Validación del proceso basado en M++ de las Trazas de Razonamiento Introspectivas de la Función Cognitiva Percepción de la Arquitectura Metacognitiva CARINA

Validation Process based in M++ for Introspective Reasoning Trace of the Cognitive Function Perception in Metacognitive Architecture CARINA

María Alejandra Flórez Madrigal[1], Adán Alberto Gómez Salgado[2], Manuel Fernando Caro Piñeres[3]

[1] *Estudiante, Informática Educativa, Informática Cognitiva y Computación Cognitiva, Universidad de Córdoba, Montería, Colombia.*
mflorezmadrigal@correo.unicordoba.edu.co

[2] *Docente, Informática Educativa, Informática Cognitiva y Computación Cognitiva, Universidad de Córdoba, Montería, Colombia.*
aagomez@correo.unicordoba.edu.co

[3] *Docente, Informática Educativa, Informática Cognitiva y Computación Cognitiva, Universidad de Córdoba, Montería, Colombia.*
manuelcaro@correo.unicordoba.edu.co

Recibido: 15/ago/2018 – Revisado: 30/sep/2018
Aceptado: 30/oct/2018 – Publicado: 30/dic/2018

Resumen Este artículo presenta la representación basada en M ++ de la traza de razonamiento introspectivo en la arquitectura metacognitiva CARINA. Las trazas de razonamiento son una estructura de conocimiento declarativo que captura los estados mentales y la secuencia de toma de decisiones en el ciclo de acción-percepción de un agente cognitivo. M ++ es un DSVL para modelar la metacognición en sistemas inteligentes e incorpora dos mecanismos de meta-razonamiento, es decir, monitoreo introspectivo y control de meta-nivel. La validación de M ++ se realizó en dos dimensiones: utilidad potencial y utilidad. Para el proceso de validación, se utilizaron los siguientes métodos: estudio empírico basado en la percepción del usuario. Con respecto a la relación de los comportamientos representados en M ++, el 85.7% de los expertos consideró que la representación es adecuada en comparación con el 14.3% que no la consideró.

Palabras claves: informática cognitiva; trazas de razonamiento; función cognitiva; CARINA; monitoreo introspectivo; arquitectura cognitiva; M ++.

Abstract this paper presents the representation based in M++ of introspective reasoning trace in cognitive architecture CARINA. Reasoning trace is a declarative knowledge structure that captures the mental states and decision-making sequence in the action-perception cycle of a cognitive agent. M++ is a DSVL for modeling metacognition in intelligent systems and incorporates two meta-reasoning mechanisms, i.e., introspective monitoring and meta-level control. M++ validation was performed on two dimensions: potential usefulness and usability. For the validation process, the following methods were used: empirical study based on user perception. Regarding the relation of the behaviors represented in M ++, 85.7% of the experts considered that the representation is adequate compared to 14.3% that did not consider it.

Keywords: cognitive informatics; reasoning trace; cognitive function; CARINA; introspective monitoring; cognitive architecture; M++.

1 Introduction

Cognitive agents are entities or pieces of software that perceive some stimuli from the environment and behave rationally to achieve its goals by selecting some action from its set of competencies (M. Cox & Raja, 2007), (Baldoni, Baresi, & Dastani, 2015). A cognitive agent with metacognitive abilities is composed at least by two cognitive levels named object-level and meta-level (Caro, Gómez, & Giraldo, 2017). The object-level contains the model that an artificial intelligent agent has for reasoning about the world (i.e. agent's environment) to solve problems (Caro, Josyula, Cox, & Jiménez, 2014). The meta-level is a level of representation of the reasoning of an artificial intelligent agent Introspective monitoring is a metacognitive mechanism, which is done through information feedback that is gathered at the meta-level from the object level (M. T. Cox, 1997) (Sun, Zhang, & Mathews, 2005a). Metareasoning consists of both the meta-level control of computational activities and the introspective monitoring of reasoning (M. Cox & Raja, 2007).

Introspective monitoring is necessary to gather sufficient information with which to make effective meta-level control decisions (Caro *et al,* 2017). Monitoring involve generating explanations for object-level choices and their effect on ground level performance (M. T. Cox & Raja, 2008). When reasoning fails at some task, monitoring may involve the explanation of the causal contributions of failure and the diagnosis of the object-level reasoning process (M. Cox & Raja, 2007).

