Hindawi Publishing Corporation
ISRN Artificial Intelligence
Volume 2013, Article ID 691707, 11 pages
http://dx.doi.org/10.1155/2013/691707

*Research Article*

# Probabilistic Multiagent Reasoning over Annotated Amalgamated F-Logic Ontologies

**Markus Schatten**

*University of Zagreb, Faculty of Organization and Informatics, Pavlinska 2, 42000 Varaždin, Croatia*

Correspondence should be addressed to Markus Schatten; markus.schatten@foi.hr

Received 2 May 2013; Accepted 17 June 2013

Academic Editors: H. A. Guvenir, J. A. Hernandez, and C. Kotropoulos



In a multiagent system (MAS), agents can have different opinions about a given problem. In order to solve the problem collectively they have to reach consensus about the ontology of the problem. A solution to probabilistic reasoning in such an environment by using a social network of trust is given. It is shown that frame logic can be annotated and amalgamated by using this approach which gives a foundation for collective ontology development in MAS. Consider the following problem: a set of agents in a multiagent system (MAS) model a certain domain in order to collectively solve a problem. Their opinions about the domain differ in various ways. The agents are connected into a social network defined by trust relations. The problem to be solved is how to obtain consensus about the domain.

## 1. Introduction

To formalize the problem let $A = \{a_1, \ldots, a_n\}$ be a set of agents, let $\tau$ be a trust relation defined over $A \times A$, and let $D = \{o_1, \ldots, o_m\}$ be a problem domain consisting of a set of objects. Let further $S$ be a set of all possible statements about $D$, and let $O$ be a relation over $A \times S$. We will denote by $O$ the social ontology expressed by the agents. What is the probability that a certain statement from the expressed statements in $O$ is true?

By modeling some domains of interest (using a formalism like ontologies, knowledge bases, or other models) a person expresses his/her knowledge about it. Thus the main concept of interest in modeling any domain is knowledge. Nonaka and Takeuchi once defined knowledge as a "justified true belief" [1] whereby this definition is usually credited to Plato. This means that the modeling person implicitly presumes that the expressed statements in his/her model are true. On the other hand if one asks the important question *what is the truth?*, we arrive at one of the fundamental philosophical questions. Nietzsche once argued in [2] that a person is unable to prove the truth of a statement which is nothing more than the invention of fixed conventions for merely practical purposes, like repose, security, and/or consistence. According to this view, no one can prove that this paper is not just a fantasy of the reader reading it.

The previously outlined definition of knowledge includes, intentionally or not, two more crucial concepts: *justified* and *belief*. An individual will consider something to be true that he believes in, and, from that perspective, the overall truth will be a set of statements that the community believes in. This mutual belief makes this set of statements justified. The truth was once that the Earth was the center of the universe until philosophers and scientists started to question that theory. The Earth was also once a flat surface residing on the back of an elephant. So an interesting fact about the truth, from this perspective, is that it evolves depending on the different beliefs of a certain community.

In an environment where a community of agents collaborates in modeling a domain there is a chance that there will be disagreements about the domain which can yield certain inconsistencies in the model. A good example of such disagreements is the so-called "editor wars" on Wikipedia the popular free online encyclopedia. A belief about the war in ex-Yugoslavia will likely differ between a Croat and a Serb, but they will probably share the same beliefs about fundamental mathematical algebra.

Following this perspective, our conceptualization of statements as units of formalized knowledge will consider the probability of giving a true statement a matter of justification. An agent is justified if other members of a social system believe in his statements. Herein we would like to outline a social network metric introduced by Bonacich [3] called eigenvector centrality which calculates the centrality of a node based on the centrality's of its adjacent nodes. Eigenvector centrality assigns relative values to all nodes of a social network based on the principle that connections to nodes with high values contribute more to the value of the node in question than equal connections to nodes with low values. In a way, if we interpret the network under consideration as a network of trust, it yields an approximation of the probability that a certain agent will say the truth in a statement as perceived by the other agents of the network. The use of eigenvector centrality here is arbitrary; any other metric with the described properties could be used as well.

In order to express knowledge about a certain domain, one needs an adequate language. Herein we will use frame logic or F-logic introduced by [4], which is an object-oriented, deductive knowledge base and ontology language. The use of F-logic here is arbitrary, and any other formal (or informal) language could be used that allows expressing an ontology of a given domain. Nevertheless, F-logic allows us to reason about concepts (classes of objects), objects (instances of classes), attributes (properties of objects) and methods (behavior of objects), by defining rules over the domain, which makes it much more user friendly than other approaches.

## 2. Introducing Frame Logic

The syntax of F-logic is defined as follows [4].

*Definition 1.* The alphabet $\Sigma_{\mathscr{F}}$ of an F-logic language $\mathscr{L}_{\mathscr{F}}$ consists of the following:

(i) a set of object constructors, $\mathscr{F}$;

(ii) an infinite set of variables, $\mathscr{V}$;

(iii) auxiliary symbols, such as, (, ), [, ], $\rightarrow$, $\twoheadrightarrow$, $\bullet\!\!\rightarrow$, $\bullet\!\!\twoheadrightarrow$, $\Rightarrow$, and $\Rrightarrow$; and

(iv) usual logical connectives and quantifiers, $\vee$, $\wedge$, $\neg$, $\leftarrow$, $\forall$, and $\exists$.

Object constructors (the elements of $\mathscr{F}$) play the role of function symbols in F-logic whereby each function symbol has an arity. The arity is a nonnegative integer that represents the number of arguments the symbol can take. A constant is a symbol with arity 0, and symbols with arity ≥1 are used to construct larger terms out of simpler ones. An id term is a usual first-order term composed of function symbols and variables, as in predicate calculus. The set of all variable free or ground id terms is denoted by $U(\mathscr{F})$ and is commonly known as Herbrand Universe. Id terms play the role of logical object identities in F-logic which is a logical abstraction of physical object identities.

A language in F-logic consists of a set of formulae constructed out of alphabet symbols. The simplest formulae in F-logic are called F-molecules.

*Definition 2.* A molecule in F-logic is one of the following statements:

(i) an is-a assertion of the form $C :: D$ ($C$ is a nonstrict subclass of $D$) or of the form $O : C$ ($O$ is a member of class $C$), where $C$, $D$, and $O$ are id terms;

(ii) an object molecule of the form O [a ";" separated list of method expressions], where $O$ is an id term that denotes an object. A method expression can be either a noninheritable data expression, an inheritable data expression, or a signature expression.

(a) Noninheritable data expressions can be in either of the following two forms.

(1) A non-inheritable scalar expression ScalMethod@$Q_1, \ldots, Q_k \rightarrow T$, $(k \geqslant 0)$.

(2) A non-inheritable set-valued expression SetMethod@$R_1, \ldots, R_l \twoheadrightarrow \{S_1, \ldots, S_m\}$ $(l, m \geqslant 0)$.

(b) Inheritable scalar and set-valued expression are equivalent to their non-inheritable counterparts except that $\rightarrow$ is replaced with $\bullet\!\!\rightarrow$ and $\twoheadrightarrow$ with $\bullet\!\!\twoheadrightarrow$.

