Defining a Strategy to Select Either of Closed/Open World Assumptions on Semantic Robots

Atilla Elçi
Department of Computer Engineering
Eastern Mediterranean University
Gazimagusa, TRNC, Mersin 10, Turkey
atilla.elci@emu.edu.tr

Behnam Rahnama
Department of Computer Engineering
Eastern Mediterranean University
Gazimagusa, TRNC, Mersin 10, Turkey
behnam.rahnama@emu.edu.tr

Saman Kamran
School of Information Technology
Eastern Mediterranean University
Gazimagusa, TRNC, Mersin 10, Turkey
kamran_saman@yahoo.com

Abstract

Semantic Web technology is based on the Open World Assumption (OWA) where absence of entities being searched does not entail negative response rather simply treated as facts "not available at the moment." On the one hand that indicates anticipation of future enhancements of the fact store while in other cases it is not preferred in situations where an authoritative answer is needed. Closed World Assumption (CWA) on the other hand returns definitive yes/no answers even in situations where future enhancements are inevitable. We wish to distinguish among the situations where one or the other assumption would be advantageous to use. Such distinction could cater for faster services discovery and more wholesome services orchestration. We apply this approach to discovering semantic Web services being offered by autonomous semantic robots out in the field and for building the unified worldview common to all. Consequently, the OWA-based paradigm alone is not suitable for most robotics platforms where cardinality and negation is not sound as well as in CWA. Therefore, we argue and display that a strategy to draw and utilize benefits of both world assumptions is beneficial.

1. Introduction

Internet and its effects to the community are discovered by people to facilitate information access; and now is the time to restructure the organization of huge mass of data to make it usable for the new era of information consumption by human and machines alike. That is the aim of the new generation of the Web, which is called "Semantic Web". Towards achieving that, there are two models rendering data understood by software agents and teaching ways of making decision based on available knowledge: Open World Assumption (OWA) and Closed World Assumption (CWA).

Web information is not too compatible with CWA. While results of OWA are more realistic, it suffers from lack of definitive answers as in CWA. Additionally, there are a few items we seek which are not satisfied due to weaknesses of OWA as with queries which require a definite yes/no answer. The answer "not available at the moment" causes ambiguity in most of robotics applications where a robot must know what to do in the next step. Table 1 presents a brief but essential comparison of features of both assumptions.

Table 1: OWA and CWA comparative definition

Open World Assumption (OWA)	**Closed World Assumption (CWA)**
When we have an empty Web Ontology Language (OWL) ontology, everything is possible. We then constrain ontology iteratively, making it more restrictive as we go. i.e., we state what is not possible. [1]	Closed World Systems require a place for each individual to be linked. i.e., we state what is possible. [1]
The existence of further individuals is possible if it is not explicitly excluded. [2]	One assumes that the knowledge base contains all known individuals and all known facts. [2]

0730-3157/08 $25.00 © 2008 IEEE
DOI 10.1109/COMPSAC.2008.182

If fact C is true in every model of Knowledge Base (KB), then C is a consequence of KB: $KB \vDash C$. If C is true in no model of KB, then its negation is true in every model of KB: $KB \vDash \neg C$. If C is true in some models but false in others, neither C nor its negation is a consequence of KB: $KB \nvDash C$; $KB \nvDash \neg C$. [1]	If a fact C is not a consequence of KB, accepted that its negation is a consequence of KB. $KB \nvDash C$ implies $KB \vDash \neg C$. [1]

We would like to compare decision-making between CWA and OWA in collaborative problem solving using multiple autonomous robots. Such comparison, we hope that, will result in finding benefits of each approaches in order to evolve a strategy to select either of CWA/OWA decision-making and reasoning systems based upon the constraints of the model.

Next section surveys applicability of CWA & OWA logic paradigms in Semantic Web. Section 3 introduces a link between Semantic Web and robotics to render intelligent robots. Then Section 4 puts forward an algorithm to take advantage of CWA and OWA based on prevailing local constraints to successfully carry out a task. Also shown is an application of the approach to collaboratively finding an exit from a labyrinth. The paper concludes with hints at possible future studies.

