Agents that grow by means of arguability and mobility

Wataru Kawakami, Yuichi Umeda, and Hajime Sawamura

Department of Information Engineering and Graduate School of Science and Technology, Niigata University
8050, 2-cho, Ikarashi, Niigata, 950-2181 JAPAN
{kawakami, umeda, sawamura}@cs.ie.niigata-u.ac.jp

Abstract. This paper explores a new frontier of mobile agents, not from technological views but from intrinsic views of mobility. We address ourselves to constructing agents, anthropomorphic computational objects, in which argumentation and mobility play important roles in agents' growth over the computer network. Argumentation is a prominent apparatus to attain both communication and computation (Argumentation = Computation + Communication), and mobility is an ability and activity found inherently to the human. Combining these two, we attempt to construct two kinds of agents that grow by means of argurability and mobility, and illustrate their application and implementation.

1 Introduction

In this paper, we are concerned with a complex of two scientific or technological cultures: argumentation and mobility that we think agents should have as the basic components of agents' capabilities. Recently there has been a growing interest in argumentation in the fields of computer science and artificial intelligence as wel as in human and social sciences. In fact, argumentation has been recognized as a very natural way of conceptualising nonmonotonic reasoning (e.g., [12]) and as a general framework for relating nonmonotonic logics of different styles (e. g., [7]), and used in the issue-based information system, gIBIS [6] and in the computer-supported collaborative argumentation (CSCA) [20].

In relation to agent-oriented computing, argumentation yields a most promising approach to such phenomena as dialogue (communication) among agents and decision by interacting agents, more specificallly, negotiation, cooperation, coordination, consensus attainment, conflict resolution, etc. For instance, Parsons et al. [16] propose BDI agents that negotiate through argumentation. Tambe and Jung [23] uses Toulmin's argument model [26] for intra-team conflict resolution. Sycara [22] proposes a combination of argumentation and case-based reasoning in negotiation, using an associated utility function. Argumentation has begun to be applied to agent-based e-commerce [14][30]. We proposed a method to reach an agreement (consensus attainment) based on dialectical logics that are considered a realizatiom of Hegelian dialectics [19][18]. In a word, we would say that argumentation can be viewed as yielding a social computing paradigm for social computation that is a most primary concern in agent-oriented computing [27]. This contrasts with another way to seek new frontiers of computing, natural computation based on natural sciences (e. g., DNA computing, quantum computing, evolutionary computing, etc.).

On the one hand, mobility has been studied on its own as a capability of programs interacting among computer systems. Lange and Oshima [11] describes the seven main benefits of mobile agents: (1) They reduce the network load, (2) they overcome network latency, (3) they encapsulate protocols, (4) they execute asynchronously and autonomously, (5) they adopt dynamically, (6) they are naturally heterogeneous, (7) they are robust and fault-torerant. These benefits emerge mainly from technological observations and considerations. Of course, some people may be suspicious of some of them, claiming that what we can do by introducing mobility can be done without it by the usual, but low-level computer communciation methods. Apart from these technological arguments on mobility, we will turn our eyes to a more fruiteful and profound aspects of mobility and mobile agents.

In this paper, we address ourselves to agents that grow over the computer network through both argumentation among agents and mobile capability. The term "growing" has a meaning similar to recent concepts such as those appearing in learning, evolutionary computing, genetic computing, emergent computing and so on, whose purpose is to realize not behaviour-fixed computing entities but environment-sensitive and metamorphic ones. By "agent growing" of this paper, we mean that agents have acquired knowledge from other agents so that thier arguments on their baffling matters become stronger to rebutting or are refined to more convincing ones. In doing so, we pay attention to human-like activities such as argumentation and mobility since we think agents shoud be anthropomorphic computational objects [28], and we believe these two are most fundamental to agent growing as well as human growing whether it is adult or child. As a matter of fact, argumentation is a way to seek truth by dialogue where truth is not a priori concept in the open and changing networked society. Mobility, on the other hand, is a way to encounter with unexpected agents and their knowledge and ideas. As a special case of mobility, we further go on to a construction of pilgrimaging agents since pilgrimaging is a human's disire and activity common to all over the world whether it is east or west.

The paper is organized as follows. In Section 2, we briefly describe our argumentation framework for arguing agents: knowledge representation and argumentation. In Section 3, we present two approaches to agents that can grow through two fundamental capabilities, arguability and mobility: Agents growing through mobility and knowledge acquisition, and growing agents by conflictive and cooperative argumentation, together with examples. In Section 4, we briefly describe the implementated systems of these mobile arguing agents. Final section includes concluding remarks and future work that need further deep insights.

