscopic aspects of the MAS as a whole, and therefore describe the overall behavior of the MAS. The evaluation framework consists of three main steps: selection, collection and application. In the selection step the evaluator chooses the metrics to be used to evaluate the MAS. In the collection step the measures are collected. Finally, in the application step, the measures collected are used to assess whether the MAS meets the evaluation objectives, or is better than another one. Because of the huge number of existing metrics and ways to apply them, the authors suggest the use of Basili's GQM approach [1] to decide which metrics are most usefully measured in their MAS. The work lacks a ready-to-use metric plan that could be adopted to measure and compare them. Moreover, although evaluating MASs seems to be very important in order to be able to predict the MAS performance and to design systems suited to various environments, MAS characteristics such as rationality, autonomy, reactivity and the environment's adaptability are still not evaluated and no approaches have yet been described in literature that are able to evaluate both the characteristics of the agents in the MAS and the characteristics of the overall MAS. For all these reasons, in [2] we proposed a MAS evaluation method that differs from the other approaches cited in literature in four principal ways. Firstly, it proposes the use of high-level metrics to evaluate the MAS, and emphasizes the measurements of agent characteristics such as rationality, autonomy, reactivity and adaptability to the environment. Secondly, the defined method merges two MAS evaluation perspectives: inter-agent and intra-agent. The inter-agent evaluation considers the overall MAS (cooperation and communication among agents), whereas the intra-agent evaluation considers the internal structure of each single agent (in terms of its ability to learn, planning capabilities, and so on). Thirdly, it provides a metric plan for assessing MASs. Fourthly, the method allows a MAS to be evaluated from the viewpoints of both the designer, wishful to measure the quality of the designed MAS, and the evaluator, wishful to verify the adequacy of several MASs in a specific context.

3 The Metric Plan

In [2], a metric plan based on the GQM approach is described. This means that the MAS assessment can be made independently of its specific implementation and context of use. The plan has five goals. The first assesses the complexity of the environment where the MAS operates, while the other four allow assessment of important features of an agent or of the whole MAS, namely the autonomy, reactivity, rationality and adaptability to the environment. For each of the five goals, questions and metrics are defined to allow assessment to be made of the complexity of the environment, or the MAS feature under study (in this paper the questions are not reported for the sake of brevity). These questions and metrics make it possible to evaluate firstly the agents as single units and then the MAS as a whole. In [9], Russell and Norvig define the environment as the problem that the agent is there to solve. When the problem is complex, the single agent approach may be insufficient, or unable to solve it. In such cases, it may be better or necessary to solve the problem via a multi-agent approach, using a set of agents that interact among themselves or with other system components to find the solution. In these cases, the environment is the complex problem to be solved, and the MAS is the solution. In the real world, complex problems are continually being posed. Although different problems may have a different complexity, it should be noted that even the same problem can be considered at different complexity levels. In this way, problem solving depends not only on the type of problem but also on the choice of the level of complexity at which it needs to be solved. For example, let us consider the problem of teaching English. It could be solved using a MAS that considers the students' knowledge level to be determined entirely by the results of questionnaire tests. Otherwise, the choice may be to use a MAS that takes into account the results of a questionnaire test as the basis for further reasoning which may lead to identification of the students' "true" knowledge of the language. Evaluating a MAS that solves the problem of teaching English independently of the environment would be meaningless. For this reason, the evaluation must relate the values obtained in the assessment of the MAS characteristics (autonomy, reactivity, rationality and adaptability) to the complexity of the environment. One of the problems encountered in defining the metric plan was the different definitions used in literature not only to refer to the environment but also to the internal characteristics of the MAS. To avoid the risk of ambiguity in defining the metric plan, a definition of both the characteristics and evaluation lines considered for each goal is provided, as well as the measurements to be made (or calculated) to estimate them.

3.1 Goal 1: The Environment Complexity

According to Russell and Norvig [12] an environment can be classified on the basis of various lines: its observability, the effect the agent's actions have on the environment, the time, the number of agents, the way the environment is perceived by the agents and the way it evolves. The number and subjectivity of these lines makes it difficult to characterize the environment. It is easy to identify which environment is the most complex, but if it presents other combinations of these properties it will be difficult to define its complexity. Moreover, these properties are not always enough to characterize the environment and the effect the agents' interactions will have on it. In the defined method the complexity of the agent and of the MAS environments is assessed on the basis of three different parameters, namely: **Inaccessibility**, **Instability**, and **Complexity of the Interaction**.

The parameters are measured for each single agent (intra-agent perspective) and for the entire MAS (inter-agent perspective). Thus, the agent environment complexity (metric: ***AgEnvCompl***) is the mean of the values obtained for each parameter measured from the intra-agent perspective, and the MAS environment complexity (metric: ***MASEnvCompl***) is the mean of the AgEnvCompl values measured from the inter-agent perspective.

Inaccessibility. The Inaccessibility parameter expresses the difficulty in gaining complete access at any instant to the resources in its environment. Such resources include the environment components (e.g. web services, DBMS, etc.) or data (e.g. metadata, ontologies, etc.). The more difficult the access to the resources, the more complex the environment. In such circumstances it is necessary to adopt strategies to deal with this inaccessibility. For example, when driving a taxi, environment resources include pedestrians that may suddenly cross the road under the taxi wheels. If the light is poor, the pedestrians are less visible so the taxi driver must have the lights on full beam and pay even closer attention to avoid running them down. In the proposed assessment method, the environment inaccessibility is evaluated using the metrics: ***CompInacc*** and ***ResInacc***, that assess the inaccessibility of the agent environment components and data; and ***AgInacc*** and ***MASInacc***, that represent the inaccessibility of the agent and the MAS environment, respectively.

- *CompInacc* assesses the inaccessibility of the agent environment components (DBMS, other MAS agents with which it interacts, etc.). For each component CompInacc is 1 if the inaccessibility is high, 0.5 if it is medium, 0 if low. The agent overall value is the mean of the measured values.
- *ResInacc* assesses the inaccessibility of the agent environment data (metadata, ontologies, etc). For each type of datum, ResInacc is 1 if the inaccessibility is high, 0.5 if it is medium, 0 if low. The overall value is the mean of the measured values.

In the intra-agent perspective the environment inaccessibility is evaluated using the AgInacc, that is the mean of the previous metrics.

Finally, in order to evaluate the MAS environment inaccessibility (inter-agent perspective), *MASInacc* is used. This is the mean of the *AgInacc* measures for all the MAS agents. The value of *MASInacc* can range between [0-1]. If *MASInacc* $\in$ [0-0.3] then the value is low, if *MASInacc* $\in$ [0.3-0.6] it is medium, and if above, the MAS environment inaccessibility is high.