(c) Signature expression can also take two different forms.

(1) A scalar signature expression ScalMethod@$V_1, \ldots, V_n \Rightarrow (A_1, \ldots, A_r)$, $(n, r \geqslant 0)$.

(2) A set-valued signature expression SetMethod@$W_1, \ldots, W_s \Rrightarrow (B_1, \ldots, B_t)$ $(s, t \geqslant 0)$.

All methods' left hand sides (e.g., $Q_i$, $R_i$, $V_i$, and $W_i$) denote arguments, whilst the right hand sides (e.g., $T$, $S_i$, $A_i$, and $B_i$) denote method outputs. Single-headed arrows ($\rightarrow$, $\bullet\!\!\rightarrow$, and $\Rightarrow$) denote scalar methods, and double-headed arrows ($\twoheadrightarrow$, $\bullet\!\!\twoheadrightarrow$, and $\Rrightarrow$) denote set-valued methods.

As in a lot of other logic, F-formulae are built out of simpler ones by using the usual logical connectives and quantifiers mentioned above.

*Definition 3.* A formula in F-logic is defined recursively:

(i) F-molecules are F-formulae;

(ii) $\varphi \vee \psi$, $\varphi \wedge \psi$, and $\neg\varphi$ are F-formulae if so are $\varphi$ and $\psi$;

(iii) $\forall X\varphi$ and $\exists Y\psi$ are F-formulae if so are $\varphi$ and $\psi$, and $X$ and $Y$ are variables.

F-logic further allows us to define logic programs. One of the popular class of logic programs is Horn programs.

*Definition 4.* A Horn F-program consists of Horn rules, which are statements of the form

$$head \longleftarrow body. \quad (1)$$

Whereby *head* is an F-molecule, and *body* is a conjunction of F-molecules. Since the statement is a clause, we consider all variables to be implicitly universally quantified.

For our purpose these definitions of F-logic are sufficient, but the interested reader is advised to consult [4] for profound logical foundations of object-oriented and frame based languages.

## 3. Introducing Social Network Analysis

A formal approach to defining social networks is graph theory [5].

*Definition 5.* A *graph* $\mathcal{G}$ is the pair $(\mathcal{N}, \mathcal{A})$ whereby $\mathcal{N}$ represents the set of *verticles* or *nodes* and $\mathcal{A} \subseteq \mathcal{N} \times \mathcal{N}$ the set of *edges* or *arcs* connecting pairs from $\mathcal{N}$.

A graph can be represented with the so-called adjacency matrix.

*Definition 6.* Let $\mathcal{G}$ be a graph defined with the set of nodes $\{n_1, n_2, \ldots, n_m\}$ and edges $\{e_1, e_2, \ldots, e_l\}$. For every $i$, $j$ $(1 \leqslant i \leqslant m$ and $1 \leqslant j \leqslant m)$ one defines

$$a_{ij} = \begin{cases} 1, & \text{if there is an edge between nodes } n_i \text{ and } n_j, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Matrix $A = [a_{ij}]$ is then the adjacency matrix of graph $\mathcal{G}$. The matrix $i$ is symmetric since if there is an edge between nodes $n_i$ and $n_j$, then clearly there is also an edge between $n_j$ and $n_i$. Thus $A = [a_{ij}] = [a_{ji}]$.

The notion of directed and valued-directed graphs is of special importance to our study.

*Definition 7.* A *directed graph* or *digraph* $\mathcal{G}$ is the pair $(\mathcal{N}, \mathcal{A})$, whereby $\mathcal{N}$ represents the set of *nodes* and $\mathcal{A} \subseteq \mathcal{N} \times \mathcal{N}$ the set of ordered pairs of elements from $\mathcal{N}$ that represents the set of graph *arcs*.

*Definition 8.* A *valued* or *weighted digraph* $\mathcal{G}_{\mathcal{V}}$ is the triple $(\mathcal{N}, \mathcal{A}, \mathcal{V})$ whereby $\mathcal{N}$ represents the set of *nodes* or *verticles*, $\mathcal{A} \subseteq \mathcal{N} \times \mathcal{N}$ the set of ordered pairs of elements from $\mathcal{N}$ that represent the set of graph *arcs*, and $\mathcal{V} : \mathcal{N} \rightarrow \mathbb{R}$ a function that attaches values or weights to nodes.

A social network can be represented as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N}$ denotes the set of actors and $\mathcal{A}$ denotes the set of relations between them [6]. If the relations are directed (e.g. support, influence, message sending, trust, etc.), we can conceptualize a social network as a directed graph. If the relations additionally can be measured in a numerical way, social networks can be represented as valued digraphs.

One of the main applications of graph theory to social network analysis is the identification of the "most important" actors inside a social network. There are lots of different methods and algorithms that allow us to calculate the importance, prominence, degree, closeness, betweenness, information, differential status, or rank of an actor. As previously mentioned we will use the eigenvector centrality to annotate agents' statements.

*Definition 9.* Let $p_i$ denote the value or weight of node $n_i$, and let $[a_{ij}]$ be the adjacency matrix of the network. For node $n_i$ let the centrality value be proportional to the sum of all values of nodes which are connected to it. Hence

$$p_i = \frac{1}{\lambda} \cdot \sum_{j \in M(j)} p_j = \frac{1}{\lambda} \cdot \sum_{j=1}^{N} a_{ij} \cdot p_j, \tag{3}$$

where $M(i)$ is the set of nodes that are connected to the $i$th node, $N$ is the total number of nodes, and $\lambda$ is a constant. In vector notation this can be rewritten as

$$p = \frac{1}{\lambda} \cdot A \cdot p \text{ or as the eigenvector equation } A \cdot p = \lambda \cdot p. \tag{4}$$

PageRank is a variant of the eigenvector centrality measure, which we decided to use herein. PageRank was developed by Google or more precise by Larry Page (from where the word play PageRank comes from) and Sergey Brin. They used this graph analysis algorithm, for the ranking of web pages on a web search engine. The algorithm uses not only the content of a web page but also the incoming and outgoing links. Incoming links are hyperlinks from other web pages pointing to the page under consideration, and outgoing links are hyperlinks to other pages to which the page under consideration points.

PageRank is iterative and starts with a random page following its outgoing hyperlinks. It could be understood as a Markov process in which states are web pages, and transitions (which are all of equal probability) are the hyperlinks between them. The problem of pages which do not have any outgoing links, as well as the problem of loops, is solved through a jump to a random page. To ensure fairness (because of a huge base of possible pages), a transition to a random page is added to every page which has the probability $q$ and is in most cases 0.15. The equation which is used for rank calculation (which could be thought of like the probability that a random user will open this particular page) is as follows:

$$\text{PageRank}(p_i) = \frac{q}{N} + (1 - q) \sum_{p_j \in M(p_i)} \frac{\text{PageRank}(p_j)}{L(p_j)}, \tag{5}$$

where $p_1, p_2, \ldots, p_N$ are nodes under consideration, $M(p_i)$ the set of nodes pointing to $p_i$, $L(p_i)$ the number of arcs which come from node $p_j$, and $N$ the number of all nodes [7, 8].