2 The argumentation framework

A variety of argument models have been proposed with respective features by many researchers, for example, [12][17][10] from the field of computer science, and [2] [8] [26] from the field of philosophy, sociology and linguistics (see [4] for detailed comparisons). So far we also have proposed our own argument models [19][27][30] that have the following features: (1) Knowledge base is distributed and not shared in general, (2) the knowledge base of each agent may be inconsistent, (3) each agent has her/his own argument strategy as a proper way of reasoning, and (4) not only two agents but many agents can be involved in argumentation. These have played important roles in arguing issues in a agents' society over the network.

The purposes of argumentation in agent-oriented computing are mainly threefold in our view:

- to resolve conflicts and choose one from them as an agreement (i)
- to attain a consensus by making better or reconciliatory arguments (ii)
- to uncover hidden issues (iii).

(i) is the main objective of almost all the works cited above as well as of the nonmonotonic logic. Non-monotonic logic, furthermore, is more concerned with maintaining the consistency of knowledge bases, while argumentation is more concerned about argument process and superiority of arguments from logical, verbal and social point of view. For (i), arguments usually proceeds with mutually casting arguments and counterarguments constructed under each agent's knowledge base, and result in 'justified' (sort of 'win') or 'overruled' (sort of 'lose') of the either side.

In addition to the objective (i) of argumentation, we have been further concerned with those aspects of argumentation (ii) and (iii) which can be seen in our ordinary argumentation very often. (ii) was the main objective of our works [19][27][30]. (ii) is important since if an argument has not been settled, it might be better or necessary for the both sides to attain an agreement (consensus) satisfactory to some extent rather than leaving it unsettled. (iii) is the main objective of this paper. if an agent can not make arguments or counter-arguments due to the lack or insufficiency of its knowledge, it can call for cooperation so that he can supply his lacking knowledge from other agents.

2.1 Knowledge representation and argument

Here we introduce some definitions relevant to the objectives of this paper which partly depend on Prakken and Sartor [17]. The object language for knowledge representation is based on the extended logic programming with two negations: $\sim$ (negation as failure) called *weak negation* and $\neg$ (classical negation) called *strong negation*. A strong literal is an atomic formula or a formula preceded by strong negation $\neg$, and a weak literal is a literal of the form $\sim L$, where L is a strong literal. Then a rule has the form of $L_0 \Leftarrow L_1, ..., L_j, \sim L_k, ..., \sim L_n$, where each L_i $(0 \leq i \leq n)$ is an atomic formula or formula negated by $\neg$. The rule is called a *defeasible rule* and has the *assumptions* $\neg L_k$, ..., $\neg L_n$.

An *argument* is a finite sequence, $r_0, ..., r_n$, of ground instances of rules such that (i) for every i $(0 \leq i \leq n)$ and for every strong literal L_j in the antecedent of r_i, there is a $k < i$ such that L_j is the consequent of r_k, and (ii) no two distinct rules in the sequence have the same consequent.

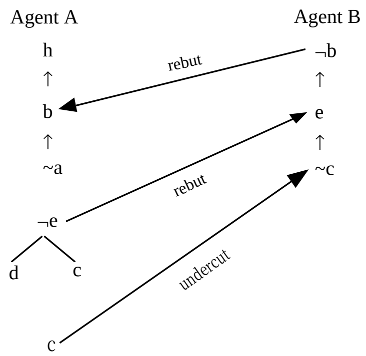
Let A_1 and A_2 be two arguments. Then a consequent L of a rule in A_1 can *attack* A_2 in such a way that: (i) it *rebuts* the consequent $\neg L$ of a rule in A_2 (head conflict), or (ii) it *undercuts* $\sim L$ in the antecedent of a rule in A_2 since this rule has an assumption $\neg L$. An argument is said to be *coherent* if it does not attack itself.

Let A_1 and A_2 be two coherent arguments. Then A_1 *defeats* A_2 iff (i) A_1 undercuts A_2 or (ii) A_1 rebuts A_2 and A_2 does not undercut A_1. A_1 *strictly defeats* A_2 iff A_1 defeats A_2 and A_2 does not defeat A_1.

In this paper, we are concerned with the procedural semantics for the extended logic programs in the top-down manner just like that of the ordinary logic programming language Prolog. This is due to the reason that it allows for a computationally feasible and practical reasoning. Under this semantics, we define two concepts: justified and overruled as follows. An argument A by Agent a is *justified* if each argument defeating A is strictly defeated by the arguments of Agent a. An argument A by Agent a is *overruled* if A is defeated by a justified argument of Agent b. A is *defensible* iff A is neither justified nor overruled. (It is noted that Prakken and Sarter defined them in terms of their fixpoint semantics [17].)

In Figure 1, we illustrate how an argument proceeds along with these definitions, using simple knowledge bases of two agents: $K_a = \{c., d., b \leftarrow\sim a., h \leftarrow b., \neg e \leftarrow d, c.\}$, $K_b = \{e \leftarrow\sim c., \neg b \leftarrow e.\}$. The argument put forward by Agent A is justified in the sense that the counterargument by B is strictly defeated by the arguments of Agent A.