Instability. The Instability parameter expresses the way the environment evolves, and how fast. In other words, the difficulty in perceiving changes in the environment. The faster and more unpredictably the environment changes, the more complex it is. In such cases, the agent must have mechanisms to perceive these rapid changes. The environment instability is assessed using the metrics: ***Time***, ***Dynam*** and ***NumEffeAct***. ***AgInstab*** and ***MASInstab*** represent the measures from the intra-agent and inter-agent perspectives, respectively.

- *Time* is the time taken to pass from one state to another. This passage can be continuous or intermittent. Clearly, an environment that evolves continually is more complex than one that evolves intermittently at set times. The evaluator identifies the agent environment (components, data, other agents) and sees whether the passage from one state to another occurs continually or intermittently. If it occurs continually, the evaluator will assign a value of 1, otherwise 0. The Time overall value is the mean of measured values for all the environment resources.
- *Dynam* is the speed at which the environment passes from one state to another, in other words the rapidity of change. The passage may be static, in the sense that the environment does not change while the agent is thinking or acting, or dynamic if the environment is changing even while the agent is thinking or acting. If the environment is dynamic it is necessary to keep it under observation while the agent is deciding how to act or it is acting, and also to take account of time. If the environment is dynamic, the value 1 is assigned to each environment component and data, otherwise 0. The Dynam overall value is the mean of the measured values for all the environment resources.

- *NumEffeAct* assesses how unpredictable changes of the environment are as a result of actions taken by the agents. If an agent's actions can have different effects, then the environment will be more unpredictable and so more complex. Let us consider the case in which an agent proposes teaching materials to a student. Of all the material available, the agent chooses to propose some dealing with the solution of second degree equations. The proposal of this material can have different effects on the student's learning depending on whether s/he already has some knowledge of the topic and on other unpredictable factors. For each agent, the evaluator lists the main possible actions and for each action, the possible different effects. If the action can have several effects the action will be scored 1, otherwise 0. The NumEffeAct overall value is the mean of the measured values for all the agent's actions.

From the intra-agent perspective the environment instability is measured by *AgInstab* metric, that is the mean of the previous metrics.

Finally, the MAS environment instability (inter-agent perspective) is measured by *MASInstab*, that is the mean of the *AgInstab* values. The value of *MASInstab* can range between [0-1]. If *MASInstab*$\in$ [0-0.3] then the value is low, if *MASInstab*$\in$ [0.3-0.6] it is medium, and if above, the instability of the MAS environment is high.

Complexity of the Interaction. The Complexity of the Interaction expresses how complex the interactions between agents are in the MAS. The more complex they are, the more needful it is to make predictions and activate coordination mechanisms or competitive strategies. Three metrics are used to assess the complexity of interaction: ***CompGrad***, ***CoopGrad*** and ***Tr&RepMod***. ***AgComplInt*** and ***MASComplInt*** represent the measures from the intra-agent and inter-agent perspectives, respectively.

- *CompGrad* is the degree of competition between the agent and the other MAS agents. The evaluator checks whether the agent competes with another agent to solve the problem; if so, the value 1 is assigned, if not 0. The agent overall value is the mean of the measured values.
- *CoopGrad* is the degree of cooperation between the agent and the other MAS agents. The evaluator checks whether the agent cooperates with another agent or not, and assigns the value of 1 if so, 0 if not. The agent overall value is the mean of the measured values.
- *Tr&RepMod* assesses the need to use trust and reputation models to verify the reliability of the data and behavior of the components in the environment. If it is necessary to use such models the value of 1 is assigned to the metric Tr&RepMod, if not, 0.

From the intra-agent perspective the complexity of interaction is measured by *AgComplInt* metric, that is the mean of the previous metrics.

Finally, the MAS complexity of interaction (inter-agent perspective) is measured by *MASComplInt*, that is the mean of the *AgComplInt* values. The value of *MASComplInt* can range between [0-1]. If *MASComplInt* $\in$ [0-0.3] then the value is low, if *MASComplInt* $\in$ [0.3-0.6] it is medium, and if above, the complexity of interaction is high.

3.2 Goal 2: The Rationality

Russell and Norvig in [12] define the rationality of an agent as its ability to take actions that can maximize its success. This ability varies according to the performance metrics, the perception sequence, the knowledge of the environment and the actions the agent can accomplish. In the defined metric plan the degree of rationality was evaluated according to two parameters: **Mode of Choice of the Actions** and **Maximization of the Success**. The parameters are measured for each single agent (intra-agent perspective) and for the entire MAS (inter-agent perspective).

Thus, the agent rationality degree (metric: ***AgRatio***) is the mean of the values obtained for each parameter measured from the intra-agent perspective, and the MAS rationality degree (metric: ***MASRatio***) is the mean of the AgRatio values measured from the inter-agent perspective.

Mode of Choice of the Actions. This parameter expresses the degree of rationality in choosing the actions to be performed. It is assessed using the metrics: ***AgType, PlaConstr***, ***LearAb*** and ***InsMod***. Then the metrics ***AgModChAct*** and ***MASModChAct*** are calculated.

- *AgType* is the type of agent (simple, stimulus-response, and goal-based agent). Different types have different degrees of rationality. For example, an agent of stimulus-response type shows no rationality because its actions are pre-established by the designer: each sequence of perceptions corresponds to a specific action or series of actions. Instead, a goal-based agent needs to achieve goals and so, at each turn, will choose the actions to be executed to achieve the goal or goals. The evaluator assesses the agent type, and if it is of simple or stimulus-response type the metric AgType will be assigned the value 0, whereas if it is goal-based it will be scored 1.
- *PlaConstr* assesses the agent's ability to build plans of action. If it can do so this is an index of a greater rationality and the value assigned will be 1, 0 if not.
- *LearAb* evaluates the agent's ability to learn. An agent that can learn is considered more rational, then the value of 1 will be assigned, otherwise 0.
- *InsMod* is the agent's possession of an internal model of the actions and intentions of the other MAS agents. An agent that takes into account these factors is more rational, so the metric will be scored 1, otherwise 0.

From the intra-agent perspective the degree of rationality in choosing the actions is measured by the *AgModChAct* metric, that is the mean of the previous metrics.

Finally, the MAS degree of rationality in choosing the actions to be performed (inter-agent perspective) is measured by *MASModChAct*, that is the mean of the *AgModChAct* values. The value of *MASModChAct* can range between [0-1]. If *MASModChAct* $\in$[0-0.3] then the value is low, if *MASModChAct* $\in$[0.3-0.6] it is medium, and if above, the rationality is high.

Maximization of the Success. The Maximization of the Success parameter expresses the ability to maximize the expected result of the actions. It is measured by the metric ***AgMaxSucc*** (from the intra-agent perspective).