2. A Survey of Related Studies on Logic Paradigms

Both OWA and CWA have their places in many applications. OWA is good at describing knowledge in a way that is extensible, whereas CWA is good at constraining and validating data [1]. Therefore, trying to omit one of the approaches and rely only on the other would amount to missing partial benefits of either of them. Comparing OWA against CWA, as two modeling paradigms that may be enforced in Semantic Web, may show us how to better benefit from either or both.

In OWA, the existence of further individuals is assumed if not explicitly excluded. OWL [22] is better expressed by OWA. CWA assumes that the knowledge base contains all known individuals and all known facts [2]. The conflict between CWA and OWA earned the well-deserved title of the first Semantic Web battle [3]. OWA is much less prone to trigger an error; and, much more likely to generate new statements based on the facts of the model. CWA, on the other hand, is much stricter and less diplomatic. Recently, the W3C held a workshop on Rule Languages for Interoperability where it turned out that many people felt the need for more "closed world" approaches or, as some like to call it, "negation as failure" [3]. The research done by Horrocks et. al. explains well the fact that there is a battle starting between the two worlds [4]; a battle that might fragment the Semantic Web into two partially incompatible worlds. They discussed language architecture for Semantic Web, and in particular different proposals for extending this architecture with a rules component. They argued that an architecture maximizing compatibility with existing languages, in particular Resource Description Framework (RDF) and OWL, will benefit the development of Semantic Web; and that could still allow for forms of closed world assumption and negation as failure. Other alternatives were represented under the similar umbrella of extending OWA to support specifications and formalism of logic behind CWA.

Pan and Horrocks works on connecting RDF(S) and OWL Description Language (OWL DL) [5]. They proposed a novel modification of RDF(S) as a firm semantic foundation for many of the latest Description Logics-based Semantic Web ontology languages, including OWL DL. Much earlier, researchers in the field of logic programming realized the necessity of reasoning not in a wholly monotonic approach. Subrahmanian provides a survey of the state of the art in nonmonotonic logic programming (or, in a simpler word to mean the same: CWA) [6]. Katz and Parsia [7] later discusses providing nonmonotonic extension to OWL. They argue that an extension to OWL would be desirable, however, at its current state, the OWL syntax is too inflexible to capture even this syntactically simple-minded extension.

Patel-Schneider et al [8] presents a comparison of the two modeling paradigms. After considering differences between them, they argue that, although some CWA characteristics have their better utility, the open environment of Semantic Web is better served by OWA [9].

Others focus on closed-world reasoning [10] and local closed-world reasoning [11] in Semantic Web. The open world assumption makes OWL principally suitable to handle incomplete knowledge in Semantic Web scenarios; however, some scenarios at a particular instance require closed-world reasoning to succeed.

Localized CWA (LCWA) can be used in applications where an agent cannot assume complete information about planning or reasoning states. This type of OWA is generally necessary in most realistic robotics applications.

Researchers also highlight problems in matchmaking process due to OWA, when handling incomplete service descriptions [12]. They propose to use autoepistemic extensions to DLs (ADLs) to overcome these problems. ADLs allow for nonmonotonic reasoning and for querying DL knowledge bases under LCWA. Induction from answer sets in nonmonotonic logic programs is done based on the theory of nonmonotonic inductive logic programs (NILP) [13]. Additionally, Negation-as-Failure is one of the most important items, which should be included in an extension to OWA under some circumstances [14].

