Asynchronous Argumentation with Pervasive Personal Communication Tools

Yuki Katsura[1], Hajime Sawamura[2], Takeshi Hagiwara[2] and Jacques Riche[3]
[1]*Graduate School of Science and Technology, Niigata University, Niigata, Japan*
[2]*Institute of Science and Technology, Niigata University, Niigata, Japan*
[3]*Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium*

Keywords: Asynchronous Argumentation, Multiple-valued Argumentation, iPad, PIRIKA.

Abstract: In this paper, we propose an argument-based communication tool for humans and agents, which supplements and alternates the current communication system such as Twitter, Line, etc. in order to allow us to make a more deliberate and logical human communication. For this purpose, we devised asynchronous argumentation based on our logic of multiple-valued argumentation. It may be as well reworded as asymptotic or incremental argumentation since agents could approach towards truth or justification every time argument is put forward by an agent. We have made real the asynchronous argumentation system, named PIRIKA (pilot of the right knowledge and argument), on the pervasive personal tool, iPad. Finally some lessons learned from the experimental uses of PIRIKA are reported.

1 INTRODUCTION AND MOTIVATION

In this paper, we propose an argument-based communication tool for humans and agents, which supplements and alternates the current communication systems such as Twitter[1], Line[2], etc. in order to allow us to make a more deliberate and logical human communication. This is an attempt to make a clear departure from surface communication with a few words toward deep communication with argumentation which emphasizes the relationship of a conclusion with reasons all the time.

Social networking software is reshaping the world we live in. In these days, much popularity has been seen in communication tools on the Internet that link people and organizations, instead of linking documents only. To mention a few, Skype[3] (for direct communication), Line and Mail (for asynchronous communication), Twitter and niconico[4] (for asynchronous communication with general public), etc.

[1]Twitter is a registered trademark of Twitter, Inc..

[2]Line is a trademark of Line company.

[3]Skype is a registered trademark or a trademark of Skype Limited.

[4]niconico is a registered trademark or a trademark of Dwango company.

Among other things, Line is an instant messaging application on smartphones and PCs. It, launched in Japan in 2011, now reached 300 million users over a short amount of time in the world. The main features of Line seems to be twofold. One is the so-called *mere-exposure effect*, which is a psychological phenomenon by which people tend to develop a preference for things merely because they are familiar with them. In social psychology, this effect is sometimes called the familiarity principle. In studies of interpersonal attraction, the more often a person is seen by someone, the more pleasing and likeable that person appears to be. The other is the so-called *elevator pitch*, which is a short summary used to quickly and simply define a person, profession, product, service, organization or event and so on. With LINE, people nowadays tend to communicate with each other in short messages very often. Short and Quick are keys there, revealing a light or surface communication.

On the other hand, electronic mail which is relatively surface communication tool is a method of exchanging digital messages from an author to one or more recipients. It has become the most widely used medium of communication not only within the business world but also in our daily lives although it has some disadvantages such as loss of context, information overload, speed of correspondence and so on. But from the viewpoint of communication style,

Katsura Y., Sawamura H., Hagiwara T. and Riche J..
Asynchronous Argumentation with Pervasive Personal Communication Tools.
DOI: 10.5220/0004751101050114
In *Proceedings of the 6th International Conference on Agents and Artificial Intelligence* (ICAART-2014), pages 105-114
ISBN: 978-989-758-016-1

email solves two basic problems of communication: logistics and synchronization.

The problem of logistics: Much of the business world relies upon communications between people who are not physically in the same building, area or even country; setting up and attending an in-person meeting, or telephone call can be inconvenient, time-consuming, and costly. Email provides a way to exchange information between two or more people with no set-up costs.

The problem of synchronization: With real time communication by meetings or phone calls, participants have to work on the same schedule, and each participant must spend the same amount of time in the meeting or call. Email allows asynchrony; each participant may control their own schedule independently.

In contrast with email, there can be two ways of use of argumentation: synchronous and asynchronous. Argumentation is usually held in such a synchronous way that participants gather in the same time and place. The asynchronous argumentation we advocate in this paper can solve the same synchronization problem as that of email above, but with taking deep communication into account all the time.

The asynchronous argumentation we intend can be seen in the flow of argumentation. Let us consider a look-and-feel scenario of pervasive arguing agents we aim at realizing on top of the pervasive personal tools such as iPad and iPhone[5]. Suppose that there are agents who have gathered knowledge on an issue concerned with on a routine basis, and conceived their own arguments on it (asynchronous preparation for argumentation). Then, the knowledge gathering may be done by humans or helped by e-secretaries who might reside in pervasive personal tools as avatars. Someday, an agent may wish to know such a collective view as what the present voices of the people around it are like and how they can be converged to a popular opinion. For example, amendment to the constitution, increase in consumption tax, etc. would be keen interest to people in any country. Then, the agent can start argumentation to know the result on an issue which it has been concerned about, using the arguing agent on the pervasive personal tool.