Fig. 1. An example of an argument process



3 Agents with arguability and mobility

In this section, we describe two enterprises to agents that can grow through two fundamental capabilities: arguability and mobility.

3.1 Agents growing through mobility and knowledge acquisition

We consider the following scenario of agents' behaviour on the network. Suppose an agent has an opinion in the form of argument on its issue. However, it is usual for such an argument to be made in terms of uncertain beliefs and knowledge. So, the agent would have a desire to make its argument more convincing one for its own self or its principal. Then the agent starts going out and visits (moves around) agents' places on the network with its own knowledge base and argument (inference) engine, where a number of other agents reside and act for thier own goals with their own belief and knowledge. Through argumentative dialogue, it then tries to make its argument a better one by applying the following subtree replacement transformations:

- Rule replacement for information refinement and diversification
- Fact replacement for information refinement and diversification
- Weak literal replacement for information completion.

Rule replacement allows agents to employ more persuasive or preferable rules in their arguments. Atom replacement allows agents to introduce more evidences to arguments. Weak literal replacement allows agents to reduce incomplete knowledge (belief) included in arguments. As defined in the subsection 2.1, the defeasible rule $L_0 \Leftarrow L_1, ..., L_j, \sim L_k, ..., \sim L_n$ has the *assumptions* $\neg L_k$, ..., $\neg L_n$. This means that agents are allowed to submit arguments without any grounds for those weak literals as can be seen in the definition of arguments. However, we could reinforce arguments or make them better if the assumptions of weak literals were replaced by other agents' arguments with those assumptions as rule heads. This obviously captures one aspect of cooperation offered by other agents. Formally, we state the definition of the weak literal replacement as follows: Let A_1 and A_2 be two arguments by Agent a and Agent b respectively, and let $L_0 \Leftarrow L_1, ..., L_j, \sim L_k, ..., \sim L_n$ in A_1 and $\neg L_i \Leftarrow M_1, ..., M_m$ in A_2 ($k \leq i \leq n$). Then the following is the weak literal replacement scheme to be collaboratively attained by Agent a and Agent b

$$\frac{L_0 \Leftarrow L_1, ..., L_j, \sim L_k, ..., \sim L_i, ..., \sim L_n \qquad \neg L_i \Leftarrow M_1, ..., M_k}{L_0 \Leftarrow L_1, ..., L_j, \sim L_k, ..., M_1, ..., M_k, ..., \sim L_n}$$

For example, an argument $\{dissent(nuke) \leftarrow\sim safe(nuke).\}$ by an agent is refined to a more detailed or persuasive one like $\{accident(Monju)., \neg information-disclosure., \neg safe(nule) \leftarrow accident(Monju), \neg information - disclosure., dissent(nule) \leftarrow \neg safe(nuke).\}$ by another agent.

These transformations are not always accepted for arguments. We introduce the following acceptability conditions under which an agent can accept the offered subarguments from other agents concerned: (i) The agent can neither undercut nor rebut any part of subarguments proposed by other agents. This acceptability condition is important since the agent should keep its knowledge base consistent, (ii) The agent replaces subtrees in arguments if the number of facts as evidences can be increased after the replacement, and
(iii) let *arg* be a subargument to be replaced and *arg'* an argument offered. Then, *arg'* is acceptable if the number of weak literals in *arg'* is less than or equal to the number of weak literals in *arg*.

Agents can learn rules (knowledge) that have been included in the subargument accepted. This may said to be a sort of knowledge acquistion or discovery attained through arguability (communication or dialogue) and mobility. Three transformations are best illustrated by representing arguments in a tree form, as in Figure 2, where readers could see three kinds of replacements introduced above: the subtree with the node b is transformed into a new subtree with the same node b (Rule replacement), the leaf with the weak literal $\sim e$ is expanded to a new subtree with a strong literal $\neg e$ and the evidence k (Weak literal replacement), the fact d is further expanded to a new subtree (Fact replacement). As the results, the subtree with the node c amounts to having been transformed into a bigger subtree that incorporates the above transformations.

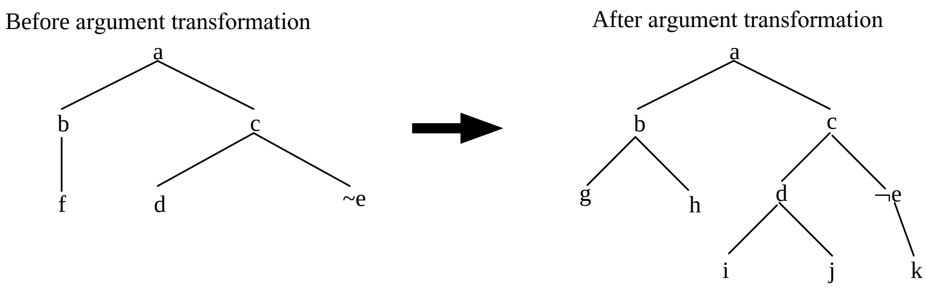


Fig. 2. Argument transformation

After having finished visiting many places and accepting replecements offered from other agents, the agent comes back to its proprietor in the original place. We would say the agent has grown in the sense that it could bring itself to a decided physical and mental state by making its argument more convinced one, consulting other agents' knowledge and wisdom.