While seeking solutions to extend OWA with specifications of CWA and merging them together, a very apparent distance between these two worlds is sensible. This has sparked a heated debate in Semantic Web community, resulting in proposals for alternative ontology languages based entirely on logic programming. Pitching in to help resolving this debate, here we investigate practical use cases which seem to be better addressed by logic programming. Motik et al [15], presents a novel logic consisting of hybrid monotonic logic with negation-as-failure (MKNF) knowledge bases, which seamlessly integrates OWL with logic programming (LP). This study is very similar to the research done by Rondogiannis and Wadge [14]. Loyer and Straccia allow a combination of CWA and OWA at the same time as the default assumption in interpreting and reasoning over a truth space [16]. The result is a new paradigm entitled Any-World Assumption (AWA), which is conservative in the sense that under the "everywhere false" default assumption (CWA) the usual stable model semantics is captured.

Other hybrid models of logic similar to MKNF or AWA, have been provided by Horrocks et al. [17]. Finally, a research by Wang, et al. [18], extends the bridge between OWA and CWA by proposing a standardized framework for having both worlds under the same umbrella.

Still, there is no standard annotation as a minimal set of extensions to OWL for joining these two world assumptions. Those mentioned above such as, LCWA, MKNF, and AWA, and others proposed such as ALCK [7], SHOIN [17] and OKBC [18] are either designed for a specific query processing or are too limited to be generalized. These initiatives are yet to be endorsed by W3C, which favors the OWA assumption. Without such an endorsement, it is not likely for them to join the legacy logic-programming league of approaches.

3. A Bridge between Semantic Web and Robotics

Semantic Web technologies were created in part to involve and embed web-based data into reasoner and decision making engines. This fact leads to provide and establish a suitable platform for building a bridge between conventional robotics system and new reasoning structure suitable for web-based data processing. Robots prefer CWA, as they are weak in processing incomplete data such as an arbitrary maze or unsupervised discovery of uncharted territory. Semantic Web Services, on the other hand, provide a wide knowledge base of incomplete or unconnected instances, which may be lined up in a string to present a uniquely powerful and complete source of information as World Wide Web based on OWA.

Systems with distributed processing and control require distributed coordination in order to achieve a shared goal. Such systems may be realized using self-actuated agents donning semantics capability such as that of an autonomous semantic agent (ASA). Implementation of an ASA [19] as a Semantic Web service offered by a robot provides most of the features required [20]. A common site acts as the central registry of the web services in the field. Cooperatively solving a problem as intelligent junctions in traffic management information systems [21] is similar to a multi-agent robotic system solving a path to exit out of a maze. Solution to routing engines in railroad switch yard and line placement in making printed circuit boards are ramifications of the same theory of labyrinth solvers where nets should be efficiently short. These issues can be covered by cooperative labyrinth discovery robots (CLDRs) benefiting from OWA.

Several questions reside: firstly, how to apply OWA & CWA as core of decision-making, and secondly, how to define a strategy to select one of the two world assumptions for proper reasoning? The answer to the first question is simple and straightforward. For example, the ontology editor Protégé provides an environment for easily working with OWL ontology files. Add-in software such as Algernon, Prolog etc. can be installed on top of Protégé. Therefore, reasoning and running queries are possible on embedded OWL file based on OWA (in Algernon) and based on CWA (in Prolog). Protégé has other APIs to tie up with other software development frameworks such as Microsoft Visual Studio or Borland J++ Builder, Netbeans, etc. One can build own ontology by keying it in or drawing it using visualization tools on Protégé, and then apply reasoning through either OWA- or CWA-based paradigm. Let's say that each query is concatenated with a command from an actual robot. If for example, the scenario is moving the robot by one cell in labyrinth matrix, then the following takes place: reading the sensors to locate walls, sending a command to Protégé to update matrix ontology, and then sending a command to request decision-making, and finally, retrieving the inference outcome to move the robot according to the decision based on one of the world assumptions.

Attempting to provide an answer to the second question requires longer explanation on the implementation of software architecture of the core of decision-making. In the next Section, we take that issue up.