For a cognitive agent performs a meta-level reasoning it needs a declarative representation and an explicit trace of the execution of the performance system where a trace of what happened is described helping explain why it happened (M. T. Cox & Ram, 1999). A trace is a structure that consists of information about a computation, where is described how the computation of a program obtained its outputs from its input (Chitil & Luo, 2007). According to Sun, Zhang, and Mathews (M. T. Cox & Ram, 1999), a reasoning trace is a structured set of mental operations that describes and produces changes in mental states, selects problem operators, and eventually results in the solution plan of a cognitive agent.

In a cognitive agent that reasons in its Object Level or in its Meta Level, it is necessary to store the reasoning trace in order to analyze it and look for reasoning failures or improve the system. From this, the objective of this work is to validate these traces of the reasoning cycle of a cognitive agent using a Domain-Specific Visual Language (DSVL) called M++. M++ is a language for modeling metacognition in Intelligent System (Caro, Josyula, Jiménez, Kennedy, & Cox, 2015). In M++, the abstract syntax is specified with MOF-based metamodels and the concrete syntax is expressed by some mapping of the abstract syntax elements to visual constructs. The main artifacts of M++ are models specified in a visual manner. The motivation of this research is the simulation of processes that commonly occur in the natural intelligence using the Metacognitive Architecture CARINA. Taking into account that in Latin America exist initiatives in the development of intelligent systems based on cognitive and metacognitive processes (Gómez, Caro, Solano, & Vega, 2018), this work extend the frontier of knowledge specifically in this area of study.

This paper is structured as follows: The second chapter shows the general characteristics and elements of the cognitive architecture CARINA. The next chapter shows the reasoning traces in M++ to represent behaviors of Perception cognitive function and the last chapter exposes the conclusions of this paper.

2 Cognitive Architecture CARINA

CARINA is a meta-cognitive architecture for artificial intelligent agents. CARINA is derived from the MISM Metacognitive Metamodel (Caro, Josvula, Gomez, & Kennedy, 2018). CARINA integrates self-regulation and metamemory with support for the metacognitive mechanisms of introspective monitoring and meta-level control; in this sense, CARINA assumes a functional approach to philosophy of mind, according to Fodor (Fodor, 1975), Piccinini (Piccinini, 2010), Scheutz (Scheutz, n.d.).

CARINA is composed of two cognitive levels named object-level and meta-level. The object-level contains the model that an artificial intelligent agent has for reasoning about the world (i.e. agent's environment) to solve problems (Caro *et al,* 2014).

Object-level behavior consists of cognitive functions such as problem solving or memory retrieval (M. T. Cox, n.d.). In CARINA, object-level has stages which are sets of cognitive functions (CF), among these functions Perception Function is found, as shown in Fig. 1. The meta-level contains a dynamic model of the object-level (M. T. Cox, n.d.).

The meta-level includes the components, knowledge and mechanisms necessary for a system to monitor and control its own learning and reasoning processes. The Metalevel of CARINA has two types of metacognition: Self-regulation and Metamemory.