- *AgMaxSucc* measures the gap between the expected result of the agent's actions and the result obtained. To calculate this metric, for each agent n intervals of observation lasting t seconds are defined. For each interval, the agent's perception sequence is derived, as well as the knowledge of the environment possessed. The possible agent's actions are defined as a function of the state it is in and the expected results on the environment in the observation interval. These results must be expressed in numerical terms. After establishing the expected result, the actual result on the environment caused by the agent's action is observed. These two outcomes are compared using the following expression: (|val.expect-val.obtain.|)/base where base is a numerical value that can normalize the value on a scale from 0 to 1. It is important to choose an optimal but realistic estimate of the agent's performance as the expected value. The base value depends on the choice of the range of expected and obtained results. After calculating the discrepancy, the value of AgMaxSucc is calculated as the mean discrepancy on the basis of the number of intervals considered. If the obtained result is equal to the expected result, the value of AgMaxSucc is 0, indicating maximum success.

Finally, the MAS ability to maximize the success (inter-agent perspective) is measured by *MASMaxSucc*, that is the mean of the *AgMaxSucc* values. The value of *MASMaxSucc* can range between [0-1]. If *MASMaxSucc* ∈ [0-0.3] then the value is low, if *MASMaxSucc* ∈ [0.3-0.6] it is medium, and if above, the ability level is high.

3.3 Goal 3: The Autonomy

According to Wooldridge [13], autonomy is the property that most strictly characterizes the agent. This refers to its ability to act without the need for human intervention or actions by other agents. In the defined metric plan the degree of autonomy was evaluated according to two parameters: **Proactivity** and **Autonomy in the Organizational Structure**. Like the previous ones, these parameters are measured using both perspectives: intra-agent and inter-agent.

Thus, the agent autonomy value (metric: ***AgAuto***) is the mean of values obtained for each parameter measured from the intra-agent perspective, and the MAS autonomy value (metric: ***MASAuto***) is the mean of the AgAuto values measured from the inter-agent perspective.

Proactivity. A key element of autonomy is proactivity, in other words the ability to “take the initiative” rather than simply acting in response to the environment. Proactivity includes the agents' capacity to exhibit behaviour directed both to satisfying their goals, and to anticipating future situations, making predictions.

The Proactivity parameter is assessed using the metrics ***MoreRol***, ***NegAg***, ***DiaErPrAb*** and ***ComAutAb***. The metrics ***AgProact*** and ***MASProact*** allow the proactivity to be estimated from the intra and inter-agent perspectives, respectively.

- *MoreRol* measures whether the agent can play several roles to solve the problem and whether this passage from one role to another was pre-established by the designer or is decided autonomously by the agent depending on particular factors in the environment. It is assigned value 0 if the agent plays only one role or, although it plays more than one role, the passage from one to the other is pre-established, and the value 1 if it passes autonomously from one to another.

- *NegAg* is the agent's ability to negotiate the assignment of tasks or resources with the other MAS agents; if it can it is more autonomous than one that does not possess this ability. The value 1 is assigned if it can negotiate, 0 if not.
- *DiaErPrAb* is the agent's ability to diagnose errors and/or problems during execution of the tasks; an agent that can diagnose errors and/or problems is more proactive than one that cannot. The evaluator assesses whether the agent has diagnostic powers and if so, assigns the value of 1, 0 if not.
- *ComAutAb* is the agent's ability to undertake and autonomously conduct communication with the other MAS agents. If the agent can do so, it will be assigned the value of 1, if not then 0.

From the intra-agent perspective the proactivity value is measured by the *AgProact* metric, that is the mean of the metrics described above.

Instead, the MAS proactivity value (inter-agent perspective) is measured using the metric *MASProact*, that is the mean of the *AgProact* values. The value of *MASProact* can range between [0-1]. If *MASProact* $\in$ [0-0.3] then the value is low, if *MASProact* $\in$ [0.3-0.6] it is medium, and if above, the proactivity value is high.

Autonomy in the Organizational Structure. The parameter Autonomy in the Organizational Structure expresses the degree of autonomy of action within the MAS organization. To assess this parameter two metrics were used: ***PosStr***, and ***SharTask***. The metrics ***AgAutoOrg*** and ***MASAutoOrg*** represent the autonomy of the agent and of the MAS in the organizational structure, respectively.

- *PosStr* assesses whether the agent occupies a subordinate position in the MAS or not. If it does then it will be less autonomous. For each agent, if it occupies a subordinate position as compared to another MAS agent, the evaluator will assign value 0, if not, then 1. The value of the metric PosStr is the mean of the obtained measures.
- *SharTask* evaluates whether the agent shares tasks with the other MAS agents. If so, its actions will be less autonomous than those of an agent that does not do any sharing. The evaluator lists the main agent’s tasks and for each task the value 0 will be assigned if the agent shares the task with other agents, 1 if not. The value of the metric SharTask is the mean of the assigned values.

From the intra-agent perspective the autonomy in the organization is assessed by *AgAutoOrg*, and is the mean of the previous metrics. The total value of the MAS agents for autonomy in the organization (*MASAutoOrg*) is the mean of the *AgAutoOrg* measures. This value can range between [0-1]. If the value $\in$ [0-0.3] then the autonomy is low, if the value $\in$ [0.3-0.6] it is medium, and if above, the autonomy in the organizational structure is high.

3.4 Goal 4: The Reactivity

Most of the proposals for classifying agents present in the literature [14] consider a reactive agent to be an agent that lacks internal states programmed to make the action to be accomplished according with a perception sequence. In the metric plan, reactivity is considered as the ability to perceive the environment and respond in a timely fashion to changes in it. This quality is assessed by taking into account the **Effectiveness of Acquisition of Perceptions** and the **Rapidity of Response in a Timely Fashion**. Both the parameters are measured for each single agent (intra-agent perspective) and for the entire MAS (inter-agent perspective).

The agent reactivity level (metric: ***AgReact***) is the mean of the values obtained for the assessment parameters. The mean of the *AgReact* values for all the MAS agents is the reactivity value of the entire MAS (metric: ***MASReact***).

Effectiveness of Acquisition of Perceptions. The parameter *Effectiveness of Acquisition of Perceptions* expresses how well the surrounding environment is perceived. This ability is measured with the metric ***AgEffAcqPerc***.

- *AgEffAcqPerc* assesses the agent's ability to use the sensors to perceive the relevant components and data in the environment. To measure this ability, firstly the relevant components and data are identified. For example, to solve a problem where it is important for the agent to perceive a user query, the relevant datum is the query. Then, the agent’s sensors are examined and which environmental components or data are perceived by the sensors is verified. If the agent perceives all the relevant components and data, a value of 1 will be assigned to metric *AgEffAcqPerc*, otherwise 0.

The value of *MASEffAcqPerc* indicates the whole MAS efficacy of perception of its environment, being the sum of the values obtained for the metric *AgEffAcqPerc* divided by the agents making up the MAS. This value ranges between 0 and 1.