A very convenient feature of PageRank is that the sum of all ranks is 1. Thus, semantically, we can interpret the ranking value of agents (or actors in the social network) participating in a given MAS as *the probability that an agent will say the truth in the perception of the others*. In the following we will use the ranking, obtained through such an algorithm in this sense.

## 4. Probability Annotation

As shown in Section 2 there are basically three types of statements agents can make: (1) is-a relations, (2) object molecules, and (3) Horn rules. While is-a relations and Horn rules can be considered atomic, object molecules can be compound since object molecules of the form

$$\begin{gathered}
o\left[a_1 \longrightarrow v_1;\ldots;a_n \longrightarrow v_n\right]\\
o\left[a_1 \twoheadrightarrow v_1;\ldots;a_n \twoheadrightarrow v_n\right]\\
o\left[a_1 \bullet\!\!\longrightarrow v_1;\ldots;a_n \bullet\!\!\longrightarrow v_n\right]\\
o\left[a_1 \bullet\!\!\twoheadrightarrow v_1;\ldots;a_n \bullet\!\!\twoheadrightarrow v_n\right]\\
o\left[a_1 \Longrightarrow v_1;\ldots;a_n \Longrightarrow v_n\right]\\
o\left[a_1 \Longrightarrow\!\!\!\!\!\Rightarrow v_1;\ldots;a_n \Longrightarrow\!\!\!\!\!\Rightarrow v_n\right]
\end{gathered} \tag{6}$$

can be rewritten as corresponding atomic F-molecules

$$\begin{gathered}
o\left[a_1 \longrightarrow v_1\right]\cdots o\left[a_n \longrightarrow v_n\right]\\
o\left[a_1 \twoheadrightarrow v_1\right]\cdots o\left[a_n \twoheadrightarrow v_n\right]\\
o\left[a_1 \bullet\!\!\longrightarrow v_1\right]\cdots o\left[a_n \bullet\!\!\longrightarrow v_n\right]\\
o\left[a_1 \bullet\!\!\twoheadrightarrow v_1\right]\cdots o\left[a_n \bullet\!\!\twoheadrightarrow v_n\right]\\
o\left[a_1 \Longrightarrow v_1\right]\cdots o\left[a_n \Longrightarrow v_n\right]\\
o\left[a_1 \Longrightarrow\!\!\!\!\!\Rightarrow v_1\right]\cdots o\left[a_n \Longrightarrow\!\!\!\!\!\Rightarrow v_n\right].
\end{gathered} \tag{7}$$

We will consider in the following that all F-molecule statements are atomic. Now we are able to define the annotation scheme of agent statements as follows.

*Definition 10.* Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of statements, let $A = \{a_1, a_2, \ldots, a_n\}$ be a set of agents, let $O : S \times A$ be a corresponding social ontology, let $\tau$ be a trust relation between agents over $A \times A$, and let $\phi : A \rightarrow [0,1]$ be a function that assigns ranks to agents based on $\tau$. Then the *annotation* $\overline{\overline{\wedge}}$ of the statements is defined as follows:

$$s \overline{\overline{\wedge}} \pi, \quad \pi = \sum_{(a,s)\in O} \phi(a). \tag{8}$$

An extension to such a probability annotation is the situation when statements can have a negative valency. This happens when a particular agent disagrees to a statement of another agent. Such an annotation would be defined as follows.

*Definition 11.* Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of signed statements, let $A = \{a_1, a_2, \ldots, a_n\}$ be a set of agents, let $O : S \times A$ be a corresponding social ontology, let $\tau$ be a trust relation between agents over $A \times A$, and let $\phi : A \rightarrow [0,1]$ be a function that assigns ranks to agents based on $\tau$. Then the *annotation* $\overline{\overline{\wedge}}$ of the statements is defined as follows:

$$\begin{aligned}
&s \overline{\overline{\wedge}} \pi, \pi\\
&= \begin{cases}
\displaystyle\sum_{(a,s)\in O} \phi(a) - \sum_{(a,-s)\in O} \phi(a) & \text{if } \displaystyle\sum_{(a,s)\in O} \phi(a) > \sum_{(a,-s)\in O} \phi(a),\\
0 & \text{if } \displaystyle\sum_{(a,s)\in O} \phi(a) \leqslant \sum_{(a,-s)\in O} \phi(a).
\end{cases}
\end{aligned} \tag{9}$$

Such a definition is needed in order to avoid possible negative probability (the case when disagreement is greater than approvement).

## 5. Query Execution

In a concrete system we need to provide a mechanism for query execution that will allow agents to issue queries of the following form:

$$Q_p : F \overline{\overline{\wedge}} p, \tag{10}$$

where $F$ is any formula in frame logic and $p$ a probability. The semantics of the query is: *does the formula F hold with probability p with regard to the social ontology?*

The solution of this problem is equivalent to finding the probabilities of all possible solutions of query $F$

$$Q : F. \tag{11}$$

*Definition 12.* Let $R_Q = \{r_1, r_2, \ldots, r_n\}$ be a set of solutions to query $Q$; then $R_{Q_p}$ is a subset of $R_Q$ consisting of those solutions from $R_Q$ which probability is greater or equal to $p$ and represents the set of solutions to query $Q_p$.

The probability of a solution $p(r_i)$ is obtained by a set of production rules.

*Rule 1.* If $r_i$ is a conjunction of two formulas $r_{i1}$ and $r_{i2}$, then $p(r_i) = p(r_{i1}) \cdot p(r_{i2})$.

*Rule 2.* If $r_i$ is a disjunction of two formulas $r_{i1}$ and $r_{i2}$, then $p(r_i) = p(r_{i1}) + p(r_{i2})$.

*Rule 3.* If $r_i$ is an F-molecule if the form is $i[an \rightarrow av]$, then $p(r_i) = \min(p(an), p(av))$.

The implications of these three definitions are given in the following four theorems.

**Theorem 13.** *If $r_i$ is an F-molecule of the form $i[an_1 \rightarrow av_1, \ldots, an_n \rightarrow av_n]$, then $p(r_i) = \prod_{i=1}^{n} \min(p(an_i), p(av_i))$.*

*Proof.* Since $r_i$ in this case can be written as:

$$i\left[an_1 \longrightarrow av_1\right] \wedge \cdots \wedge i\left[an_v \longrightarrow av_v\right], \tag{12}$$

and due to Rule 3 the probabilities of the components of this conjunction are $\min(p(an_1), p(av_1)), \ldots, \min(p(an_n), p(av_n))$. Due to Rule 1 the probability of a conjunction is the product of the probabilities of its elements which yields $\prod_{i=1}^{n} \min(p(an_i), p(av_i))$. □

**Theorem 14.** *If $r_i$ is an F-molecule of the form $i : c[an_1 \rightarrow av_1, \ldots, an_n \rightarrow av_n]$, then $p(r_i) = p(i : c) \cdot \prod_{i=1}^{n} \min(p(an_i), p(av_i))$.*

*Proof.* Since the given F-molecule can be written as

$$i : c \wedge i\left[an_1 \longrightarrow av_1\right] \wedge \cdots \wedge i\left[an_n \longrightarrow av_n\right], \tag{13}$$

the proof is analogous to the proof of Theorem 13. □

**Theorem 15.** *If $r_i$ is a statement of generalization of the form $c_1 :: c_2$, if $P$ is the set of all paths between $c_1$ and $c_2$, and if $\blacktriangleright$ is the relation of immediate generalization, then*

$$p(r_i) = \sum_{pa \in P} \prod_{c_j \blacktriangleright c_i \in pa} p(c_j \blacktriangleright c_i). \tag{14}$$

*Proof.* Since any class hierarchy can be presented as a directed graph, it is obvious that there has to be at least one path from $c_1$ to $c_2$. If the opposite was true, the statement would not hold and thus wouldn't be in the initial solution set.