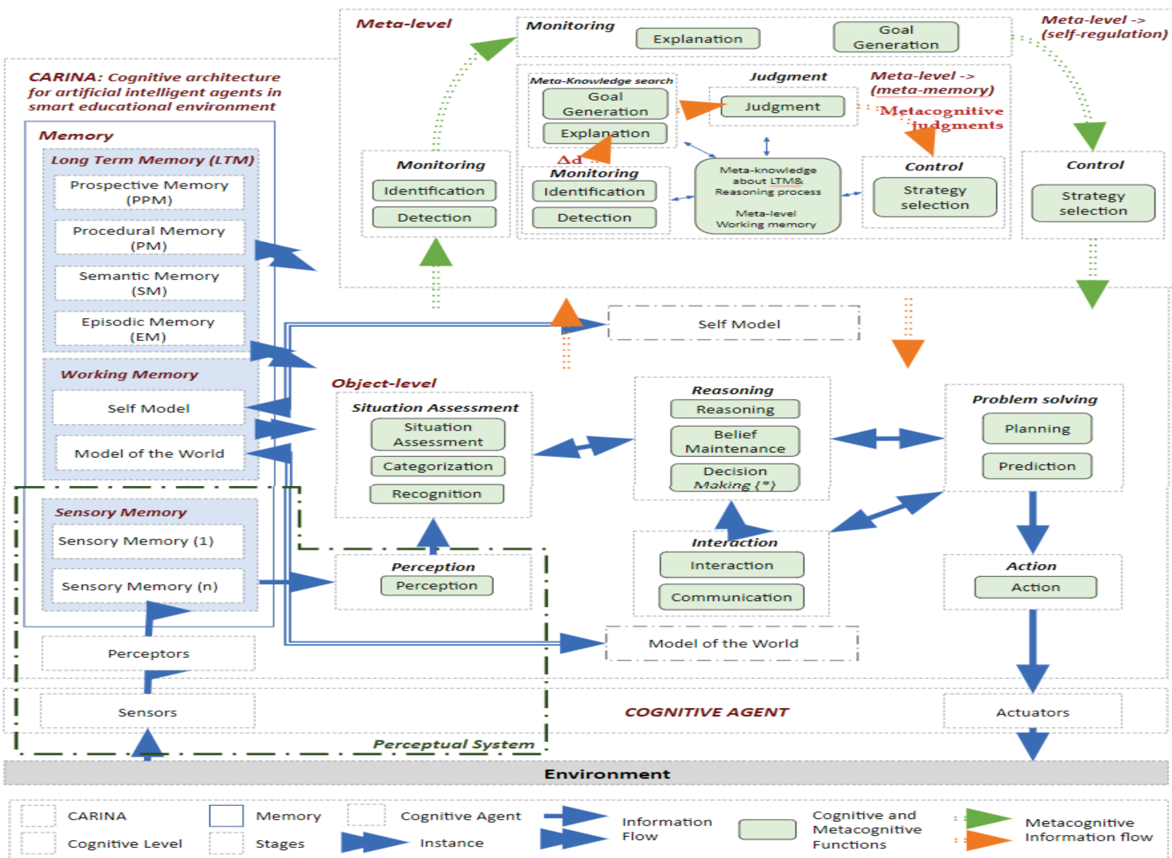

Fig 1. Structural view of CARINA

A cognitive loop (action-perception loop) in CARINA starts when the agent perceives changes to the environment resulting from actions. The situation assessment stage takes as input the perception and processes it. The processing of the information perceived includes the recognition of situations or events as instances of known or familiar patterns and the categorization of objects, situations, and events into known concepts or categories. The output of the situation assessment stage is the combination of perceptual information about many objects and events to compose a comprehensive model of the current environment (i.e. a model of the world). Introspective monitoring and meta-level control are two meta-reasoning mechanisms implemented at the meta-level in CARINA (Caro et al., 2017). Introspective monitoring includes mechanisms for detecting reasoning failures at the object-level. The main purpose of monitoring is to provide enough information to make effective decisions in the meta-level control. The monitoring process is done through information feedback that is gathered at the meta-level from the object level Fig. 1. Thus, each cognitive task executed in at the object level has a performance profile that is continuously updated in the meta-level. The performance profile is used to evaluate the results of each reasoning task (Caro *et al,* 2017). According to Caro, Gómez, and Giraldo, (2017) in CARINA, an Algorithmic Knowledge Profile is a profile that holds the local state of a cognitive function in the form of algorithmic local state and is defined as a profile (P) of a cognitive function (χ) that consists of a data set λ, a set of algorithms α, and a feeling of confidence v, as shown in:

$$P\chi = < \lambda, \alpha, v > \quad (1)$$

Where λ represents the local state of the system with respect to a cognitive function of object-level. Data set λ consists of the set of values related to the processing and performance of the cognitive function (χ). λ = {ID, B, E, S, C, IP, OP}, with:

ID is the identifier of the cognitive function.

B is the time stamp of when the cognitive function was started.

E is the time stamp of when the cognitive function is finished.

S is the state of the cognitive function, s S and S = {active, inactive}

C is the priority level for focus attention c C and C = {low, medium, high}.

IP is the set of parameters used as input of the cognitive function.