4. Defining a Strategy of Selecting CWA / OWA for a Task Based on Local Constraints

Problem spaces having fixed size, such as knowing all instances in a maze, vertices and edges in a graph etc. are better served through CWA. We know that the domain knowledge is complete. Therefore, CWA assumes all instances and facts are defined. On the other hand, problems with ambiguity of answer sets, having incomplete results for queries are better served through OWA. We come across many examples of that: finding a way out of an uncharted maze; routing through traffic network in a city with distributed control architecture; problems where estimation of probability is the main key for decision-making, etc. Suitability of OWA in such usecases is due to openness feature of problem set stemming from unknown state variables. In a way of exemplifying this, let us study the case of solving labyrinth discovery problem by cooperative mobile semantic robots.

In order to solve a maze with more than one entry to reach an exit gate, we may use more than one agent (a physical robot with ability to understand the semantics of the context, i.e., a Semantic Robot). Each agent discovers a part of the labyrinth area and shares its discovered knowledge. Agents behave in concordance with their terms and priorities helping each other to find exit (See Figure 1). For instance, while the agent working in a quarter of the matrix has first-forward priority another one elsewhere may have first-left priority. In addition, the intelligence of our maze solver ASAs focus on reshaping its priorities based on the visited neighboring blocks. Other decision-making algorithms such as Blind-Man, Virtual-Agent, and Deadlock-Finder are also used in ASAs to rearrange its action priority [20].

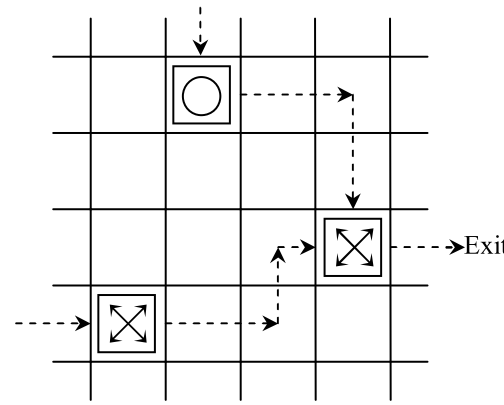

Figure 1: Cooperation scheme of robots to share results.

In order to decide to move to a side, an agent at a junction runs Min/Max Algorithm on the number of visits at each side. It selects the one with minimum number of visits or by running shortest path algorithm. What if all are the same or none of them is discovered yet?

Based on theoretical implication of our research, we have developed software to solve the Cooperative Labyrinth Discovery (CLD) problem using cooperating agents (robots) based on CWA restrictions. On the other hand, we have also developed an OWA version of decision making to provide a framework for testing and comparing these two approaches.

Each agent can only see the walls around the visited cell. In CWA version of the example a shared memory is updated by all robots while in OWA version, knowledge of paths are shared but each robot has own local counter of all visited cells. Therefore, in both cases, the knowledge on labyrinth is incomplete, however, in OWA decision is based on knowledge of agent itself. If a decision is not reachable, then agent tries to communicate with other agents to get their visit counts on neighboring cells.

Let us consider the case represented in Figure 2, for robot R at (7, 4). Notice that left hand side number in each cell represents local counter of first robot, and the right hand side value represents the local counter of the second robot in case of solving the problem using two robots.

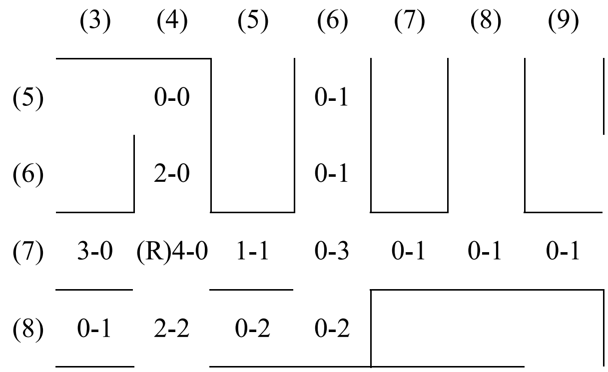

Figure 2: Local Counters in OWA decision Maker.