The argument participants will be general public who now connect to the argument server. But they could obtain argument results which are not assertions of opinions only but lines of reasoning leading from some premises to a conclusion. It should be noted that this is a kind of non-monotonic phenomenon of reasoning realized by argumentation. Actually, conclusions, once drawn, may later be withdrawn after a new agent will have come on argumentation scene or stage with additional information. In this manner, arguing agents on the pervasive personal tools could produce a well-reasoned judgment (warranted assertion), construed as a form of inquiry conducted conjoinedly and asynchronously.

In this paper, we describe a realization of asynchronous argumentation which allows such a scenario of futuristic communication on pervasive personal tools. The paper is organized as follows. In the following sections 2 and 3, we briefly introduce part of EALP (Extended Annotated Logic Programming) and LMA (Logic of Multiple-valued Argumentation)(Takahashi and Sawamura, 2004) to make the paper self-contained. They are an underlying logic for practicing the asynchronous argumentation on iPad. In Section 4, we illustrate a series of use of PIRIKA on top of iPad which allows for asynchronous argumentation, using typical screenshots appearing in the argument process. Section 5 summarizes advantageous points of our work as an evaluation, which we have confirmed from participants in experimental and daily uses. The final section includes conclusion and future work.

[5]iPad and iPhone are trademarks of Apple Inc.

2 OVERVIEW OF EALP

EALP is an underlying knowledge representation language that we formalized for our logic of multiple-valued argumentation LMA. EALP has two kinds of explicit negation: epistemic explicit negation '$\neg$' and ontological explicit negation '$\sim$', and the default negation '**not**'. Intuitively, $\sim$ is almost the same as the classical negation, **not** the negation-failure as in Prolog, and $\neg$ a negation based on our epistemology. They are supposed to yield a momentum or driving force for argumentation or dialogue in LMA. In what follows, we describe an outline of EALP.

2.1 Language

Definition 1 (**Annotation as Truth-values and Annotated Atoms** (Kifer and Subrahmanian, 1992)). We assume a complete lattice $(\mathcal{T}, \leq)$ of truth values, and denote its least and greatest element by $\bot$ and $\top$ respectively. The least upper bound operator is denoted by $\sqcup$. An annotation is either an element of $\mathcal{T}$ (constant annotation), or an annotation variable on $\mathcal{T}$. If A is an atomic formula and μ is an annotation, then $A:\mu$ is an annotated atom. We assume an annotation function $\neg : \mathcal{T} \rightarrow \mathcal{T}$, and define that $\neg(A:\mu) = A:(\neg\mu)$. $\neg A:\mu$ is called the epistemic explicit negation(e-explicit negation) of $A:\mu$.

Definition 2 (**Annotated Literals**). Let $A:\mu$ be an annotated atom. Then $\sim(A:\mu)$ is the ontological explicit negation (o-explicit negation) of $A:\mu$ (we simply write $\sim(A:\mu)$ as $\sim A:\mu$ when no confusion arises). An annotated objective literal is either $\sim A:\mu$ or $A:\mu$. The symbol $\sim$ is also used to denote complementary annotated objective literals. Thus $\sim\sim A:\mu = A:\mu$. If L is an annotated objective literal, then $\mathbf{not}\,L$ is a default negation of L, and called an annotated default literal. An annotated literal is either of the form $\mathbf{not}\,L$ or L.

Definition 3 (**Extended Annotated Logic Programs (EALP)**). An extended annotated logic program (*EALP*) is a set of annotated rules of the forms: $H \leftarrow L_1 \& \ldots \& L_n$, or H, where H is an annotated objective literal, and L_i $(1 \leq i \leq n)$ are annotated literals.

The head of a rule is called a *conclusion* of a rule. Annotated objective literals and annotated default literals in the body of the rule are called *antecedents* of the rule and *assumptions* of the rule respectively. For simplicity, we assume that a rule with annotation variables or objective variables represents every ground instance of it. We identify a distributed EALP with an *agent*, and treat a set of EALPs as a *multi-agent system*.

2.2 Interpretation

Definition 4 (**Extended Annotated Herbrand Base**). The set of all annotated literals constructed from an EALP P on a complete lattice $\mathcal{T}$ of truth values is called the extended annotated Herbrand base $H_P^{\mathcal{T}}$.