OP is the output of the cognitive function.

However, it is necessary to add other components to this data set (λ) to achieve the objective of this work, such as:

σ_{pre} is the necessary precondition mental state in order to execute the cognitive function.

σ_{post} is the postcondition mental state generated after cognitive function is executed.

γ is the goal of cognitive function.

$\zeta\gamma$ is the set of subgoals that belongs to the main goal of cognitive function.

$\sum\rho$ is a set of rules that were fired to achieve the cognitive function main goal

$\sigma\delta$ is a sequence of states that has been modified by subgoals which belongs to cognitive function.

Caro, Gómez, and Giraldo, (Caro *et al*, 2017) Affirm that the set of algorithms α represents the behavior of a cognitive function, having as input a local state at an instant of time. Given a local state $<\lambda_i, \alpha_i, v_i>$ in P_χ, the following possibilities can be found: i) $\alpha_i = \{\}$, ii) $\alpha_i = \{a_1\}$, iii) $\alpha_i = \{a_1, \dots, a_n\}$, with a_i is an algorithm. This point of view needs to be expanded, conceiving α not as a set of algorithms but as a Cognitive Tasks Space ($\Gamma\sum$), where the reasoner traces paths in this space, ordering cognitive tasks in a specific sequence according to goal that cognitive function wants to achieve:

$$\Gamma\sum = \alpha_i = \{a_1, \dots, a_n\} \text{ where } a_i \text{ is a cognitive task}$$

Each cognitive function triggers a feeling of confidence v associated with the correctness of the decision of use α having as input λ, with v ϵ N and N = {low, medium, high}. The meta-level keeps an updated model of the object-level called the "self-model" (M. T. Cox & Raja, n.d.), (Caro *et al*, 2014), (Sun, Zhang, & Mathews, 2005b). An Intelligent System with metacognitive abilities makes explicit its components, capabilities, actions, precepts, and internal state information in its self-model (Madera-Doval & Cardozo-Soto, 2018). According to Caro, Gómez, and Giraldo, (Caro *et al,* 2017) this self-model is based on an internal representation of the reasoning processes that occur at the object-level. This same author affirms that the Self-model (Sm) consists of the set of elements that store information about the reasoning process at the object-level.

Sm = {D, T, M, J}, with:

D is the set of computational data generated by cognitive and metacognitive task.

T is the set of Reasoning Traces generated by reasoning and metacognitive tasks. T is the topic which in this research is referred.

M is the set of performance profiles used to evaluate the results of each cognitive function or strategy.

Metacognitive judgments (J) represent assessments performed at the meta-level about events that occur in object-level. Metacognitive judgments are triggered when the knowledge is acquired (M. T. Cox & Ram, 1999), in our approach this is referred when the algorithmic knowledge profile is updated (Caro *et al,* 2017). Reasoning Traces belong to Self-Model structure as observed. Next chapter explains the formal representation of this Reasoning Traces of cognitive function Perception in CARINA. Before describing how this Reasoning Traces are represented, it is necessary to observe in detail the internal behaviors of cognitive function Perception in CARINA. These internal behaviors (Fig. 2). For this research we have represented these behaviors of the cognitive function perception (goals, mental states and actions) through of M++.

A goal is an objective the system under consideration should achieve.

Mental states are variables Booleans that can be true or false. (Caro *et al,* 2018)

Action is a class of events; viewed intuitively, those that result from the activity of some agent or agents in accomplishing some goal (Georgeff, 1984).

In figure 2 each goal, mental State, action, pre-conditional mental State and post-conditional mental State that belongs to Perception Cognitive Function in CARINA is shown. For example, Goal $\gamma_{\rho 101}$ named detected stimuli point to new stimuli is detected which is Mental State σ_{101}. This Mental State is modified by Action α_{101} named Read Stimuli, which in turn has a Precondition σ_{pre} named pre_read_stimuli. The behaviors of Perception Cognitive Function in CARINA are: *γρ105***:**

copy_input_fact_BCPU; *γρ104***:**
save_input_fact_into_ssm; γρ103:
copySMUtoBCPUinput(aSMU); γρ102:
encode_input_fact; γρ101:
read_stimuli; γρ001:
perceive_stimuli_from_environment; σ101:
new_stimuli_is_detected; σ102:
input_is_read_from_sensor; σ103:
input_fact_is_encoded; σ104:
input_fact_is_saved_into_ssm; σ105:
input_fact_is_copied_into_BCPU_input; σρ001:
input_is_perceived; α101:
read_Stimuli (s); α103:
encode_SMU(inputData); α104:
save_SMU_to_SSM(aSMU); α105:
copy_SMU_to_BCPU_input(aSMU).