Rapidity of Response in a Timely Fashion. This parameter measures how fast each single agent, and the whole MAS, can respond to environmental needs. The metrics defined for assessing this parameter are: ***PercQual***, ***DefBeh***, ***InsMod***, and ***ComMin***; the metrics ***AgRapRespTimFash*** and ***MASRapRespTimFash*** represent the rapidity of the agent and of the overall MAS, respectively.

- *PercQual* measures how well the perceptions are processed, working on the assumption that an agent that can process its perceptions of the environment to a refined degree will take time to do this and will therefore be slower than an agent that does not. The same applies when assessing the MAS as a whole. For each agent, the evaluator identifies its sensors and checks whether they process crude perceptions, for example by choosing the most significant perceptions or aggregating large perception sequences. If this processing occurs the value 0 will be assigned, otherwise 1.
- *DefBeh* ascertains whether the agent's reactions were pre-established by the designer. Such agents have faster reactions to the environment than agents that

need to reason before acting. If the actions are pre-established by the designer, the value 1 is assigned to the metric, 0 otherwise.

- *InsMod* assesses the agent's possession of an internal model of the actions and intentions of the other MAS agents. An agent that takes these into account is slower to react than an agent that does not. If the agent possesses such a model, metric InsMod is given value 0, otherwise 1.
- *ComMin* is the minimization of communication; in other words the agent's ability to carry out tasks or goals with minimal communication with other agents, since this would increase the response times. For this purpose, n intervals of time are defined, each interval lasting time *t* with the number *gr* of goals achieved in interval *t*. Then the mean number of messages exchanged to achieve the goal is calculated. If this value is equal to or less than the previously defined expected value, *ComMin* has a value of 1, otherwise 0.

From the intra-agent perspective, the value for the agent rapidity is calculated using the metric *AgRapRespTimFash*; this is the mean of the values assigned to the three parameters.

The speed of the whole MAS response (MASRapRespTimFash) is the mean of the AgAutoOrg measures. This value ranges from 0 to 1. If the value 0 is assigned it means that the MAS is slow to respond to the environment, whereas 1 shows maximum rapidity of response.

3.5 Goal 5: The Adaptability to the Environment

Since the environmental conditions can change rapidly, the agent (and the entire MAS) must be able to adapt to these changes. This involves being able to modify the plan of actions to be undertaken to achieve the goal and in some cases, also the possibility of changing the short term goal if pursuit of this would lead to failure to achieve the main goal.

The adaptability is evaluated by taking into account the **Ability to Respond to new External Stimuli** and the **Ability to Manage Different Situations**.

These parameters are measured for each single agent (intra-agent perspective) and for the entire MAS (inter-agent perspective). At the end of the assessment both the agent and the MAS adaptability level can be calculated. The first (***AgAdapt***) is the mean of the values obtained for each evaluation parameter from the intra-agent perspective; the second (***MASAdapt***) is the mean of the AgAdapt values measured from the inter-agent perspective.

Ability to Respond to New External Stimuli. The parameter represents the ability to respond to changes of the environment. This capacity has been evaluated using the metrics ***CorrChangReact*** and ***RightRol***; the metrics ***AgAbRespExtStim*** and ***MASAbRespExtStim*** allow calculation of the agent's and MAS ability to respond to changes in the environment, respectively.

- *CorrChangReact* evaluates the correlation between the agent's reactions and the changes in the environment. The evaluator identifies which components belong to the agent's environment. It observes and verifies, during the problem resolution

process, the harmony between the change of the environment and the agent's reactions during the time *t* intervals defined. If there is a high relationship the evaluator assigns the value 1 to the metric CorrChangReact, otherwise 0. For example, in taxi-driving an environment component could be an avalanche sliding down the street. An agent that is able to respond to new external stimuli should change the path of the taxi.
- *RightRol* is the agent's ability to change roles during problem resolution according to changes in the environment. If the agent has this ability the value 1 is assigned to the metric, otherwise 0.

Finally, the agent's ability level (***AgAbRespExtStim***) is the mean of the parameter values; instead, ***MASAbRespExtStim*** calculates the MAS ability to respond to new external stimuli. This value is the mean of the metric AgAbilRespExtStim of all MAS agents and can range from 0 to 1. If the value ∈ [0-0.3] then the ability to respond to changes of the environment is low, if the value ∈ [0.3-0.6] it is medium and if above, this ability is high.

Ability to Manage Different Situations. This parameter measures the ability to cope with different and unpredictable situations. It is evaluated using ***LearAb***, ***EurFinAb***, and ***ExcManAb***. ***AgAbManFiffSit*** and ***MASAbManDiffSit*** measure the agent's and MAS ability to cope with different and unpredictable situations.

- The *LearAb* metric is the same one used to evaluate the agent's rationality. It is alsoused in the adaptability to the environment evaluation because if an agent is able to learn, it can use its experience to manage unusual and unpredictable situations.
- *EurFinAb* calculates the agent's effectiveness in finding suitable heuristics for achieving the goals (for goal-oriented agents) or performing tasks (for non goal-oriented agents). Its value is calculated by comparing the average number of messages sent by the agent to obtain useful information (*vmr*) and the number of messages sent by the agent in the environment expected by the evaluator (*va*). If $vmr > va$ the metric EurFinAb has a value of 0, otherwise it will be 1.
- *ExcManAb* measures the agent's effectiveness in handling exceptions. This effectiveness is calculated by comparing the number of exceptions managed by the code of each single agent (*nem*) and the number of exceptions the agent is expected to manage (*ea*). If $nem \geq ea$ ExcManAb has a value of 1, otherwise 0.

Finally, the agent's capacity to manage different situations (***AgAbManFiffSit***) is the mean of the previous measures, while the MAS capacity (***MASAbManDiffSit***) is the mean of the values assigned for AgAbManFiffSit to all MAS agents. The value of *MASAbManDiffSit* ranges from 0 to 1. If *MASAbManDiffSit* ∈ [0-0.3] then the value is low, if *MASAbManDiffSit* ∈ [0.3-0.6] it is medium and if above, the value is high.

4 Guidelines for Interpreting the Metric Plan Measures

The metric plan illustrated above allows the complexity of the MAS environment and several MAS characteristics to be measured, but does not enable assessment of its adequacy to the environment where it operates. For this purpose, it is necessary to define some guidelines to compare the evaluation of the environment with that of the MAS, and to go into the details of the parameters considered for both the MAS environment evaluation and the entire MAS itself.

Rationality (Goal 2) vs. Environment Complexity (Goal 1). In an environment with a high level of inaccessibility it is difficult to have access to the resources, and so it would be better for the MAS to use planning and learning strategies and to be able to keep an internal state of the environment. In the case of environments with a medium or low inaccessibility level, a medium and low level of rationality when choosing the actions is acceptable.