Definition 5 (**Interpretation**). Let $\mathcal{T}$ be a complete lattice of truth values, and P be an EALP. Then, the interpretation on P is the subset $I \subseteq H_P^{\mathcal{T}}$ of the extended annotated Herbrand base $H_P^{\mathcal{T}}$ of P such that for any annotated atom A,

1. If $A:\mu \in I$ and $\rho \leq \mu$, then $A:\rho \in I$ (downward heredity);
2. If $A:\mu \in I$ and $A:\rho \in I$, then $A:(\mu \sqcup \rho) \in I$ (tolerance of difference);
3. If $\sim A:\mu \in I$ and $\rho \geq \mu$, then $\sim A:\rho \in I$ (upward heredity).

The conditions 1 and 2 of Definition 5 reflect the definition of the ideal of a complete lattice of truth values. The ideals-based semantics was first introduced for the interpretation of GAP by Kifer and Subrahmanian (Kifer and Subrahmanian, 1992). Our EALP for argumentation also employs this since it was shown that the general semantics with ideals is more adequate than the restricted one simply with a complete lattice of truth values (Takahashi and Sawamura, 2004). We define three notions of inconsistencies corresponding to three concepts of negation in EALP.

Definition 6 (**Inconsistency**). Let I be an interpretation. Then,

1. $A:\mu \in I$ and $\neg A:\mu \in I \iff I$ is epistemologically inconsistent (e-inconsistent).
2. $A:\mu \in I$ and $\sim A:\mu \in I \iff I$ is ontologically inconsistent (o-inconsistent).
3. $A:\mu \in I$ and $\mathbf{not}A:\mu \in I$, or $\sim A:\mu \in I$ and $\mathbf{not} \sim A:\mu \in I \iff I$ is inconsistent in default (d-inconsistent).

When an interpretation I is o-inconsistent or d-inconsistent, we simply say I is *inconsistent*. We do not see the e-inconsistency as a problematic inconsistency since by the condition 2 of Definition 5, $A:\mu \in I$ and $\neg A:\mu = A:\neg\mu \in I$ imply $A:(\mu \sqcup \neg\mu) \in I$ and we think $A:\mu$ and $\neg A:\mu$ are an acceptable differential. Let I be an interpretation such that $\sim A:\mu \in I$. By the condition 1 of Definition 5, for any ρ such that $\rho \geq \mu$, if $A:\rho \in I$ then I is o-inconsistent. In other words, $\sim A:\mu$ rejects all recognitions ρ such that $\rho \geq \mu$ about A. This is the underlying reason for adopting the condition 3 of Definition 5. These notions of inconsistency yield a logical basis of attack relations described in the multiple-valued argumentation of the next section.

Definition 7 (**Satisfaction**). Let I be an interpretation. For any annotated objective literal H and annotated literal L and L_i, we define the satisfaction relation denoted by '$\models$' as follows.

- $I \models L \iff L \in I$
- $I \models L_1 \& \cdots \& L_n \iff I \models L_1, \ldots, I \models L_n$
- $I \models H \leftarrow L_1 \& \cdots \& L_n \iff I \models H$ or $I \not\models L_1 \& \cdots \& L_n$.

3 OVERVIEW OF LMA

In formalizing logic of argumentation, the most primary concern is the rebuttal relation among arguments since it yields a cause or a momentum of argumentation. The rebuttal relation for two-valued argument models is most simple, so that it merely appears between the contradictory propositions of the form A and $\neg A$. In case of multiple-valued argumentation based on EALP, much complication is to be involved into the rebuttal relation under the different concepts of negation. One of the questions arising from multiple-valuedness is, for example, how a literal with truth-value ρ confronts with a literal with truth-value μ in the involvement with negation. In the next subsection, we outline important notions proper to logic of the multiple-valued argumentation LMA in which the above question is reasonably solved.

3.1 Annotated Arguments

Definition 8 (**Reductant and Minimal Reductant**). Suppose P is an EALP, and C_i $(1 \leq i \leq k)$ are annotated rules in P of the form: $A:\rho_i \leftarrow L_1^i \,\&\, \ldots \,\&\, L_{n_i}^i$, in which A is an atom. Let $\rho = \sqcup\{\rho_1,\ldots,\rho_k\}$. Then the following annotated rule is a reductant of P.
$A:\rho \leftarrow L_1^1 \,\&\, \ldots \,\&\, L_{n_1}^1 \,\&\, \ldots \,\&\, L_1^k \,\&\, \ldots \,\&\, L_{n_k}^k$.
A reductant is called a minimal reductant when there does not exist non-empty proper subset $S \subset \{\rho_1,\ldots,\rho_k\}$ such that $\rho = \sqcup S$