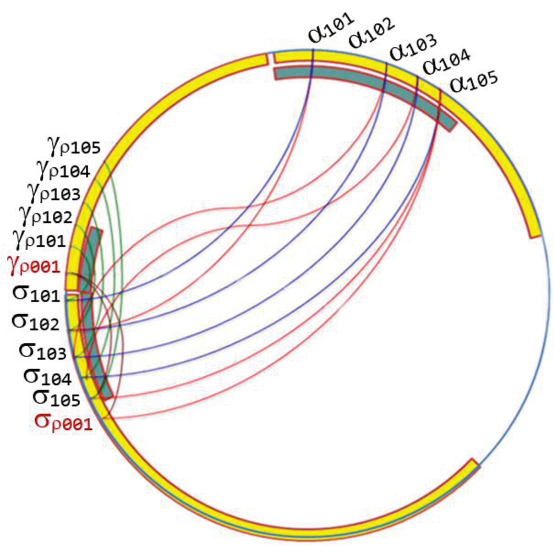

Fig 2. Behaviors of Perception Cognitive Function in CARINA

3 Formal Representation of Reasoning Traces of CARINA Perception Cognitive Function

A reasoning trace is a declarative knowledge structure that captures the mental states and decision-making sequence in the action-perception cycle of a cognitive agent. A reasoning traces in CARINA perception cognitive function is composed of three elements: goals ( ), mental states ( ) and actions (α). With a goal that points to a mental state and this mental state is associated with an action, a reasoning trace is a logical sequence of goals, and mental state is associated with an action. Formally a Reasoning Trace (ρT) of CARINA's Perception Cognitive Function is a 5-tuple, i.e.:

$$\rho T \triangleq < \gamma, \sigma, \alpha, \sigma_{pre}, \sigma_{post} > \quad (2)$$

Where:

ρT is a reasoning trace of cognitive function Perception in CARINA. is a set of goals that belongs to CARINA Perception Cognitive Function.

 is a set of mental states that belongs to CARINA Perception Cognitive Function.

α is a set of actions that belongs to CARINA Perception Cognitive Function.

σ_{pre} is a set of preconditional mental states that belongs to CARINA's Perception Cognitive Function.

σ_{post} is a set of postconditional mental states that belongs to CARINA's Perception Cognitive Function (Florez, Gomez, & Caro, 2018).

4 Representation of Reasoning Trace in M++

M++ is a DSVL for modeling metacognition in intelligent systems and incorporates two meta-reasoning mechanisms, i.e., introspective monitoring and meta-level control. In M++, the abstract syntax is specified with MOF-based metamodels and the concrete syntax is expressed by some mapping of the abstract syntax elements to visual constructs (Caro *et al,* 2018). The main artifacts of M++ are models specified in a visual manner. In Fig. 3, section (A) shows the icons used to represent object-level tasks and section (B) displays icons representing elements that interact with the tasks at object-level.

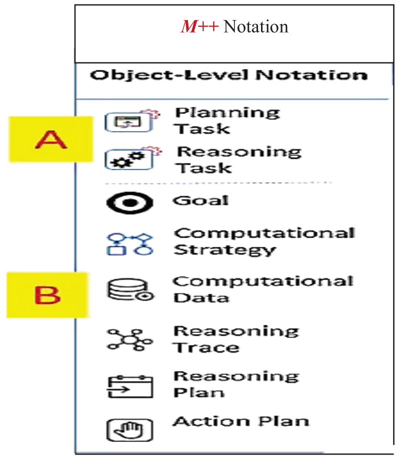

Fig 3. Main elements in M++ notation.