Moreover, in an environment in which the resources are difficult to access it would be difficult for the MAS to maximize the success. For this reason a medium level of this ability is acceptable. In any case, according to the importance that this ability has in the domain where the MAS will be used, the evaluator can decide if a low level is acceptable. On the contrary, if the environment is characterized by a low level of inaccessibility, it should be high.

In addition, even in an environment with a high level of instability it would be better if the MAS used planning and learning strategies (high value of Mode of Choice of the Actions), because in this way it can face environment evolutions. Nevertheless, if the response time is a critical factor for the environment, a medium level of rationality when choosing the actions is acceptable. Instead, an environment with a low level of instability does not impose constraints as regards this value.

As to the maximization of the success value, it should be high in an unstable environment because this means that the results obtained by the MAS are close to the expected ones. However, since it is difficult to obtain this in an unstable environment, a medium value is acceptable. On the contrary, for a stable environment a high value is expected.

When the value of the interaction complexity is high, a high value of rationality in choosing the actions to be performed will be necessary. Otherwise, both values could be low. Moreover, in a complex environment it would be better to have a high level of ability to maximize the success.

Autonomy (Goal 3) vs. Environment Complexity (Goal 1). In an environment with a high level of inaccessibility it is difficult to have access to the resources, so a medium/high level of proactivity can facilitate access to the resources in order to be able to adopt strategies to deal with the lack of accessibility. For example, if a MAS agent is not able to access a web service due to Internet connection problems, a proactive attitude can allow the necessary information to be found in the cache memory. A medium/high level of proactivity is also necessary in the case of a high level of complexity of interaction, because in an environment in which the agents have to cooperate or compete, proactivity is important. Otherwise, if the environment has a medium or low level of inaccessibility or complexity of interaction, low levels of proactivity are acceptable. Finally, the autonomy of the MAS agents in the organizational structure is not related to the inaccessibility level, nor to the degree of complexity of the interaction, because it depends on the MAS organization structure.

Reactivity (Goal 4) vs. Environment Complexity (Goal 1). The effectiveness of acquisition of perceptions is related to all the complexity parameters of the environment and is one of the necessary conditions for the MAS to be able to react to its environment. For this reason, its value should always be high. Instead, the ability to respond in a timely fashion is not related to the inaccessibility, nor to the interaction complexity, but depends on the instability values. If the instability is high a high rapidity value is necessary.

Adaptability (Goal 5) vs. Environment Complexity (Goal 1). In an environment with a high level of inaccessibility, regardless of the environment instability level unpredictable situations could occur, so the ability of MAS agents to respond to new external stimuli should be high. If the inaccessibility is medium, the ability to respond to new external stimuli can also be medium and so on. The ability to manage different situations is related to both the environment inaccessibility, because this ability supports the agent in gaining complete access to the resources, and the instability, because an increased instability level will mean that an increased number of different situations needs to be managed. For the same reason, the ability to manage different situations and the inaccessibility are also related and so should have the same values.

Moreover, the ability to manage different situations is related to the instability because an increased instability will result in an increased number of different situations to be managed. If the instability is high, a high value for the ability to manage different situations is needed, if it is medium a medium value is sufficient, and so forth. The capacity to respond to external stimuli is not related to the interaction complexity, whereas the ability to manage different situations is.

An environment with a high level of complexity has complex interactions (collaborative or competitive) that do not depend on the behavior of each single agent, but also on the community of agents. For this reason a good capacity to manage different situations is needed. The higher the complexity, the higher this capacity should be.

5 Evaluation of GeCo-Automotive System

GeCo-Automotive MAS has been evaluated from the designer point of view. The evaluation goal was to verify the adequacy of the MAS to the environment it was designed for.

5.1 GeCo-Automotive System

The GeCo-Automotive MAS aims at developing an ICT environment to manage small-medium-sized company knowledge about automotive spare parts. It is an ICT environment prototype integrating functionalities for the analysis and management of the human resources, skills, management of training activities and of documentation. The GeCo-Automotive environment was designed to respond to the knowledge and training requirements of the different professional figures involved in the automotive sector, providing personalized solutions for the work context, skills and tasks of each individual user. The system architecture (shown in Fig. 1) includes two repositories, two components and two static agents: (I) the Learning Object Repository (LOR) named e-TER that manages the Learning Objects [15], their description, and relative publication; (II) the Document Management System (DMS) is a set of tools, software and hardware, that allows management of digital documents (experience or good practices), building and sharing within an organization; (III) the Document Repository (DR) that contains the documents; (IV) the Skill Gap Analysis (SGAS) component that allows the user to self-assess her/his knowledge and to be evaluated by colleagues, on the basis of these evaluations the SGAS component builds the user model; (V) the Learning Management System (LMS) component manages the use of LOs, choosing them according to their content and the user model; (VI) a Classifier Agent that classifies teaching and documentary resources; (VII) a Search Agent that selects the teaching and documentary resources.

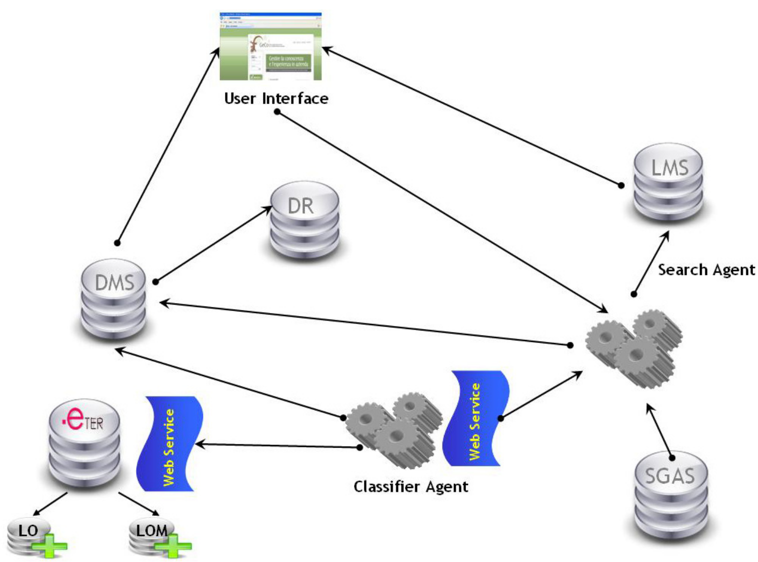

Fig. 1. Geco-Automotive architecture

The sensors of the classifier agent, even if in an embryonic state, are two web services that allow the agent to perceive the resources within repositories, whereas the agent actuator is the web service that sends the research agent the set of classified resources. In the same way, the research agent sensors are web services through which the agent perceives the user model built by the SGAS component, the classification of the resources built by the classifier agent, the association of teaching resources - user competences built by the LMS, and the set of user interface functions that allows the agent to perceive the queries inserted by the user. The agent actuator is the set of interface functions that the agent uses to propose the resources in response to searches by the user.