For the statement $c_1 :: c_2$ to hold, at least one path statement of the form

$$pa_x = c_1 \blacktriangleright c_{x1} \wedge c_{x1} \blacktriangleright c_{x2} \wedge \cdots \wedge c_{xn} \blacktriangleright c_2 \tag{15}$$

has to hold as well. This yields according to Rule 1 that the probability of one path would be

$$p(pa_x) = \prod_{c_j \blacktriangleright c_i \in pa} p(c_j \blacktriangleright c_i). \tag{16}$$

Since there is a probability that there are multiple paths which are alternative possibilities for proving the same premise, it holds that

$$pa_1 \vee pa_2 \vee \cdots \vee pa_m. \tag{17}$$

Thus from Rule 2 we get

$$p(c_1 :: c_2) = \sum_{pa \in P} \prod_{c_j \blacktriangleright c_i \in pa} p(c_j \blacktriangleright c_i) \tag{18}$$

what we wanted to prove. □

**Theorem 16.** *If $r_i$ is a statement of classification of the form $i : c$, then*

$$p(r_i) = p(i) \cdot \sum_{pa \in P} \prod_{c_j \blacktriangleright c_i \in pa} p(c_j \blacktriangleright c_i). \tag{19}$$

*Proof.* Since the statement $r_i$ can be written as

$$r_i = i : c_1 \wedge c_1 :: c, \tag{20}$$

the given probability is a consequence of Rule 1 and Theorem 15. □

A special case of query execution is when the social ontology contains Horn rules. Such rules are also subject to probability annotation. Thus we have

$$\text{rule} : Head \longleftarrow Body \overline{\overline{\wedge}}\, p, \tag{21}$$

where $p$ is the annotated probability of the rule. In order to provide a mechanism to deal with such probability annotated rules, we will establish an extended definition by using an additional counter predicate for each Horn rule. Thus, each rule is extended as

$$Head \longleftarrow Body \wedge CounterPredicate, \tag{22}$$

whereby *CounterPredicate* is a predicate which will count the number of times the particular rule has been successfully executed for finding a given solution.

The query execution scheme has to be altered as well. Instead of finding only the solutions from formula $F$ an additional variable for every rule in the social ontology is added to the formula. For $n$ rules we would thus have

$$Q : F \wedge count(?r_1) \wedge count(?r_2) \wedge \cdots \wedge count(?r_n). \tag{23}$$

In order to calculate the probability of a result obtained by using some probability annotated rule we establish the following definition.

*Definition 17.* Let $r$ be a result obtained with probability $p_F$ by query $F$ from a social ontology, let $p_r$ be the probability of rule $R$, and let $c$ be the number of times rule $R$ was executed during the derivation of result $r$. The final probability of $r$ is then defined as

$$p(r) = p_F \cdot p_r^c. \tag{24}$$

This definition is intuitive since for the obtainment of result $r$ the rule $R$ has to hold $c$ times. Thus if a social ontology contains $n$ rules $(R_1, \ldots, R_n)$ their corresponding annotated probabilities are $p_{r1}, \ldots, p_{rn}$, and numbers of execution during derivation of result $r$ are $c_1, \ldots, c_n$, then the final probability is defined as

$$p(r) = p_F \cdot \prod_{i=1}^{n} p_{ri}^{c_i}. \tag{25}$$

## 6. Annotated Reasoning Example

In order to demonstrate the approach we will take the following (imaginary) example of an MAS (all images, names, and motives are taken from the 1968 movie "Yellow Submarine" produced by United Artists (UA) and King Features Syndicate). Presume we have a problem domain entitled "Pepperland" with objects entitled "Music" and "Purpose of life." Let us further presume that we have six agents collaborating on this problem, namely, "John," "Paul," "Ringo," "George," "Max," and "Glove."

Another intelligent agent "Jeremy Hilary Boob Ph.D (Nowhere man)" tries to reason about the domain, but as it comes out, the domain is inconsistent. Table 1 shows the different viewpoints of agents.

Due to the disagreement on different issues a normal query would yield at least questionable results. For instance, if the disagreement statements are ignored in frame logic syntax, the domain would be represented with a set of sentences similar to the following:

$$\begin{aligned} o_{\text{Music}} &: \text{evil noise} \\ o_{\text{Music}} &: \text{harmonious sounds} \\ o_{\text{Purpose of life}} &[\text{main purpose} \twoheadrightarrow \\ &\{\text{glove, love, drums}\}]. \end{aligned} \tag{26}$$

Table 1: Viewpoints of "Pepperland" agents.

| | Music | Purpose of life |
|---|---|---|
| John | : harmonious sounds | Main purpose $\rightarrow$ love |
| Paul | : harmonious sounds | Main purpose $\rightarrow$ love |
| Ringo | : harmonious sounds | Main purpose $\rightarrow$ drums |
| George | Disagrees to (: evil noise) | Main purpose $\rightarrow$ love |
| Max | : evil noise | Disagrees to (main purpose $\rightarrow$ love) |
| Glove | : evil noise | Main purpose $\rightarrow$ glove |

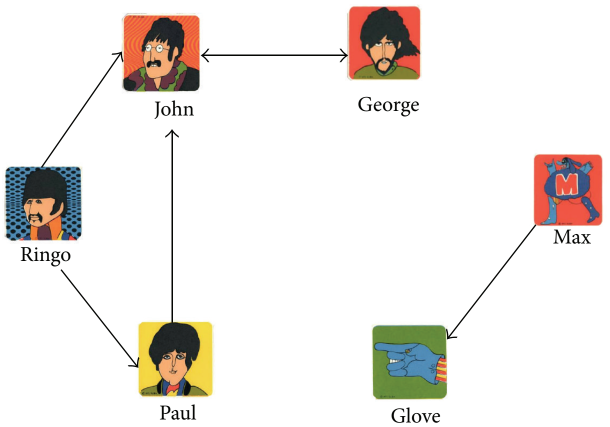



Figure 1: Social network of "Pepperland."

Thus a query asking for the class to which the object entitled "Music" belongs

$$? - \ o_{\text{Music}} \ : \ ?class \tag{27}$$

would yield two valid answers, namely, "evil noise" and "harmonious sounds." Likewise if querying for the value of the "main purpose" attribute of object $o_{\text{Purpose of life}}$, for example,

$$? - \ o_{\text{Purpose of life}} \ [\text{main purpose} \ \longrightarrow ?purpose] \tag{28}$$

the valid answers would be "glove," "love," and "drums." But, these answers do not reflect the actual state of the MAS, since one answer is more meaningful to it than the others.