CARINA represents the problems that intend to solve through Mental States. A mental state is a representation that can build a plan for executing tasks in order to accomplish a goal. The mental state responds to events from the environment and infers something (Isern, Gómez-Alonso, & Moreno, n.d.). These Mental States are stored in its working memory structure called "model of the world". To achieve these Mental States CARINA generates a series of Goals stored in its motivational system. Goals are objectives that drive a task or process+ *xx*. These Goals point towards Mental States of working memory in order to modify them through a plan composed by actions located in its procedural memory. An action is a class of events; viewed intuitively, those that result from the activity of some agent or agents in accomplishing some goal (including the achievement of desired conditions, the maintenance of desired invariants, the prevention of other events) (Georgeff, 1984) . Below, a model based in M++ of the behaviors of Perception Cognitive Function is presented. Goals, mental states, actions as well as pre-conditional and post-conditional mental states that belong to this cognitive function are detailed. (Fig. 4)

The model of the world in CARINA is represented through mental states in its working memory, defining the current situation of problem and the ideal situation where the problem has been already solved. In this cognitive model, six mental states are specified, where "stimuli_is_perceived" mental state is the central mental state of the cognitive model. This central mental state is modified just if the others mental states are revised in the following order:

stimuli_is_detected;
stimuli_is_read;
SMU_is_encoded;
Input_fact_is_saved_into_SSM;
SMU_is_copied_into_BCPU.

Each goals that composes the cognitive model point to a mental state having a current state and a target state. These goals trigger actions that need preconditions to modify the mental state which points and generates effects that allow to continue with the execution of next action until the central mental state of cognitive model is modified.

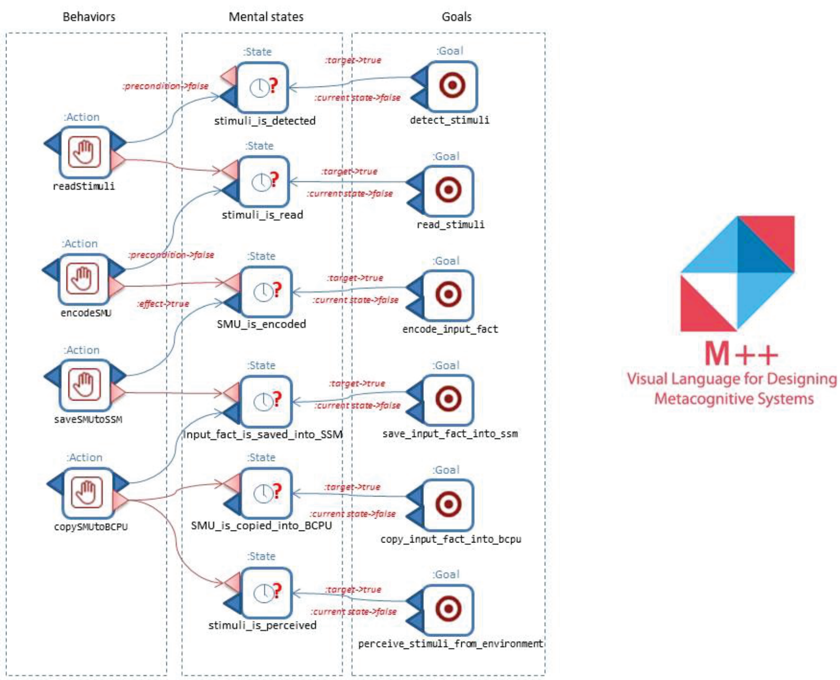

Fig 4. Representation of mental states, actions and goals in M++

5 Validation of the Model M++

M++ validation was performed on two dimensions: potential usefulness and usability. For the validation process, the following methods were used: Empirical study based on user perception. In empirical study the user perception about the quality of the M++ notation was measured [53]. A practical experiment was used to verify the potential utility and usability of M++ based cognitive model. The experimental study was developed based on the designed parameters of the software engineering experiments described in the works of (Molina, Gallardo, Redondo, Ortega, & Giraldo, 2013), ("Ethics in Research and Experimentation," n.d.), (Sjøberg et al., n.d.).