It should be noted that the purposes of argument transformation is, in a sense, similar to those of proof transformation or program transformation that can be seen in logic or computer science. The typical ones in logic are the cut elimination in LK, proof normalization in ND and various deductive equivalences between proof architectures. In addition to the argument tranformation introduced above, there could be other useful and versatile directions. They include: (iv) argument transformation based on the concept of similarity (for example, an argument on the issue $p(a)$ is changed into the argument on the issue $p(b)$, using a certain similarity relation $a \sim b$), (v) argument transformation based on the concept of strengthening (or specificity) (for example, an argument on the issue p is changed into the argument on the issue q, where $p \to q$), and (vi) argument transformation based on the concept of weakening (for example, an argument on the issue p is changed into the argument on the issue q, where $q \to p$). But we leave the details of these untouched here for the future work.

3.2 Applications of the growing agents with arguability and mobility

[1] Application to knowledge mining
Figure 3 is a final screenshot of an agent growing through mobility and knowledge acquistion, where the mobile agent has visited 4 agents to confirm his belief on the disapproval of gene-altered crops and foods such as genetically modified corn through argumentation, and finally the agent has come bact to the host 1 with many arguments supporting its issue under a wide variety of persuasive grounds. For example, the agent had these simple arguments on the disapproval of gene-altered crops and foods (gacf) before it went out for traveling:

```
not approve(gacf) <== ~not made(unexpected_substance).
not approve(gacf) <== ~not decrease(nutritive_value)).
```

After the travel, those were refined, completed and diversified as follows (the central window in Figure 3 shows those):

```
made(substance_of_allergy).
made(unexpected_substance) <== made(substance_of_allergy).
not approve(gacf) <== made(unexpected_substance).

not safe(altered_gene).
not safe(produced_protein).
decrease(nutritive_value) <==
   not safe(altered_gene), not safe(produced_protein).
not approve(gacf) <== decrease(nutritive_value).

confuse(ecosystem).
not approve(gacf) <== confuse(ecosystem).

not approve(gacf) <== ~not made(new_toxic_substance).
```

This approach obviously differs from the usual information retrieval and finding by search engines on the internet in the sense that the issue is made clear from scratch in the form of arguments, and the agent's goal is to refine, complete its own arguments and find arguments based on varied grounds. In this model, agents continue to keep their first opinion on the issus without changing their minds while they are visiting and arguing with other agents. In order to capture the dialectic development of scientific controversies such as the pros and cons of gene-altered crops and foods, in a time series, it might be better to introduce other apparatus such as the law of the negation of the negation in Hegelian and Marxist dialectics (interested readers shoud refer to[18]).

Fig. 3. A screenshot of an agent growing through mobility and knowledge acquistion

[2] Application to collaborative theorem proving
We here select a mini proof checker as the focus of a collaborative problem solving by argumentation. We deal with a reasoning task (issue) that arises in distributed environments where no agent has complete knowledge. In fact, the two agents below have incomplete knowledge about the logic respectively although their combined knowledge becomes complete. The issue to cooperatively solve is to examine whether the following sequence of formulas is a proof or not,

```
proof( [(((p -> p) -> (p -> p)) -> (p -> p)) -> (p -> p),
             ((p -> p) -> (p -> p)) -> (p -> p), p -> p] ),
```

under the following knowledge of Agent a and Agent b.

Agent a's knowledge (Agent a might forget the axioms):

```
proof(L) <= proof_checker(L, []).
```

```
proof_checker([], _) <= write('This is a proof').
proof_checker([H|T], Stack) <= ~ not axiom(H),
                            proof_checker(T, [H|Stack]).
proof_checker([H|T], Stack) <= derived(H, Stack),
                            proof_checker(T, [H|Stack]).
derived(G, Stack) <= modus_ponens(E, F, G),
                            member(E, Stack), member(F, Stack).
modus_ponens(A -> B, A, B).
member(X, [X|Xs]).
member(X, [Y|Ys]) :- member(X, Ys).
```

Agent *b*'s knowledge (Agent *b* only knows what the axioms are.):

```
axiom((A -> B) -> ((B -> C) -> (A -> C))).
axiom(((A -> A) -> B) -> B).
axiom((A -> (A -> B)) -> (A -> B)).
```

Agent *a* can ascertain that it is a proof under her/his belief on the axioms although s/he forgets them, for two reasons. First, the weak literal ∼not axiom(H) is used in Agent *a*'s knowledge, and this means that Agent *a* believes axiom(H) without its verification, in other words, axiom(H) is Agent *a*'s assumption by default (see Figure 4). Second, there is no counterargument from Agent *b* for Agent *a*'s proposed argument (proof). Then Agent *b* might feel inclined to make a better proof and cooperate with Agent *a*, persuading Agent *a* to guarantee that it is certainly a proof if Agent *b*'s knowledge is provided (see Figure 5).