Counter values of shared memory are summation of these two numbers. For instance, shared memory counter value of cell (7, 5) is equal to two. The general query is if there is a path from current cell to exit. CWA decision maker returns false as exit have not been found yet, consequently, some paths cannot be retrieved by shortest path algorithm. Min/max algorithm of the decision maker is undecided between selecting (6, 4) or (7, 5) for both have shared memory values equal to 2. Switching to OWA strategy, decision maker returns (7, 5) immediately as target of the next move where minimum local counter is equal to 1 among all neighbor cells of (7, 4). Where OWA decision maker returns “not available now,” in more complicated cases, local counters of robots are asked one by one until a solid answer is reached. In worst case of sharing all local counters like CWA, decision is made by simple priorities. Let’s study the example of Figure 3.

In Figure 3, the left-hand side labyrinth represents the shared memory that robots are updating; and, the right hand side is the complete maze. As introduced at the beginning of this section, due to fixed size of problem set and knowing entrance and exit positions in labyrinth, CWA may be used to obtain a better answer set.

However, in an uncharted labyrinth, probability and estimation should be used to select a path among a set of possible but yet undiscovered ones. For this reason in the OWA version, we provide separate memory for the counter of number of visits of each cell by a robot.

The cooperative labyrinth solver program applies the following sequence of steps for each robot:

1. Has the current cell been visited before?
   1.1. Read walls of the current cell if not visited.
   1.2. Update local counter and shared counter
   1.3. Update shared memory of paths
2. Run Decision-Maker based on CWA.
   2.1. Has the next-move-to cell been visited before?
      2.1.1. If yes run shortest path algorithm
      2.1.2. Otherwise run min/max algorithm on shared counter of visited cells
         2.1.2.1. if min/max algorithm returns more than one minimum, run OWA based on local counters
         2.1.2.2. if OWA results in ‘unknown’ answer, then choose one of the results of CWA randomly or based on a priority
3. Move the robot to the next-move-to position
4. Check if the selected cell is an exit cell.
   4.1. If not, repeat from Step 1.

The experience on our algorithm proved that there are cases for which CWA alone cannot answer the query; and in such cases OWA returns a better estimation of possible answer. Therefore, we argue that a strategy to draw and utilize benefits of both world assumptions is necessary side-by-side. This strategy is based on following local constraints of problem definition. We cannot forego one of the two world assumptions but having them together leads to better answer sets to queries in robotic platforms. This powers the robot with better decision estimates and answers while CWA alone would have returned false as answer to unknown problems.

5. Conclusions and Future Study

Semantic Web Services provide a new approach to communication, situation- and context-awareness, and knowledge representation for reasoning by multiple agents. To this end, there are two models for rendering data understandable for computers and teaching ways of reaching conclusions based on knowledge: Open World Assumption (OWA) and Closed World Assumption (CWA).

As Web information is not compatible with closeness assumption, OWA returns more realistic results and reasoning out of web data. However, it suffers from lack of strict formalism of logic as in CWA. Additionally, in some circumstances there are a few items in our interest, which are not satisfied due to weaknesses of OWA.

Employing both CWA and OWA together helps garnering better results out of reasoning on semantic data for robotics applications. In this study, we focused