The objective of the experiment was to evaluate the M++ notation with respect to the readability and usefulness of the M++ based cognitive model. The variables used to measuring the user perception with regard to the quality of the notation are based on (Sjøberg et al., n.d.), ("Ethics in Research and Experimentation," n.d.). (i) Perceived ease of read: This variable represents a perceptual judgment of the effort required to read M++ based cognitive models; (ii) Perceived usefulness: This variable expresses the degree to which a person believes that the use of

M++ will achieve its intended objectives regarding the appropriate representation of goals, mental states and actions in cognitive model based in M++. This table describes the codes used in figure 2

Table 1. Reading Perception

Graphical specification	Professionals
Usability of M ++ to read cognitive models	Mean 1,57

Regarding the relation of the behaviors represented in M ++, 85.7% of the experts considered that the representation is adequate compared to 14.3% that did not consider it (Fig. 5)

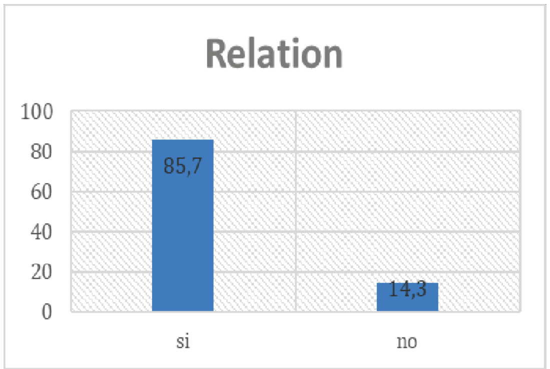

Fig.5 Relation of the Behaviors Represented in M ++

6 Conclusions

The formal representation of a trace reasoning of a specific cognitive function CARINA will allow structuring the bases for the design of metacognitive processes in CARINA architecture.

The relevance of this research lies in the possibility of having basic structures that allow of CARINA to "read" quickly what happens in the object level of architecture. The relevance of this research lies in the possibility of having basic structures that allow to the metalevel of CARINA "read" quickly what happens at the object level of the architecture and that way in future studies can be built metacognitive processes in architecture

Finally, with the realization of this research, the mathematical formalization and the stable computational implementation of the introspective monitoring mechanism based on reasoning traces of reasoning of the CARINA architecture were advanced.

Acknowledgment. This research was funded in part by the Internal Call for Sustainability of Research Groups of the University by Córdoba and by the project COL / 86815 – CONV. 647/15 of the Ministry of Information Technologies and Communications of Colombia (MinTIC)

References

Baldoni, M., Baresi, L., & Dastani, M. (2015). Engineering Multi-Agent Systems: Third International Workshop, EMAS 2015, Istanbul, Turkey, May 5, 2015, Revised, Selected, and Invited Papers (Vol. 9318). Springer.

Caro, M. F., Gómez, A. A., & Giraldo, J. C. (2017). Algorithmic Knowledge Profiles for Introspective Monitoring in Artificial Cognitive Agents. In 2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC) (pp. 475–481).

Caro, M. F., Josvula, D. P., Gomez, A. A., & Kennedy, C. M. (2018). Introduction to the CARINA Metacognitive Architecture. Proceedings of 2018 IEEE 17th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2018, (October), 530–540. https://doi.org/10.1109/ICCI-CC.2018.8482051

Caro, M. F., Josyula, D. P., Cox, M. T., & Jiménez, J. A. (2014). Design and validation of a metamodel for metacognition support in artificial intelligent systems. Biologically Inspired Cognitive Architectures, 9, (pp.82–104).

Caro, M. F., Josyula, D. P., Jiménez, J. A., Kennedy, C. M., & Cox, M. T. (2015). A domain-specific visual language for modeling metacognition in intelligent systems. Biologically Inspired Cognitive Architectures, 13, (pp.75–90). https://doi.org/10.1016/j.bica.2015.06.004

Chitil, O., & Luo, Y. (2007). Structure and Properties of Traces for Functional Programs. Electronic Notes in Theoretical Computer Science, 176(1), (pp.39–63). https://doi.org/10.1016/j.entcs.2006.10.032

Cox, M., & Raja, A. (2007). Metareasoning: A manifesto. BBN Technical.

Cox, M. T. (n.d.). Metacognition in Computation: A selected research review. Retrieved from https://pdfs.semanticscholar.org/cd31/8fa0c6abe1bb7b88e263862ef5dfe16703e8.pdf

Cox, M. T. (1997). An explicit representation of reasoning failures. In International Conference on Case-Based Reasoning (pp. 211–222).