Interestingly, if the issue is something incorrect like

```
proof( [(((p -> p) -> (p -> p)) -> (p -> p)) -> (p -> p), q, p -> p] ),
```

the agents could have a proof corrected by the process of argument and collaboration.

We have described the cases where the reasoning leads to a success whether it is a true proof or putative one. In the case of the argument' failures that actually occur very often, we have not given an appropriate idea, but this can be viewed as sort of distributed problem solving such as distributed automated reasoning [13].

Figure 6 is a snapshot of the collaborative proof construction. The completed proof is reported in the lowest window of the host 1.

3.3 Growing agents by conflictive and cooperative argumentation

In this subsection, we consider the following scenario of a mobile arguing agent. We assume such a virtual society that both antagonists and protagonists reside. Then, it starts roaming around the network to resolve its baffling matter, expecting to meet those two kinds of agents and argue about it. Firstly the mobile arguing agent begins with arguing with one of antagonists who always argue

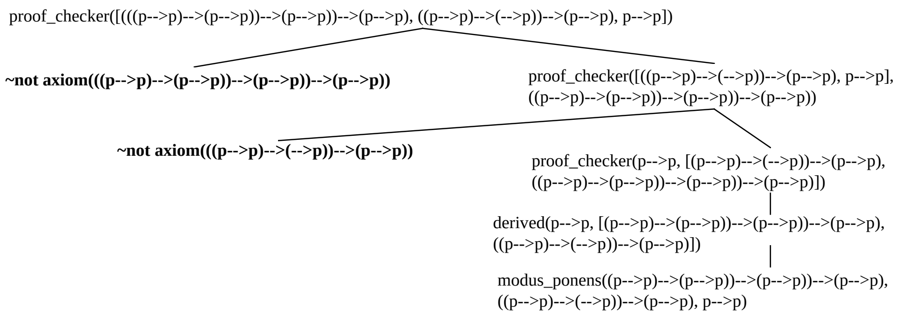


Fig. 4. Argument tree before collaboration

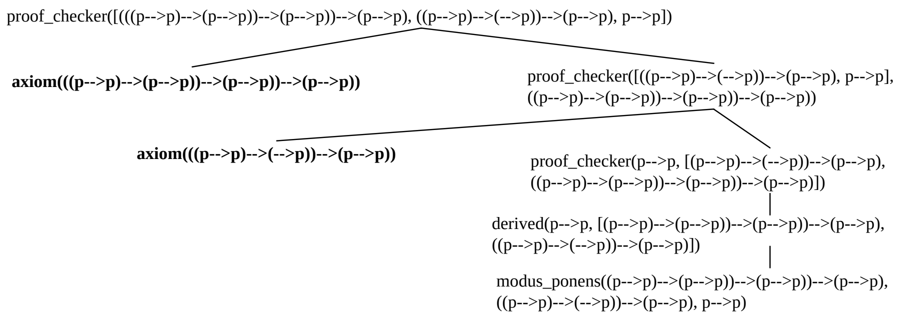


Fig. 5. Argument tree after collaboration

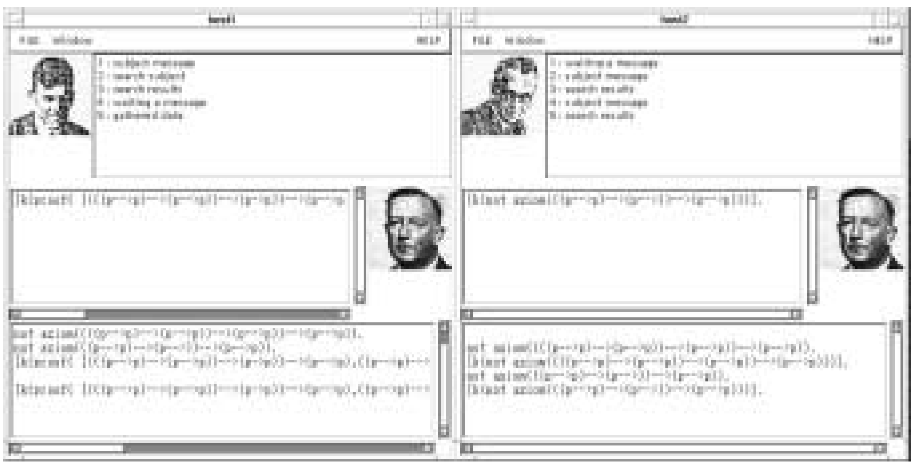


Fig. 6. A snapshot of a collaborative proof construction

against about the agent's baffling matter. In this argument, if the mobile arguing agent wins, then it goes on its roaming, seeking other antagonists. Otherwise, it learns and acquires the knowledge which the antagonist used in arguing against the the agent's matter, and then it visits one of protagonist to confirm how good this opposing argument is. This is done by arguing with the protagonist, using the opposing argument. Then, if the protagonist wins, then the mobile arguing agent brings this winning argument and revists the next antagonist to repeat the same procedure as the former antagonist. Otherwise, the mobile arguing agent will visit the next