The MAS acts in the following way. The classifier agent accesses, by web services, e-TER and the DMS, then it catalogs all the resources on the basis of their descriptions and of previously defined taxonomies. In particular, the documents are classified on the basis of the document descriptions produced by the DR. The LOs are classified on the basis of the taxonomy of the resources, defined according to the LOM Educational category [3].

The catalogued resources are available through web services to the search agent that can make semantic searches for the resources. The search agent selects from the set of available resources those that best suit the user's specific needs. To provide this service, it uses the knowledge about the user (stored in the user model built by the SGAS component), the organization of the resources (expressed using decision rules inside the knowledge base of the agent itself), and its perception of the user's query gained by processing the syntax.

5.2 Environment Complexity Evaluation

The classifier agent environment is composed of different components (the Document Management System that manages the documents, the Repository e-TER that contains the LOs, and the search agent) and data (document and LOs metadata and related ontologies). The inaccessibility, measured as the difficulty in gaining complete access to each single component at any moment, is low, whereas the inaccessibility of data, measured in terms of the incompleteness of the metadata, is medium because the information could otherwise be very incomplete, in view of the fact that some metadata are optional. The total inaccessibility value of the classifier agent environment is medium.

The search agent environment is composed of different components and data. The components are the Skill Gap Analysis component, the Learning Management System, the classifier agent, the interface and the Document Management System. The data are the user's domain knowledge, the user's query, the document and LO ontologies. The inaccessibility of the components in this case, too, is low. Instead, the data inaccessibility is medium, because even if the ontologies are completely accessible, the user's query might not be very clear and the user's knowledge might not be available. Moreover, the user's knowledge may vary during the interactions, and would be accessible only if the user does the assessment test. Thus, the inaccessibility of the search agent is medium, as is the inaccessibility of the MAS environment.

The classifier agent environment evolves in a discrete, static way because it does not change during the cataloguing process. Moreover, the main action of this agent, namely cataloguing all the resources on the basis of both their descriptions and previously defined taxonomies, has only one effect on the environment. Thus, the instability environment value is low. The search agent environment also evolves in a discrete way, but it is dynamic because users can continuously modify their knowledge independently of their use of the system. The effects of the agents' actions can be multiple, because, according to the user profile, the search agent selects different types of resources. Overall, the instability of the search agent environment is medium, as is the instability of the MAS environment.

The GeCo-Automotive classifier agent does not compete with the search agent during the resolution process, nor collaborate, because the relationship is limited to sending the resources to the other agent using a web service. Moreover, in the classifier agent environment it is not necessary to use trust and reputation models to verify the reliability of the component behaviors, but it would be necessary to verify the reliability of the metadata describing both LOs and documents. Thus, the value of the interaction complexity of the agent environment is low.

Also, the search agent environment does not compete, nor collaborate, because the agent does not interact with the classifier agent during the search process to find the resource best suited to the user's learning gap. The search agent receives the classified resources from the other agent. In this environment, trust and reputation models are necessary to evaluate the reliability of the information about the user's knowledge. The complexity of interaction of the search agent environment is low.

Overall, the Complexity of the Interaction of the MAS environment is low.

5.3 Evaluation of the GeCo-Automotive Characteristics

Application of the metric plan showed a minimal level of rationality for both the classifier agent and the search agent in choosing the actions. Both agents have simple reflexes, do not build plans of actions to reach their goals, are unable to learn and do not possess an internal model of the actions and intentions of the other agent. The ability to maximize the expected result was measured for the classifier agent as the percentage of correctly classified resources in the time interval considered, and for the search agent as the percentage of proposed resources that satisfy the user's needs. The classifier agent showed a high level of ability to maximize the success, whereas the search agent had a medium value. Overall, the MAS rationality is low.

The MAS agents show the same level of autonomy. They not make diagnoses of errors or problems occurring during the performance of their tasks, nor can they autonomously undertake or maintain any communication. They have fixed roles in the MAS, defined a priori by the designer, and no ability to negotiate. The only values revealing autonomy are those relative to their non subordinate position in the MAS structure and lack of sharing of tasks with other agents. Overall, the MAS autonomy value is low.

The evaluation of the MAS reactivity demonstrated a very high reactivity level. The two agents do not perform any processing of the perceptions, carry out actions defined during the design phase, do not have an internal model of the environment and play a single role in the MAS. In addition, they do not communicate between themselves while carrying out their activities. This results in maximum rapidity of the reactions. The MAS classifier and search agents of the GeCo-Automotive system present poor adaptability to the environment. For both agents the only value showing a degree of adaptability is that of the management of exceptions.

To conclude, the evaluation of the environment complexity points out that the complexity of the classifier agent environment is low, while the complexity of the search agent environment is medium; overall, the MAS environment shows a medium complexity value.

6 Evaluation of the Multi-agent Tourism Recommender System

The goal of the evaluation of the Multi-agent Tourism Recommender System [4] using the proposed metric plan is to measure the adequacy of the MAS for a specific environment assumed by the evaluator. The aim of this evaluation is to show how the proposed method could be applied by an evaluator in order to verify if a MAS is adequate to a specific environment problem. In the following sections, the evaluator makes a set of assumptions both on the environment and the MAS.

6.1 Multi-Agent Tourism Recommender System

The Multi-agent Tourism Recommender system [4] is a knowledge-based system prototype that is aimed at promoting tourism in Argentina. For this reason it suggests the best tourist packages (a package consists of transport, accommodation, cost, activities to do during the holiday, etc.) for Argentinian destinations according to the user's needs and preferences.

The Multi-agent Tourism Recommender architecture is inspired by the different components of a tourism chain. It includes the following components: (I) the Package Repository (PR) that contains the tourist packages; (II) the Destination Ontology (DO) that contains information about the destinations and the resources available in them (geographical coordinates, types of resources, etc.); (III) a set of n Provider Agents (P-Agents) that supply the tourist packages to the T-Agent; (IV) a Travel Assistant Agent (T-Agent) that selects the packages best suited to the user's needs and preferences.

In this paper the prototype system architecture, that includes only two P-Agents, is considered. The agent sensors of a P-Agent are the set of functions and procedures that allows tourist packages to be acquired from external sources. Instead, the actuators are the functions that allow the agent to send the packages (each single package is a message) to the T-Agent. In the same way, the T-Agent sensors are the set of functions that allows the agent to receive the packages, the preferences and restrictions expressed by the user. The actuators, instead, are the set of functions that allows the agent to propose the tourist packages to the users using the interface.