Nowhere man thinks hard and comes up with a solution. The agents form a social network of trust are shown in Figure 1.

The figure reads as follows: Ringo trusts Paul and John, Paul trusts John, John trusts George, George trusts John, Max trusts Glove, and Glove does not trust anyone. Using the previously described PageRank algorithm Nowhere man was able to order the agents by their respective rank (Table 2).

Now, Nowhere man uses these rankings to annotate the statements given by the agents:

$$\begin{aligned} p\,(\text{evil noise}) &= \text{Rank}\,(\text{Max}) \\ &\quad + \text{Rank}\,(\text{Glove}) \\ &\quad - \text{Rank}\,(\text{George}) \\ &= 0.065609. \end{aligned} \tag{29}$$

As we can see the probability that object $o_{\text{Music}}$ is and "evil noise" is equal to the sum of agents' rankings who agree to this statement (Glove and Max) minus the sum of agents' rankings who disagree (George). Note that if an agent had expressed the same statement twice with the same attribute name, his ranking would be counted only once. Also note that, if an agent would have agreed and disagreed to a statement, his sum would be zero, since he would be at the agreed and disagreed side.

Table 2: Trust ranking of the "Pepperland" agents.

| Agent | Ranking |
|---|---|
| John | 0.303391 |
| Glove | 0.289855 |
| George | 0.267724 |
| Paul | 0.060667 |
| Max | 0.043478 |
| Ringo | 0.034884 |

From this probability calculation Nowhere man is able to conclude that the formula $o_{\text{Music}} \ : \ \text{evil noise}$ holds with probability 0.065609. Likewise he calculates the probability of $o_{\text{Music}} \ : \ \text{harmonious sounds}$

$$\begin{aligned} p\,(\text{harmonious sounds}) &= \text{Rank}\,(\text{John}) \\ &\quad + \text{Rank}\,(\text{Paul}) \\ &\quad + \text{Rank}\,(\text{Ringo}) \\ &= 0.398942. \end{aligned} \tag{30}$$

He can now conclude that $o_{\text{Music}} \ : \ \text{harmonious sounds}$ holds more likely than $o_{\text{Music}} \ : \ \text{evil noise}$ with regard to the social network of agents. From these calculations Nowhere man concludes that the final solutions to query $? - \ o_{\text{Music}} \ : \ ?class$ are

$$\begin{aligned} &?class = \text{evil noise} \overset{=}{\wedge} 0.065609 \\ &?class = \text{harmonious sounds} \overset{=}{\wedge} 0.398942. \end{aligned} \tag{31}$$

Nowhere man continues reasoning and calculates the probabilities for the other queries

$$\begin{aligned} p\,(\text{main purpose}) &= \text{Rank}\,(\text{John}) \\ &\quad + \text{Rank}\,(\text{Paul}) \\ &\quad + \text{Rank}\,(\text{Ringo}) \\ &\quad + \text{Rank}\,(\text{George}) \\ &\quad - \text{Rank}\,(\text{Max}) \\ &\quad + \text{Rank}\,(\text{Glove}) \\ &= 0.913043, \end{aligned}$$

$$\begin{aligned} p(\text{love}) &= \text{Rank}(\text{John}) \\ &\quad + \text{Rank}(\text{Paul}) \\ &\quad + \text{Rank}(\text{George}) \\ &\quad - \text{Rank}(\text{Max}) \\ &= 0.588304, \\ p(\text{glove}) &= \text{Rank}(\text{Glove}) \\ &= 0.289855, \\ p(\text{drums}) &= \text{Rank}(\text{Ringo}) \\ &= 0.034884. \end{aligned} \tag{32}$$

From these calculations Nowhere man concludes that $o_{\text{Purpose of life}}$ [main purpose $\rightarrow$ love] is most likely to hold with $p = 0.588304$. The final result of the query

$$?- \ o_{\text{Purpose of life}} \ [\text{main purpose} \longrightarrow ?\textit{purpose}] \tag{33}$$

is then

$$\begin{aligned} ?\textit{purpose} &= \textit{love} \overline{\overline{\wedge}} 0.588304 \\ ?\textit{purpose} &= \textit{glove} \overline{\overline{\wedge}} 0.289855 \\ ?\textit{purpose} &= \textit{drums} \overline{\overline{\wedge}} 0.034884. \end{aligned} \tag{34}$$

Now we can complicate things a bit to see the other parts of the approach in action. Assume now that John has expressed a statement that relates the object entitled "Music" to the object entitled "Purpose of life" and named the attribute "has to do with." We would now have the following social ontology:

$$\begin{aligned} &o_{\text{Music}} \ : \ \text{evil noise} \\ &o_{\text{Music}} \ : \ \text{harmonious sounds} \\ &o_{\text{Music}} \ [\longrightarrow \ o_{\text{Purpose of life}}] \\ &o_{\text{Purpose of life}} \ [\text{main purpose} \twoheadrightarrow \{\text{glove; love, drums}\}]. \end{aligned} \tag{35}$$

Now suppose that Nowhere man wants to issue the following query:

$$?- \ ?o1 \ : \ ?c \ [?a \longrightarrow ?o2] \ \wedge \ ?o2 \, [\text{main purpose} \longrightarrow ?p]. \tag{36}$$

The solutions using "normal" frame logic are

$$\begin{aligned} s_1 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{evil noise} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{glove}, \\ s_2 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{evil noise} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{love}, \\ s_3 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{evil noise} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{drums}, \\ s_4 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{harmonious sounds} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{glove}, \\ s_5 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{harmonious sounds} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{love}, \\ s_6 &: \\ &?o1 = o_{\text{Music}} \\ &?c = \text{harmonious sounds} \\ &?a = \text{has to do with} \\ &?o2 = o_{\text{Purpose of life}} \\ &?p = \text{drums}. \end{aligned} \tag{37}$$

To calculate the probabilities Nowhere man uses the following procedure. The variables in the query are exchanged with the actual values for a given solution:

$s_1$: $o_{\text{Music}}$ : evil noise [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ glove], s$_2$: $o_{\text{Music}}$ : evil noise [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ love],

s$_3$: $o_{\text{Music}}$ : evil noise [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ drums],

s$_4$: $o_{\text{Music}}$ : harmonious sounds [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ glove],

s$_5$: $o_{\text{Music}}$ : harmonious sounds [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ love],

s$_6$: $o_{\text{Music}}$ : harmonious sounds [has to do with $\longrightarrow$ $o_{\text{Purpose of life}}$] $\wedge$ $o_{\text{Purpose of life}}$[main purpose $\longrightarrow$ drums].