protagonist to confirm how good that opposing argument is. These processes are repeted until there are no antagonists and protagonists to visit. We illustrate an example of the agent migration flow in Figure 7, where the mobile arguing agent, "agent Bee" generated by the host agent "agent P1", meets two antagonists and two protagonists over the network during the migration, and finally returns to the origical place and reports to the principal agent, Agent P1, the final knowledge base that has been obtained and considered a form of growing or evolution. Figure 8 is a snapshot of this agent migration.

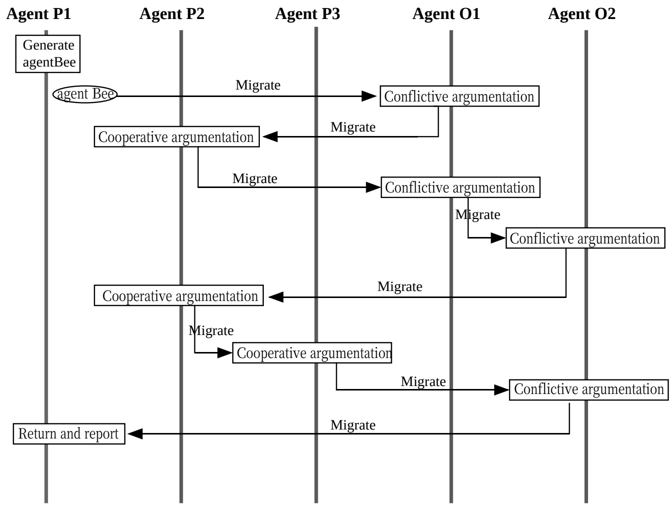


Fig. 7. An example of an agent migration flow

In this scenario, we would say that both arguability and mobility yield growth to agents, by traveling and meeting both kinds of agents. This style of meeting is important for agent growing since meeting only one of either antagonists or protagonists tends to lead to a biased formation of knowledge base, and hence not good for agent growing as well as human growing.

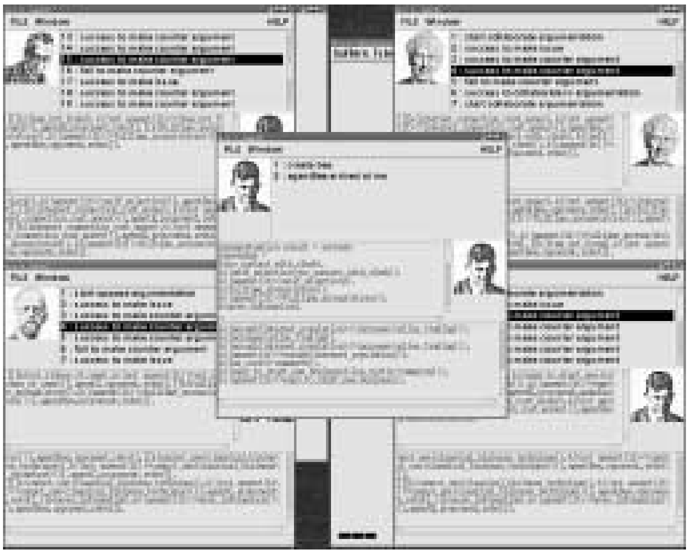


Fig. 8. A snapshot of a growing agent by conflictive and cooperative argumentation

3.4 Towards agents roaming for spiritual enlightenment

There are a number of discussions about why we need to migrate programs or agents. Turning our eyes to our actual lives, the similar questions may arise, such as why we make a travel or go on a pilgrimage. What could be brought to us by roaming, traveling, pilgrimaging or whatever in our lives or on the network? It must be no doubt true that roaming, traveling, or pilgrimaging capability as a form of mobility is not only a fun but also an important activity since we can expect that we might enhance and accelerate the attainment with it at least in our real life. Pilgrimage invites you to travel with your heart, guided toward an inner goal. [29] and [5], for example, invite you to make a pilgrimage, and then state that travel with a pilgrim heart which is an open heart should be receptive to change and spiritual growth .