The MAS acts in the following way. The P-Agents supply all the available tourist packages to the T-Agent, that identifies the packages that satisfy user's preferences and restrictions. To do this, the T-Agent accesses both the PR and the DO and, apart from the Travel Assistant role, also has that of Repository Maintenance in the MAS to update its information about the packages (before beginning the recommendation task), and an Interface role (during the recommendation task), to manage the user interface. When the T-Agent requests information, the P-Agents send it all the current packages they can offer. The communication between the agents is message-driven. The T-Agent's overall aim is to maximize the satisfaction of tourists' preferences. Thus, it acquires user preferences and restrictions from the interface, accesses the PR in order to relate the packages to the domain knowledge and infers which packages should be recommended.

6.2 Environment Complexity Evaluation and Prevision

It is assumed that the P-Agents receive the tourist package from the T-Agent and from a set of external sources. The environment data are the tourist packages, that are always accessible; the evaluator should assume, however, that the P-Agents environment components are not completely accessible to the agents. Instead, the T-Agent components (P-Agents, Destination Ontology, package repositories) are completely accessible but the user's preferences are not; in this case the inaccessibility value of T-Agent environment is medium, as is the MAS environment value.

If the packages are supplied by the provider agents at discrete time intervals, these agents' environment is discrete and, assuming that new packages cannot be available to the P-Agents while they supply the packages to the T-Agent, the environment is static. The only actions that the P-Agents can perform are those of acquiring the packages from the sources and supplying them to the T-Agent. For these reasons the P-Agents environment has a low level of instability. Moreover, the evaluator assumes that the T-Agent environment also evolves in a discrete and static way, because it is presumed that the user's preferences will not change the user's request during the agent process. The T-Agent actions have only one effect on the environment: to recommend the tourist packages. The T-Agent environment instability value is also low. Thus, the instability level of the entire MAS is low.

As regards the interaction complexity, the evaluator assumes that the P-Agents do not compete nor collaborate during the recommendation process, and they do not compete nor collaborate with the T-Agent. In this case, the value for competition or collaboration is null. In addition, if the sources of P-Agent packages in the considered environment were unreliable, it would be necessary to use trust and reputation models to verify the reliability of the sources. Assuming that the components and data in the T-Agent are reliable, it is possible to assign a low level to the interaction complexity of both P-Agents and the T-Agent. Thus, the Complexity of the Interaction of the entire MAS environment is low.

6.3 Evaluation of the Multi-Agent Recommender System Characteristics

The application of the metric plan to the Multi-Agent Recommender System shows that the P-Agents have a low level of rationality when choosing the actions in the environment assumed by the evaluator. Even if the P-agents have a high level of ability to maximize the success of the process they have simple reflexes, do not build plans of actions to reach their goals, are unable to learn and do not have an internal model of the actions and intentions of the T-agent. The T-Agent, instead, shows a very high grade of rationality in choosing the actions to be performed. It is a graded BDI agent and therefore an intentional agent, and it is able to build plans of actions. Overall, the MAS rationality value is medium.

Since there is no evidence that the two P-Agents are able to make a diagnosis of errors or problems occurring during the performance of their task, the evaluator assumes that they do not have this ability. They have fixed roles in the MAS, defined a priori by the designer, no ability to negotiate, but they can send messages at any moment. Therefore, their proactivity level is low. Moreover, each P-Agent ignores the other P-Agents, and has a subordinate position in the MAS structure with respect to the T-Agent and does not share its tasks with others. The level of autonomy in the organizational structure of the MAS is high. The P-Agents autonomy value in the MAS is medium.

For the T-Agent, too, the evaluator assumes that it cannot make a diagnosis of errors or problems occurring during the performance of its tasks. Like the P-Agents, it can autonomously engage in communication and has several roles in the MAS; its main role is to provide tourists with recommendations about Argentinian packages, but it also has a repository maintenance role. Moreover, it has not the ability to negotiate. For these reasons, the T-Agent proactivity level is medium. The T-Agent does not have a subordinate position in the MAS and it does not share its tasks, therefore its autonomy value in the MAS structure is high. The T-Agent shows a medium level of autonomy. Overall, the autonomy level of the MAS is medium.

Since the only task of the P-Agents is to provide the packages, it can be assumed they are able to perceive all the relevant components and data. In addition, it is assumed that the two P-Agents do not process the perceptions, i.e. the tourist packages. They carry out actions defined during the design phase, do not have an internal model of the actions and intentions of the other MAS agents, nor communicate between themselves while carrying out their activities; they only communicate with the T-Agent aiming to provide the tourist packages. Therefore, the P-Agents are very good at responding in a timely fashion. Overall, the P-Agents show a high reactivity level. The T-Agent, too, is able to perceive all the relevant components and data environment, but it is less rapid in responding to the environment than the P-Agents. Even if the T-Agent has minimal communication with the P-agents when asking for the packages, it processes the received messages (the packages) using a set of actions that are not predefined by the designer. It is an intentional agent, since it is able to build different plans of actions according to the messages received. Therefore, the agent shows a low value for rapidity to respond in a timely fashion. Overall, the T-Agents have a low reactivity level. For these reasons, the reactivity of the entire MAS is medium.

In [4] there is evidence that the P-Agents are able to respond to the packages requests of the T-Agent and they cannot change roles during the packages recommendation process. In this context, if there were changes of the environment, for example, and the T-Agent cannot receive the packages because it has not finished the internal deductions process, the P-Agents would send the messages anyway.

Thus, the P-Agents have a low level of ability to respond to new external stimuli. The P-Agents ability to manage different situations is also low because they do not learn, and are not able to find heuristics for performing the tasks. The adaptability of the P-Agents to their environment is low. The T-agent, instead, shows a high level of ability to respond to new external stimuli, because it is able to consider new user preferences and to build suitable plans of actions; it shows a medium ability level to manage different situations, because it is able to find heuristics for achieving its goals and to handle exceptions, but it is unable to learn. Overall, the adaptability level of the T-Agent to its environment is medium.

On the whole, the value of adaptability to the environment of the entire MAS is medium.

7 Conclusions

The paper proposes a method for evaluating MAS that, unlike other evaluation approaches presented in the literature, uses high level metrics that highlight characteristics like autonomy, reactivity, environment adaptability, thus allowing the agents to be distinguished from the objects (of the O.O. paradigm). Moreover, it merges the inter-agent and intra-agent characteristics evaluations, supplying a ready-to-use GQM.