Cox, M. T., & Raja, A. (n.d.). Metareasoning: An Introduction Metareasoning: A Manifesto. Retrieved from https://mitpress-request.mit.edu/sites/default/files/titles/content/9780262014809_sch_0001.pdf

Cox, M. T., & Raja, A. (2008). Metareasoning: A Manifesto. Retrieved from http://www.aaai.org/Papers/Workshops/2008/WS-08-07/WS08-07-001.pdf

Cox, M. T., & Ram, A. (1999). Introspective multistrategy learning: On the construction of learning strategies. Artificial Intelligence, 112, 1–55. Retrieved from https://ac.els-cdn.com/S0004370299000478/1-s2.0-S0004370299000478-main.pdf?_tid=d42b7fce-41d9-4ecd-b995-1c1ef2b8e31e&acdnat=1522765442_e9bd74b7c0f8a1d28075f9b0025b376c

Ethics in Research and Experimentation. (n.d.). Retrieved from http://www.cs.uu.nl/docs/vakken/arm/handouts/M7-ethics.pdf

Florez, M. A., Gomez, A. A., & Caro, M. F. (2018). Formal Representation of Introspective Reasoning Trace of a Cognitive Function in CARINA. Proceedings of 2018 IEEE 17th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2018, (October), (pp.620–627). https://doi.org/10.1109/ICCI-CC.2018.8482053

Fodor, J. A. (1975). The language of thought (Vol. 5). Harvard University Press.

Georgeff, M. (1984). A Theory of Action for MultiAgent Planning. Retrieved from http://www.aaai.org/Papers/AAAI/1984/AAAI84-015.pdf

Gómez, A. A., Caro, M. F., Solano, A. M., & Vega, Y. M. (2018). Trends of Educational Informatics in Latin America. International Journal of Software Science and Computational Intelligence (IJSSCI), 10(1), (pp.80–87). https://doi.org/10.4018/IJSSCI.2018010106

Isern, D., Gómez-Alonso, C., & Moreno, A. (n.d.). Methodological Development of a Multi-Agent System in the Healthcare Domain. Retrieved from https://s3.amazonaws.com/academia.edu.documents/7842867/isern08c.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1529112673&Signature=mxl7O8sagQyBY8ZTZeqSpvAOz8M%3D&response-content-disposition=inline%3B ilename%3DMethodological_development_of_a_multi-ag.pdf

Madera-doval, D. P., & Cardozo-soto, A. M. (2018). Design of metacognitive expectations of cognitive functions through. Diseño de expectativas metacognitivas de funciones cognitivas a través de representaciones ontológicas, 85(206), (pp.194–201).

Molina, A. I., Gallardo, J., Redondo, M. A., Ortega, M., & Giraldo, W. J. (2013). The Journal of Systems and Software Metamodel-driven definition of a visual modeling language for specifying interactive groupware applications: An empirical study. The Journal of Systems and Software, 86, (pp.1772–1789). https://doi.org/10.1016/j.jss.2012.07.049

Piccinini, G. (2010). Computation in physical systems. Scheutz, M. (n.d.). Computational versus Causal Complexity. Retrieved from https://hrilab.tufts.edu/publications/scheutz01mm.pdf

Sjøberg, D. I. K., Hannay, J. E., Hansen, O., Kampenes, V. B., Karahasanovi, A., Liborg, N.-K., & Rekdal, A. C. (n.d.). A Survey of Controlled Experiments in Software Engineering. Retrieved from http://www.idi.ntnu.no/grupper/su/publ/ebse/R01-formalexper-sjoberg-sep05.pdf

Sun, R., Zhang, X., & Mathews, R. (2005a). Modeling Meta-Cognition in a Cognitive Architecture. Retrieved from https://pdfs.semanticscholar.org/d145/5a166a388079001a09c3094c0fbe2d4e4407.pdf

Sun, R., Zhang, X., & Mathews, R. (2005b). Modeling meta-cognition in a cognitive architecture Action editor: Vasant Honavar. https://doi.org/10.1016/j.cogsys.2005.09.001