On the other hand, argument usually takes a form of dialogue or polylogue. As a special case, argumentation by monologue also awakens our interest since monologue may be considered an interaction in a self. This might be sort of an inner mental process and inner journey to mind of the human as well as agents, in which they grow and enrich their inner mind.

As a next step of our mobile arguing agents, we just have started studying a reflective and meditative agent who might be expected to reach enlightenment in Buddhism and Taoism. Here, reaching enlightenment is said to be attained by 'gedatsu' in Japanese, nirvana or vimukti in Sanskrit that may be interpreted

as an eastern counterpart of Aufheben in Western dialectics [21]. In this section, we will attempt to depict how an agent makes a pilgrimage in a cyberspace to get a spiritual satisfaction or enlightenment by purifying its own beliefs through arguing with other agents or its self. The following is one scenario:
1. The agent starts on a journey with issues to resolved or understood.
2. The agent exchanges questions and answers with other agents it has visited, for each issue.
3. If the issues are justified, then the agent keeps its knowledge used in its arguments. Otherwise, it is to be gifted with wisdom in the form of the other party's knowledge used in their arguments.
4. When the issues are justified, the agent tries to make its arguments better or deeper by begging the other party for knowledge, in such several ways as (a) asking cooperation for facts, weak negations and subtrees, (b) asking to supply for lacking knowledge [25], (c) trying to make dialectical agreements and understanding [19].

After having visited many places where other agents reside and exchanged questions and answers (like mondos in Japanese Zen or koans in Chinese Zen), agents could get to the new states of knowledge. They are sort of the great truth about life dawned upon them. Like this, agents could be spiritually awakened (spiritual enlightenment, Buddhahood) in the sense of knowledge evolution and purification by monologue or dialogue. Incorporating the role of emotion, one quality of humanity [1], into spiritual growth of pilgimaging agents, must not be forgotten. But we leave it as another problem to be taken into account for future research.

4 Implementation of the mobile arguing agents

The agents with arguability and mobility described in Section 3 have been implemented within the Bee-gent^{TM} (Bonding and Encapsulation Enhancement aGENT) agent framework [24]. The Bee-gent framework is comprised of two types of agent: Agent Wrappers and Mediation Agents. Agent Wrappers allow us to agentify existing applications, while Mediation Agents support inter-application co-ordination by handling all communications. The mediation agents move from the site of an application to another where they interact with the agent wrappers. The messages are communicated in terms of XML/ACL in the Bee-gent. Using the class libraries for Agent Wrappers and Mediation Agents, the part of the communication control among agents has been implemented in Java, and the part of the construction of arguments and counterarguments in Prolog. Mobile arguing agents are realized as mediation agents with the inference (argumentation) engine and knowledge base.

5 Concluding remarks and future work

Exploring a new frontier of mobile agents, we have attempted to construct two kinds of agents that can grow in our sense by means of arguability and mobility.

Growth, argumentation and mobility are originally independent concepts. However, we showed that the latter two are closely and deeply involved in the growth of computational objects with knowledge, agents. We believe that what we have done could yield a wedge of more fruitful growth mechanism for knowledge-based agents in the future, and the illustrative examples showed its potential usefulness. Our results on mobile agents with arguability are only preliminary buds to further development in the future, and hence might not be easy to evaluate them throughly at this stage. But, it is worthy to note some possibilities to suggest some practical usefulness of this work. At the moment we would like to point out that it might lead to the following contemoporary or future-oriented research areas: (a) agents that mine data and knowledge instead of principals, (b) agents that put information together and summarize, and (c) agents that philosopize, deepen and understand our reality, thought and world, etc.

There are many important directions to be pursued further. We will touch upon some of them.

Incorporation of knowledge assimilation Agents of this paper are simply to perform knowledge acquisition as far as knowledge to be acquired is not rebut. Knowledge assimilation is the next step to be pursued

The extended paraconsistent logic programming language The paraconsistent logic programming language is more relevant as a knowledge representation language than the extended logic programming language since it is inconsistency-tolerant [3].