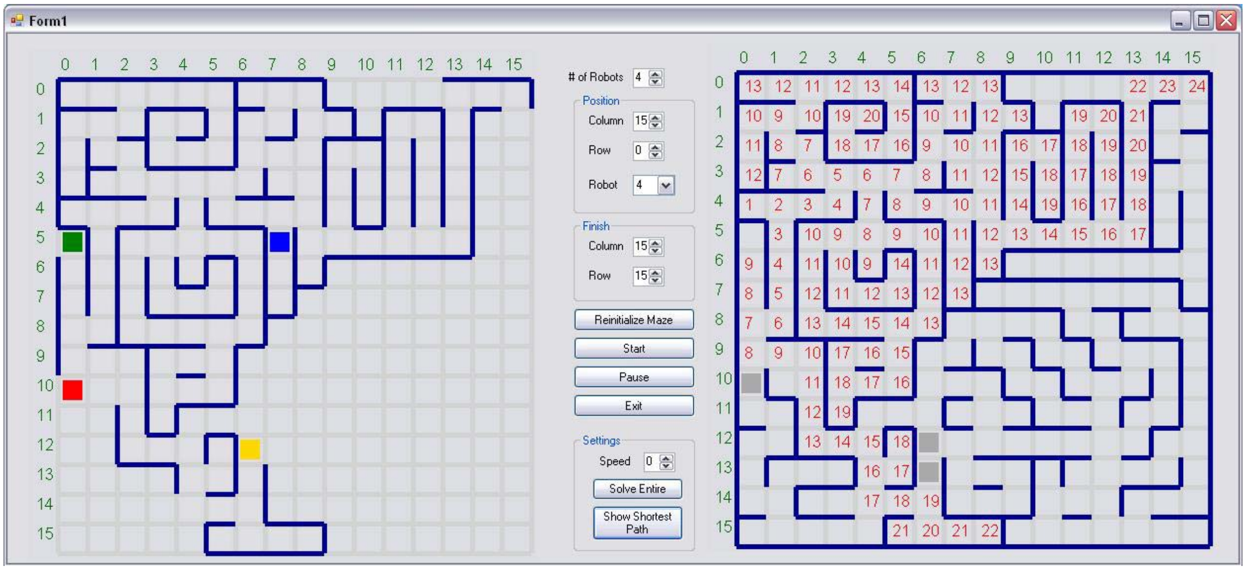

Figure 3: Implementing CLDR based on CWA: sample game instance.

firstly on providing OWA based solution to labyrinth problems. Where these were previously handled through CWA, we thus showed that OWA is also applicable to replace the conventional logic programming approaches. Then, we compared the achieved results. Furthermore, we refined outcome by providing a strategy in deciding on selecting either of CWA/OWA to power decision-making and reasoning. Finally, we applied this paradigm as the core of decision-making on collaborative labyrinth discovery robots. The usecase was the situation where each semantic robotic agent utilizes a combined set of results out of CWA- and OWA-based inference engines to solve the given task cooperatively.

We would like to extend our research on top of our implemented platform to formalize a possibly minimal set of extensions to OWA based on the results we will achieve in our practical tests. This might constitute a step towards a new generation of collective intelligence on cooperative robots and a convenient alternative in aid of legacy AI approaches.

6. References

[1] Drummond, Nick and Shearer, Rob. The Open World Assumption. [Online] 2006. [Cited: November 1, 2007.] http://cs.man.ac.uk/~drummond/presentations/OWA.pdf.

[2] Staab, Steffen. Semantic Web OWL. *ISWeb – Lecture "Semantic Web".* [Online] 2007. [Cited: November 12, 2007.] http://www.uni-kobenz.de/~staab/lehre/ss05/sw/6-owl.pdf.

[3] Mazzocchi, Stefano. Closed World vs. Open World: the First Semantic Web Battle. *Stefano's Linotype.* [Online] June 16, 2005. [Cited: October 1, 2007.] http://www.betaversion.org/~stefano/linotype/news/91/.

[4] *Semantic Web Architecture: Stack or Two Towers?* Horrocks, Ian, et al. [ed.] In Francois Fages and Sylvain Soliman. Springer LNCS, 2005. Principles and Practice of Semantic Web Reasoning. Vol. 3703, pp. 37-41.

[5] *RDFS(FA): Connecting RDF(S) and OWL DL.* Pan, Jeff and Horrocks, Ian. 2, 2007, IEEE Trans. on Knowledge and Data Engineering, Vol. 19, pp. 192-206.

[6] *Nonmonotonic logic programming.* Subrahmanian, V.S. 1, IEEE, Feb 1999, IEEE Transactions on Knowledge and Data Engineering, Vol. 11, pp. 143-152.