Now according to rule 1 the conjunction becomes

$$\begin{aligned}
p(s_1) &= p\left(o_{\text{Music}} : \text{evil noise}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{glove}\right]\right), \\
p(s_2) &= p\left(o_{\text{Music}} : \text{evil noise}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{love}\right]\right), \\
p(s_3) &= p\left(o_{\text{Music}} : \text{evil noise}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{drums}\right]\right), \\
p(s_4) &= p\left(o_{\text{Music}} : \text{harmonious sounds}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{glove}\right]\right), \\
p(s_5) &= p\left(o_{\text{Music}} : \text{harmonious sounds}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{love}\right]\right), \\
p(s_6) &= p\left(o_{\text{Music}} : \text{harmonious sounds}\left[\text{has to do with} \longrightarrow o_{\text{Purpose of life}}\right]\right) \\
&\quad \cdot p\left(o_{\text{Purpose of life}}\left[\text{main purpose} \longrightarrow \text{drums}\right]\right).
\end{aligned} \tag{38}$$

The second parts of the equations were already calculated, and according to Theorem 14 the first parts of the equations become

$$\begin{aligned}
p(s_1) &= p(o_{\text{Music}} : \text{evil noise}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.289855, \\
p(s_2) &= p(o_{\text{Music}} : \text{evil noise}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.588304, \\
p(s_3) &= p(o_{\text{Music}} : \text{evil noise}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.034884, \\
p(s_4) &= p(o_{\text{Music}} : \text{harmonious sounds}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.289855, \\
p(s_5) &= p(o_{\text{Music}} : \text{harmonious sounds}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.588304, \\
p(s_6) &= p(o_{\text{Music}} : \text{harmonious sounds}) \\
&\quad \cdot \min\left(p(\text{has to do with}), p\left(o_{\text{Purpose of life}}\right)\right) \\
&\quad \cdot 0.034884.
\end{aligned} \tag{39}$$

We already know the probabilities of the is-a statement, and since

$$\begin{aligned}
p(\text{has to do with}) &= p\left(o_{\text{Purpose of life}}\right) \\
&= \phi(\text{John}) = 0.303391,
\end{aligned} \tag{40}$$

the equations become

$$\begin{aligned}
p(s_1) &= 0.065609 \cdot 0.303391 \cdot 0.289855, \\
p(s_2) &= 0.065609 \cdot 0.303391 \cdot 0.588304, \\
p(s_3) &= 0.065609 \cdot 0.303391 \cdot 0.034884, \\
p(s_4) &= 0.398942 \cdot 0.303391 \cdot 0.289855, \\
p(s_5) &= 0.398942 \cdot 0.303391 \cdot 0.588304, \\
p(s_6) &= 0.398942 \cdot 0.303391 \cdot 0.034884,
\end{aligned} \tag{41}$$

and finally

$$p(s_1) = 0.005770,$$
$$p(s_2) = 0.011710,$$
$$p(s_3) = 0.000694,$$
$$p(s_4) = 0.035083, \quad (42)$$
$$p(s_5) = 0.071206,$$
$$p(s_6) = 0.004222.$$

## 7. Amalgamation

To provide a mechanism for agents to query multiple annotated social ontologies we decided to use the principles of amalgamation. The model of knowledge base amalgamation which is based on online querying of underlaying sources is described in [9]. The intention of amalgamation is to show if a given solution holds in any of the underlaying sources.

Since the local annotations of different ontologies that are subject to amalgamation do not necessarily hold for the global ontology, we need to introduce a mechanism to integrate the ontologies in a coherent way which will yield global annotations. Since the set of ontologies is a product of a set of respective social agent networks surrounding them, we decided to firstly integrate the social networks in order to provide the necessary foundation for global annotation.

*Definition 18.* The integration of $n$ social networks represented with the valued digraphs $(\mathcal{N}_1, \mathcal{A}_1, \mathcal{V}_1), \ldots, (\mathcal{N}_n, \mathcal{A}_n, \mathcal{V}_n)$ is given as the valued digraph $(\mathcal{N}_1 \cup \cdots \cup \mathcal{N}_z, \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_n, \mathcal{V})$, where $\mathcal{V}$ is a function $\mathcal{V} : \mathcal{N}_1 \cup \cdots \cup \mathcal{N}_n \to \mathbb{R}$ that attaches values to nodes.

In particular $\mathcal{V}$ will be a social network analysis metric or in our case a variant of the eigenvector centrality. Now we can define the integration of ontologies as follows.

*Definition 19.* Let $O_1, \ldots, O_n$ be sets of statements as defined above representing particular social ontologies. The integration is given as $O_1 \cup \cdots \cup O_n$.

What remains is to provide the annotation that is at the same time the amalgamation scheme.

*Definition 20.* Let $(\mathcal{N}_1 \cup \cdots \cup \mathcal{N}_n, \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_n, \mathcal{V})$ be the integration of $n$ social networks of agents, let $O_1 \cup \cdots \cup O_n$ be the integration of their corresponding social ontologies, let $\tau$ be a trust relation between agents, and let $\phi : A \to [0, 1]$ be a function that assigns ranks to agents based on $\tau$; then the amalgamated annotation scheme $\overline{\overline{\wedge}}$ of the metadata statements is defined as follows:

$$s \overline{\overline{\wedge}} \pi, \quad \pi = \sum_{(a,s) \in O_1 \cup \cdots \cup O_n} \phi(a). \quad (43)$$

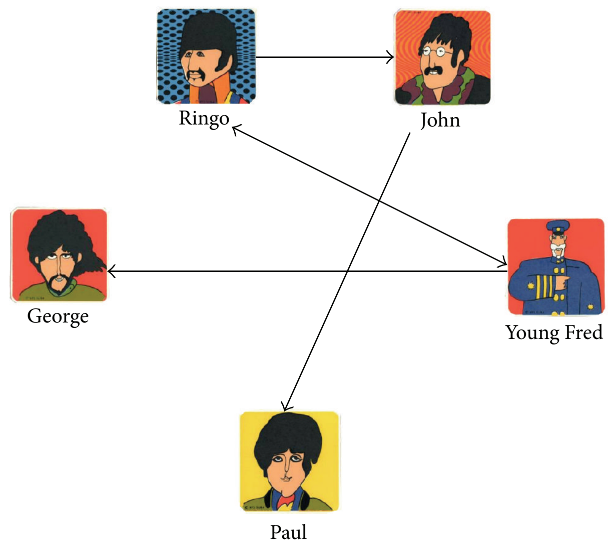



Figure 2: Social network of "Yellow submarine."

## 8. Amalgamated Annotated Reasoning Example

To demonstrate the amalgamation approach proposed here let us again assume that our intelligent agent "Jeremy Hilary Boob Ph.D. (Nowhere man)" tries to reason about the "Pepperland" domain, but this time he wants to draw conclusions from the domain "Yellow submarine" as well. The "Yellow submarine" domain is being modeled by "Ringo," "John," "Paul," "George," and "Young Fred" which form the social network shown in Figure 2. Since the contents of this domain as well as the particular ranks of the agents in it will not be used further in the example, they have been left out.

Since Nowhere man wants to reason about both domains he needs to find a way to amalgamate these two domains.

Again he thinks hard and comes up with the following solution. All he needs to do is to integrate the two social networks together and recalculate the ranks of all agents of this newly established social network in order to reannotate the metainformation in both domains.