The defined metric plan has numerous applications. It can be used by a MAS designer as a guideline during the building process or by an evaluator who wishes to compare different MAS in order to choose the one best suited to solve a specific problem. Considering the high level of abstraction of the approach, only the metrics need to be contextualized to the specific MAS to be evaluated. This flexibility is possible because the MAS evaluation is related to the environment in which it operates. Thus, during the analysis phase the defined metric plan supports the designer’s definition of the problem that the MAS should solve, helping to define all the abilities that each single MAS agent should have in order to be able to cope with the problem. This is possible because the metric plan allows comparisons between the agents’ capacities and the environment complexity. During the test phase it supports the evaluator aiming to understand whether the agent and the MAS have all the characteristics necessary to deal with the problem to be solved. Moreover, the defined metric plan helps the evaluator to find out which are the key characteristics (or desired qualities) to be considered during the evaluation of different MASs.

In order to validate the method’s independence of the problem domain and to investigate the efficacy of its application according to the different evaluation goals, two applications of the defined metric plan have been described in this paper. The MASs considered are: GeCo-Automotive and the Multi-agent Tourist Recommender. The first is aimed at developing an ICT environment to manage small-medium-sized company knowledge about automotive spare parts, suggesting learning activities and best practices to employees of companies in the automotive sector; the second is aimed at promoting tourism in Argentina, suggesting the best tourist packages for Argentinian destinations according to the tourist's needs and preferences. The goal of the application of the method to GeCo-Automotive system has been to verify the MAS's adequacy to the environment complexity for which it was designed. In fact, the authors are the MAS designers. This evaluation allowed us to make some observations about both the agents' and MAS's suitability to the environment complexity. In particular, application of the method to the GeCo-Automotive MAS and its environment allowed us both to observe specific weaknesses of the single agents in terms of poor rationality, autonomy, reactivity and adaptability to the environment, and to assess whether the evaluation parameter values are suitable for the MAS environment complexity. The evaluation of the MAS using the metric plan highlighted the fact that the considered environment has a medium level of complexity. The rationality level of the MAS is appropriate because it depends on its ability to maximize the success of the process. The agents' autonomy, instead, is not adequate to the environment because it is related to each agent's independence during the execution of its tasks and does not depend on its ability to diagnose errors or problems. The reactivity level and the environment adaptability are acceptable. The estimated value of the adaptability to the environment is higher than the expected value, which is an important point in view of the need to face rapid evolutions of the environment. Thus, the analyzed MAS is adequate to its environment, even if the agents' autonomy level needs to be improved by increasing their proactivity.

Instead, the goal of the Multi-agent Tourism Recommender System evaluation was to show that the metric plan can be used to verify a MAS's adequacy to a specific environment assumed by an evaluator. In this case, the metrics highlighted the fact that the MAS is adequate to the considered environment, even if an improvement of the environment adaptability ability of the P-Agent would be desirable.

In fact, the rationality level of the MAS is medium but adequate to the environment complexity, that has been assumed as low. This value of rationality is due mainly to the T-Agent's capacity to choose and build plans of action during the recommendation process. The autonomy and reactivity values are sufficient for the defined environment, but the adaptability is not. When going into the details of the agent evaluations, it was noted that the P-Agents are not able to respond to new external stimuli and to manage different situations. Both capacities are useful in the given environment, where the providers acquire the packages from different sources and supply them to the T-Agent. Moreover, those capacities would be useful to face unpredictable situations of source inaccessibility and they would be helpful as a means of designing a synchronous Exchange of packages among the P-Agents and T-agent. Therefore, the results of this evaluation also provided a useful basis for reflections on further developments of multi-agent recommender systems working in the environments studied.

References

1. Basili, V., Caldiera, G., Rombach, H.: The Goal Question Metric Approach. In: Marciniak, J.J. (ed.) Encyclopedia of Soft. Eng., vol. 2, pp. 528–532. John Wiley & Sons, Inc. (1994)
2. Di Bitonto, P., Laterza, M., Roselli, T., Rossano, V.: An Evaluation Method for Multi-Agent Systems. In: Jędrzejowicz, P., Nguyen, N.T., Howlet, R.J., Jain, L.C. (eds.) KES-AMSTA 2010. LNCS, vol. 6070, pp. 32–41. Springer, Heidelberg (2010)

3. Di Bitonto, P., Plantamura, V.L., Roselli, T., Rossano, V.: A taxonomy for cataloging LOs using IEEE educational metadata. In: 7th IEEE International Conference on Advanced Learning Technologies, pp. 139–141. IEEE Press, Los Alamitos (2007)
4. Casali, A., Von Furth, A., Godo, L., Sierra, C.: A Tourism Recommender Agent: From theory to practice. J. of Inteligencia Artificial 12, 23–38 (2008)
5. Chidamber, S.R., Kemerer, C.F.: A Metrics Suite for Object Oriented Design. J. IEEE Transactions on Software Engineering 20, 476–493 (1994)
6. Jang, K.S., Nam, T.E., Wadhwa, B.: On measurement of Objects and Agents, `http://www.comp.nus.edu.sg/~bimlesh/ametrics/index.htm`
7. Klügl, F.: Measuring Complexity of Multi-agent Simulations – An Attempt Using Metrics. In: Dastani, M.M., El Fallah Seghrouchni, A., Leite, J., Torroni, P. (eds.) LADS 2007. LNCS (LNAI), vol. 5118, pp. 123–138. Springer, Heidelberg (2008)
8. Król, D., Zelmozer, M.: Structural Performance Evaluation of Multi-Agent Systems. J. of Universal Computer Science 14, 1154–1178 (2008)
9. Gutiérez, C., García-Magariño, I., Gómez-Sanz, J.J.: Evaluation of Multi-Agent System Communication in INGENIAS. In: Cabestany, J., Sandoval, F., Prieto, A., Corchado, J.M. (eds.) IWANN 2009. LNCS, vol. 5517, pp. 619–626. Springer, Heidelberg (2009)
10. Ben Hmida, F., Lejouad Chaari, W., Tagina, M.: Performance Evaluation of Multiagent Systems: Communication Criterion. In: Nguyen, N.T., Jo, G.-S., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2008. LNCS (LNAI), vol. 4953, pp. 773–782. Springer, Heidelberg (2008)
11. Lass, R.N., Sultanik, E.A., Regli, W.C.: Metrics for Multi-agent Systems. In: Madhavan, R., Tunstel, E., Messina, E. (eds.) Performance Evaluation and Benchmarking of Intelligent Systems. LNCS, pp. 1–19. Springer, US (2009)
12. Russell, S., Norvig, P.: Artificial intelligence: A modern approach. Prentice-Hall, Englewood Cliffs (1995)
13. Wooldridge, M., Jennings, N.R.: Intelligent agents: Theory and practice. The Knowledge Engineering Review 10, 115–152 (1995)
14. Nwana, H.S.: Software Agents: An Overview. The Knowledge Engineering Review 11, 205–244 (1996)
15. Wiley, D.A.: Connecting learning objects to instructional design theory: A definition, metaphor and taxonomy. J. of Instructional Use of Learning Objects 2830, 1–35 (2000)