[7] Katz, Y. and Parsia, B. *Towards a Nonmonotonic Extension to OWL.* University of Maryland.

[8] *A Comparison of Two Modelling Paradigms in the Semantic Web.* Patel-Schneider, Peter F and Horrocks, Ian. ACM, 2006. Fifteenth International World Wide Web Conference. pp. 3-12.

[9] *A comparison of two modelling paradigms in the Semantic Web.* Patel-Schneider, Peter F and Horrocks, Ian. September 16, 2007, Journal of Web Semantics, pp. 240-250.

[10] *Closed World Reasoning in the Semantic Web through Epistemic Operators.* Grimm, Stephan and Motik, Boris. [ed.] Bernardo Cuenca Grau, et al. Galway, Ireland, 2005. Workshop on OWL: Experiences and Directions.

[11] Doherty, Patrick, Lukaszewicz, Witold and Szalas, Andrzej. Efficient Reasoning Using the Local Closed-World Assumption. *Artificial Intelligence: Methodology, Systems, and Applications.*Springer Berlin / Heidelberg, 2000, Vol. 1904, pp. 21-34.

[12] *Matching Semantic Service Descriptions with Local Closed-World Reasoning.* Grimm, Stephan, Motik, Boris and Preist, Chris. 2006. pp. 575-589.

[13] *Induction from answer sets in nonmonotonic logic programs.* Sakama, C. 2, ACM Press, 2005, ACM Transactions on Computational Logic (TOCL), Vol. 6.

[14] *Minimum model semantics for logic programs with negation-as-failure.* Rondogiannis, P. and Wadge, W. 2, s.l. : ACM Press, 2005, ACM Transactions on Computational Logic (TOCL), Vol. 6.

[15] *Can OWL and Logic Programming Live Together Happily Ever After?* Motik, Boris, et al. Springer, Lecture Notes in Computer Science, 2006. 5th International Semantic Web Conference. Vol. 4273, pp. 501-514.

[16] *Any-world assumptions in logic programming.* Loyer, Yann and Straccia, Umberto. 2-3, Essex, UK : Elsevier Science, September 2005, Theoretical Computer Science, Vol. 342, pp. 351-381. ISSN:0304-3975 .

[17] *Hybrid Logics and Ontology Languages.* Horrocks, Ian, Glimm, Birte and Sattler, Ulrike. Electronic Notes in Theoretical Computer Science, 2007. International Workshop on Hybrid Logic. Vol. 174, pp. 3-14.

[18] Wang, Hai H, et al. Frames and OWL Side by Side. [Online] http://protege.stanford.edu/conference/2006/submissions/abstracts/7.2_Wang_Hai_Protege_conf.pdf.

[19] *Considerations on a New Software Architecture for Distributed Environments Using Autonomous Semantic Agents.* Elçi, Atilla and Rahnama, Behnam. Proc. IEEE, 2005 2nd International Workshop on Software Cybernetics, 29th IEEE COMPSAC 2005. pp. 133-138.

[20] *Applying Semantic Web in Engineering a Modular Architecture for Design and Implementation of a Cooperative Labyrinth Discovery Robot.* Elçi, Atilla and Rahnama, Behnam. Gemikonağı, Northern Cyprus, 30 Nov - 1 Dec 2006. Proc. 4th FAE International Symposium on Computer Science and Engineering.

[21] *Intelligent Junction: Improving the Quality of Life for Mobile Citizens through better Traffic Management.* Elçi, Atilla and Rahnama, Behnam. Ankara, Turkey : TBD Publications, in Turkish, 8-9 June 2006. Proc. YvKB 2006.

[22] Smith, M. K., Welty, C., and McGuinness, D. L. (eds.). OWL Web Ontology Language Guide. W3C Recommendation, 10 Feb 2004. http://www.w3.org/TR/owl-guide/