Since the networks of "Pepperland" and "Yellow submarine" can be represented as the following sets of tuples:

$$\mathcal{G}_{\text{Pepperland}} = \{ (\text{Ringo}, \text{John}), (\text{Ringo}, \text{Paul}), (\text{Paul}, \text{John}), (\text{John}, \text{George}), (\text{George}, \text{John}), (\text{Max}, \text{Glove}) \}, \quad (44)$$

$$\begin{aligned}\mathscr{G}_{\text{Yellow submarine}} = \{ & \\ & (\text{Ringo}, \text{John}), \\ & (\text{Ringo}, \text{Young Fred}), \\ & (\text{John}, \text{Paul}), \\ & (\text{Young Fred}, \text{Ringo}), \\ & (\text{Young Fred}, \text{George}) \\ \}, &\end{aligned} \tag{45}$$

all he needs is to find $\mathscr{G}_A = \mathscr{G}_{\text{Pepperland}} \cup \mathscr{G}_{\text{Yellow submarine}}$ and recalculate the ranks of this new network. Thus

$$\begin{aligned}\mathscr{G}_A = \{ & \\ & (\text{Ringo}, \text{John}), \\ & (\text{Ringo}, \text{Paul}), \\ & (\text{Paul}, \text{John}), \\ & (\text{John}, \text{George}), \\ & (\text{George}, \text{John}), \\ & (\text{Max}, \text{Glove}) \\ & (\text{Ringo}, \text{Young Fred}), \\ & (\text{John}, \text{Paul}), \\ & (\text{Young Fred}, \text{Ringo}), \\ & (\text{Young Fred}, \text{George}) \\ \}. &\end{aligned} \tag{46}$$

The newly established integrated social network is shown in Figure 3.

Now Nowhere man calculates the ranks of this new network and uses the previously described procedure to annotate the meta information (Section 4) and reason about the amalgamated domain (Section 5).

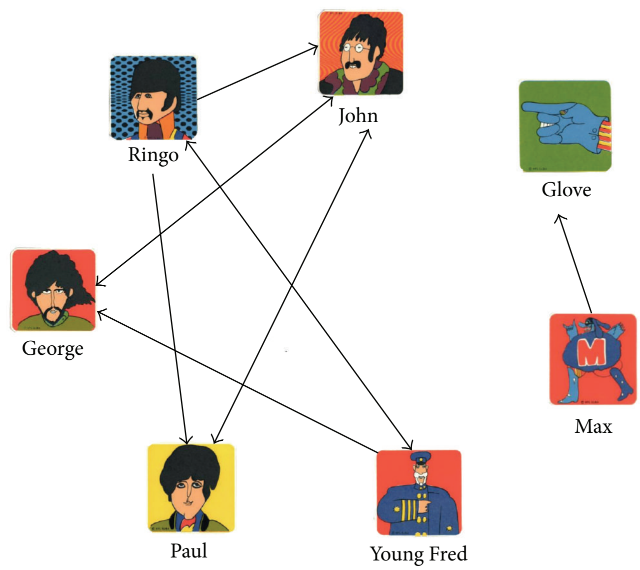


Figure 3: The integration of two social networks.

## 9. Towards a Distributed Application

As we could see from the previous examples, in order to gain accurate knowledge and accurate probabilities about a certain domain, we had to introduce an all-knowing agent (Nowhere man). This agent had to be aware of all knowledge of each agent and all trust relations they engage in. Such a scenario is not feasible for large-scale MAS (LSMAS). Thus we need to provide a mechanism to let agents reason in a distributed manner and still get accurate enough results.

This problem consists of two parts; namely, an agent needs (1) to acquire an accurate approximation of the ranks of each agent in its network and (2) to acquire knowledge about the knowledge of other agents. The first part deals with annotation and the second with amalgamation of the ontology.

A solution to the first problem might be to calculate ranks in a distributed manner, as has been shown in [10]. In this way agents acquire approximate knowledge about the ranks of their neighbouring agents.

The second problem could be used by the proposed algorithm for amalgamation. Each agent can ask agents it trusts about their knowledge and then amalgamate their ontology with its own. In this way the agent acquires continuously better knowledge about its local environment. We could have easily considered Nowhere man in the last example to be doing the just described procedure—asking one agent after another about their knowledge.

## 10. Possible Application areas

In order to provide a practical example, consider a network of store-and-forward e-mail routing agents in which spam bots try to send unsolicited messages. Some routers (agents) might be under the control of spam bots and send out messages which might be malicious to users and other routers. The domain these agents reason about is the domain of spam messages—for example, which message from which user forwarded by which router and what kind of content is spam and should be discarded.

This scenario can be modeled by using the previously described approach: agents form trust relations and mutually exchange new rules about spam filtering. An agent will amalgamate rules (ontologies) of other agents with its own but will decide about a message (using an adequate query) based not only on the given rules but also on the probability annotation given by the network of trust.

## 11. Related Work

Alternative approaches to measuring trust in the form of the reputation inference and the SUNNY algorithm are presented in [11, 12], respectively. Both of these could have been used instead of PageRank in the approach outlined herein. A much more elaborated system of measuring reputation and likewise trust in MAS called the *Regret* system is presented in [13]. It is based on three different dimensions of reputation (individual, social, and ontological) and allows for measuring several types of reputation in parallel. The approach is partly incompatible with our approach, but several adjustments would allow us to combine both approaches.

A different approach to a similar problem related to trust management in the Semantic Web is presented in [14]. It provides a profound model based on path algebra and inspired by Markov models. It provides a method of deriving the degree of belief in a statement that is explicitly asserted by one or more individuals in a network of trust, whilst a calculus for computing the belief in derived statements is left to future research. Herein a formalism for deriving the belief in any computable statement is presented for F-logic.

## 12. Conclusion

When agents have to solve a problem collectively, they have to reach consensus about the domain since their opinions can differ. Especially when agents are self-interested, their goals in a given situation can vary quite intensively. Herein an approach to reaching this consensus based on a network of trust between agents has been presented which is a generalization of the work done in [15, 16] which dealt with semantic wiki systems and semantic social networks, respectively. By using a network analysis trust ranks of agents can be calculated which can be interpreted as an approximation of the probability that a certain agent will say the truth. Using this interpretation an annotation scheme for F-logic based Horn programs has been developed which allows agents to reason about the modeled domain and make decisions based on the probability that a certain statement (derived or explicit) is true. Based on this annotation scheme and the network of trust an amalgamation scheme has been developed as well, which allow agents to reason about multiple domains.

Still, there are open questions: how does the approach scale in fully decentralized environments like LSMAS? What are the implications of self-interest or could agents develop strategies to "lie" on purpose to attain their goals? These and similar questions are subject to our future research.

## References


[1] I. Nonaka and H. Takeuchi, *The Knowledge-Creating Company, How Japanese Companies Create the Dynamics of Innovation*, Oxford University Press, 1995.

[2] F. Nietzsche, "Über wahrheit und lüge im außermoralischen sinn," 1873, http://www.textlog.de/455.html.