References

1. Bates, J.: *The Role of Emotion in Believable Agents*, CACM, Vol. 37, No.7, pp. 122-125, 1994.
2. Benthem, J. van et al. (eds.): *Logic and Argumentation*, Proc. of the Colloquium 'Logic and Argumentation', Royal Netherlands Academy of Arts and Sciences, 1994.
3. Blair, A. H. and Subrahmanian, V. S.: *Paraconsistent Logic Programming*, Theoretical Computer Science, Vol. 68, pp. 135-154, 1989.
4. Chesñevar, C. I., Maguitman, A. G. and Loui, R. P. (2000). *Logical Models of Argument.* ACM Computing Surveys 32(4), 337-383.
5. Cousineau, P.: *The Art of Pilgrimage*, Conari Press, 1998.
6. Conklin, J. and Begeman, M. L.: *gIBIS: A Hypertext Tool for Exploratory Policy Discussion*, ACM Transactions on Office Information Systems, Vol. 6, No. 4, pp. 1303-331, 1988.
7. Dung, P. M.: *An Argumentation-Theoretic Foundation for Logic Programming with Classical Negation*, The Journal of Logic Programming, pp. 151-177, 1995.
8. Eemeren, F. H., Grootendorst, R., Henkmans, A. F. S., et al.: *Fundamentals of Argumentation Theory, A Handbook of Historical Backgrounds and Contemporary Developments*, Lawrence Erlbaum, 1996.
9. Krause, P., Ambler, S., Goransson, M. E. and Fox, J.: *A Logic of Argumentation for Reasoning under Uncertainty*, Computational Intelligence, Vol. 11, No. 1, pp. 113-131, 1995.

10. Kraus, S., Sycara, K. and Evenchik, A.: *Reaching Agreements through Argumentation: a Logical Model and Implementation*, Artificial Intelligence, Vol. 104, pp. 1-69, 1998.
11. Lange, D. B. and Oshima, M.: *Seven Good Reasons for Mobile Agents*, CACM, Vol. 41, No. 3, pp. 88-89, 1999.
12. Loui, R. P.: *Defeat among Arguments: a System of Defeasible Inference*, Computational Intelligence, Vol.2, pp. 100-106, 1987.
13. Macintosh, D. J. and Conry, S. E. and Meyer, R. A.: *Distributed Automated Reasoning: Issues in Coordination, Cooperation, and Performance*, IEEE Transactions on Systems, Man and Cybernetics, Vol. 21, No. 6, pp. 1307-1316, 1991.
14. Maes, P., Guttman, R. H. and Moukas, A. G.: *Agents that Buy and Sell*, Communications of ACM, Vol. 42, No. 3, pp. 81-91, 1999.
15. Noriega, P. and Sierra, C.: *Agent Mediated Electronic Commerce*, Lecture Notes in Artificial Intelligence 1571, 1998.
16. Parsons, S., Sierra, C. and Jennings, N.: *Agents that Reason and Negotiate by Arguing*, J. of Logic and Computation, Vol. 8, No. 3, pp. 261-292, 1998.
17. Prakken, H. and Sartor, G.: *Argument-based Extended Logic Programming with Defeasible Priorities*, J. of Applied Non-Classical Logics, Vol. 7, No. 1-2, pp. 25-75, 1997.
18. Sawamura, H.: *A Computational Challenge to Hegelian and Marxist Dialectics for Agents' World*, to be presented at PCL2002, 2002.
19. Sawamura, H., Umeda, Y. and Meyer, R. K.: *Computational Dialectics for Argument-based Agent Systems*, Proc. of the Fourth International Conference on MultiAgents Systems (ICMAS2000), IEEE Computer Society, pp. 271-278, 2000.
20. Buckingham, S. S.: http://kmi.open.ac.uk/people/sbs/csca/.
21. Sueki,T.: *Eastern Rationalism*, Houzoukan, 2001 (in Japanese).
22. Sycara, K. P.: *Persuasive Argumentation in Negotiation*, Theory and Decision, Vol. 28, pp. 203-242, 1990.
23. Tambe, M. and Jung, H.: *The Benefits of Arguing in a Team*, AI Magazine, pp. 85-92, 1999.
24. Toshiba Corporation: *Bee-gent*, http://www.toshiba.co.jp/beegent/, 1999-2001.
25. Toda, Y., Yamashita, M. and Sawamura, H. : *An Argument-based Agent System with KQML as an Agent Communication Language*, Proc. of 4th Pacific International Workshop on Multi-agents (PRIMA'2001), Lecture Notes in Artificial Intelligence, Vol. 2132, pp. 48-62, 2001.
26. Toulmin, S.: *The Uses of Argument*, Cambridge Univ. Press, 1958.
27. Umeda, Y., Yamashita, M., Inagaki, M. and Sawamura, H.: *Argumentation as a Social Computing Paradigm*, Proc. of 3rd Pacific RIM Int. Workshop on Multi-Agents (PRIMA'2000), Lecture Notes in Artificial Intelligence, Vol. 1881, pp. 46-60, 2000.
28. Wooldridge, M. and Jennings, N. R.: *Agent Theories, Architectures, and Languages: A survey*, Lecture Notes in Artificial Intelligence, Vol. 890, pp. 1-39, 1994.
29. York, S.: *Pilgrim Heart*, Jossey-Bass, A Wiley Company, 2001.
30. Yamashita, M. and Sawamura, H.: *Reaching an Undrestanding through Dialectical Argumentation*, Proc. of the 2001 International Conference on Artificial Intelligence (AI-2001), CSREA Press, pp. 1164-1170, 2001.