[3] P. Bonacich, "Factoring and weighting approaches to clique identification," *Journal of Mathematical Sociology*, vol. 2, pp. 113–120, 1972.

[4] M. Kifer, G. Lausen, and J. Wu, "Logical foundations of object-oriented and frame-based languages," *Journal of the Association for Computing Machinery*, vol. 42, no. 4, pp. 741–843, 1995.

[5] S. Wasserman and K. Faust, "Social network analysis, methods and applications," in *Structural Analysis in the Social Sciences*, Cambridge University Press, 1994.

[6] P. Mika, *Social Networks and the Semantic Web*, Springer, New York, NY, USA, 2007.

[7] S. Brin, "The anatomy of a large-scale hypertextual Web search engine," *Computer Networks*, vol. 30, pp. 107–117, 1998.

[8] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: bringing order to the web," 1999.

[9] A. Lovrenčić and M. Čubrilo, "Amalgamation of heterogeneous data sources using amalgamated annotated hilog," in *Proceedings of the 3rd IEEE Conference on Intelligent Engineering Systems (INES'99)*, 1999.

[10] Y. Zhu and X. Li, "Distributed pagerank computation based on iterative aggregation-disaggregation methods," in *Proceedings of the 14th ACM international conference on Information and knowledge management*, pp. 578–585, 2005.

[11] J. Golbeck and J. Hendler, "Accuracy of metrics for inferring trust and reputation in semantic web-based social networks," in *Proceedings of the 14th International Conference (EKAW '04)*, pp. 116–131, October 2004.

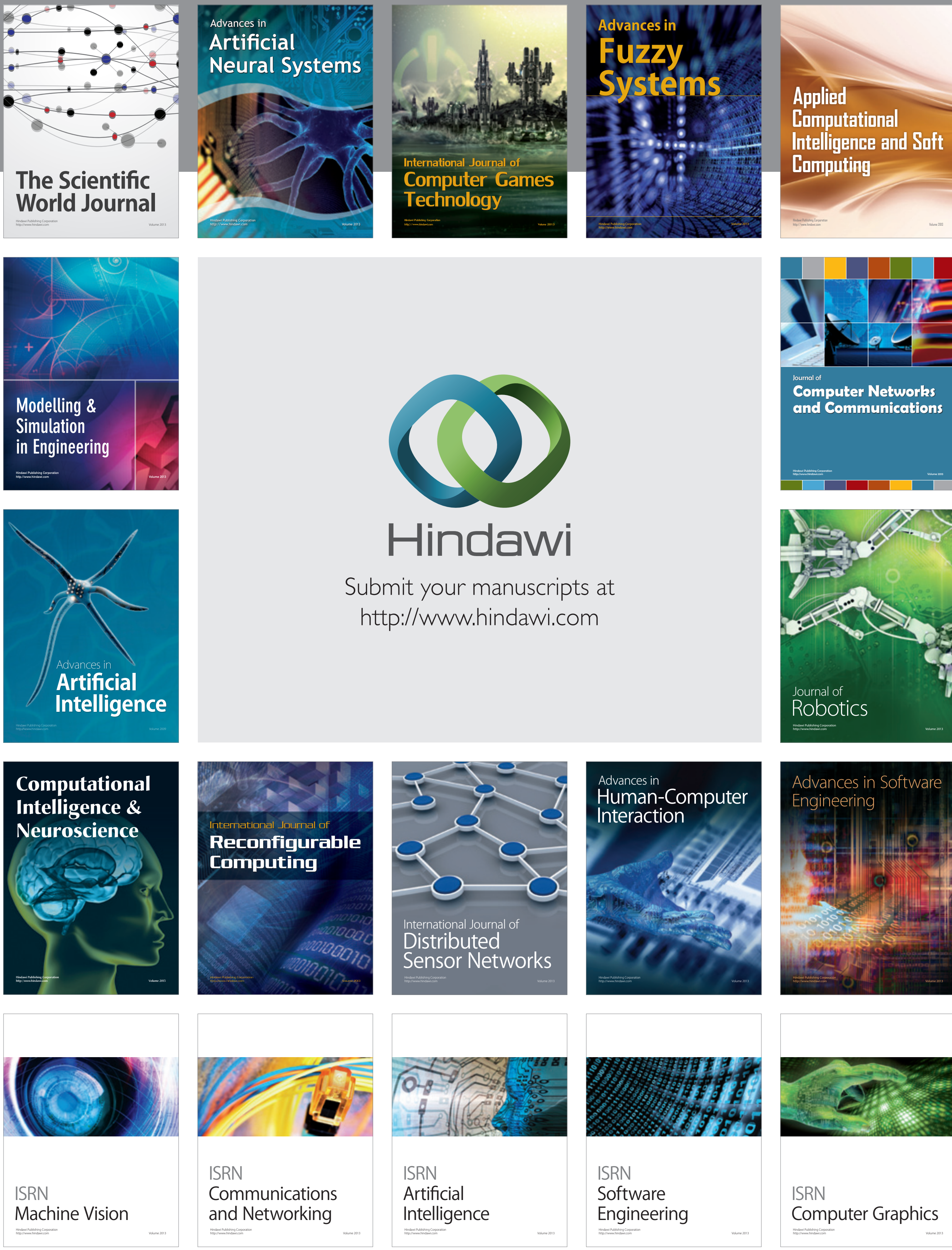
[12] U. Kuter and J. Golbeck, "Using probabilistic confidence models for trust inference in web-based social networks," *ACM Transactions on Internet Technology*, vol. 10, no. 2, article 8, 2010.

[13] J. Sabater and C. Sierra, "Reputation and social network analysis in multi-agent systems," in *Proceedings of the 1st International Joint Conference on Autonomous Agents adn Multiagent Systems (AAMAS '02)*, pp. 475–482, New York, NY, USA, July 2002.

[14] M. Richardson, R. Agrawal, and P. Domingos, "Trust management for the semantic web," in *Proceedings of the 2nd International Semantic Web Conference*, pp. 351–368, 2003.

[15] M. Schatten, *Programming Languages for Autopoiesis Facilitating Semantic Wiki Systems [Ph.D. thesis]*, University of Zagreb, Faculty of Organization and Informatics, Varaždin, Croatia, 2010.

[16] M. Schatten, "Knowledge management in semantic social networks," *Computational and Mathematical Organization Theory*, pp. 1–31, 2012.


The Scientific World Journal
Advances in Artificial Neural Systems
International Journal of Computer Games Technology
Advances in Fuzzy Systems
Applied Computational Intelligence and Soft Computing
Modelling & Simulation in Engineering
Journal of Computer Networks and Communications
Hindawi
Submit your manuscripts at
http://www.hindawi.com
Advances in Artificial Intelligence
Journal of Robotics
Computational Intelligence & Neuroscience
International Journal of Reconfigurable Computing
International Journal of Distributed Sensor Networks
Advances in Human-Computer Interaction
Advances in Software Engineering
ISRN Machine Vision
ISRN Communications and Networking
ISRN Artificial Intelligence
ISRN Software Engineering
ISRN Computer Graphics