Definition 9 (**Annotated Arguments**). Let P be an EALP. An annotated argument in P is a finite sequence $Arg = [r_1,\ldots,r_n]$ of rules in P such that for every i $(1 \leq i \leq n)$,

1. r_i is either a rule in P or a minimal reductant in P.
2. For every annotated atom $A:\mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $A:\rho$ $(\rho \geq \mu)$ is head of r_k.
3. For every o-explicit negation $\sim A:\mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $\sim A:\rho$ $(\rho \leq \mu)$ is head of r_k.
4. There exists no proper subsequence of $[r_1,\ldots,r_n]$ which meets from the first to the third conditions, and includes r_1.

We denote the set of all arguments in P by $Args_P$, and define the set of all arguments in a set of EALPs $MAS = \{KB_1,\ldots,KB_n\}$ by $Args_{MAS} = Args_{KB_1} \cup \cdots \cup Args_{KB_n}$ $(\subseteq Args_{KB_1 \cup \cdots \cup KB_n})$. This means that each agent has its own knowledge base and do not know other agent's ones before starting arguments. This is a natural assumption for argument settings, differently from other argumentation models (Rahwan and Simari, 2009).

3.2 Attack Relation

The semantics of the argumentation depends on what sort of attack relation is considered to deal with conflicts among arguments. It would be reasonable to think that conflicts among arguments occur when the interpretation satisfying a set of arguments is inconsistent.

Definition 10 (**Rebut**). Arg_1 rebuts $Arg_2 \iff$ there exists $A:\mu_1 \in concl(Arg_1)$ and $\sim A:\mu_2 \in concl(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A:\mu_1 \in concl(Arg_1)$ and $A:\mu_2 \in concl(Arg_2)$ such that $\mu_1 \leq \mu_2$.

Definition 11 (**Undercut**). Arg_1 undercuts $Arg_2 \iff$ there exists $A:\mu_1 \in concl(Arg_1)$ and $\mathbf{not} A:\mu_2 \in assm(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A:\mu_1 \in concl(Arg_1)$ and $\mathbf{not} \sim A:\mu_2 \in assm(Arg_2)$ such that $\mu_1 \leq \mu_2$.

Definition 12 (**Strictly Undercut**). Arg_1 strictly undercuts $Arg_2 \iff Arg_1$ undercuts Arg_2 and Arg_2 does not undercut Arg_1.

We also define the combined attack relation associated with o-inconsistency and d-inconsistency.

Definition 13 (**Defeat**). Arg_1 defeats $Arg_2 \iff Arg_1$ undercuts Arg_2, or Arg_1 rebuts Arg_2 and Arg_2 does not undercut Arg_1.

When an argument defeats itself, such an argument is called a *self-defeating argument*. For example, $[p:\mathbf{t} \leftarrow \mathbf{not}\, p:\mathbf{t}]$ and $[q:\mathbf{f} \leftarrow \sim q:\mathbf{f},\ \sim q:\mathbf{f}]$ are all self-defeating. In this paper, however, we rule out self-defeating arguments from argument sets since they are in a sense abnormal, and not entitled to participate in argumentation or dialogue.

In this paper, we employ defeat and strictly undercut to specify the set of justified arguments where d stands for defeat and su for strictly undercut.

Definition 14 (**Acceptable and Justified Argument** (Dung, 1995)). Suppose $Arg_1 \in Args$ and $S \subseteq Args$. Then Arg_1 is acceptable wrt. S if for every $Arg_2 \in Args$ such that $(Arg_2, Arg_1) \in d$ there exists $Arg_3 \in S$ such that $(Arg_3, Arg_2) \in su$. The function $F_{Args,d/su}$ mapping from $\mathcal{P}(Args)$ to $\mathcal{P}(Args)$ is defined by $F_{Args,d/su}(S) = \{Arg \in Args \mid Arg$ is acceptable wrt. $S\}$. We denote a least fixpoint of $F_{Args,d/su}$ by $J_{Args,d/su}$. An argument Arg is justified if $Arg \in J_{d/su}$; an argument is overruled if it is attacked by a justified argument; and an argument is defensible if it is neither justified nor overruled.

Since $F_{x/y}$ is monotonic, it has a least fixpoint, and can be constructed by the iterative method (Dung, 1995).

Justified arguments can be dialectically determined from a set of arguments by the dialectical proof theory.

Definition 15 (**Dialogue** (Prakken and Sartor, 1997)). An dialogue is a finite nonempty sequence of moves $move_i = (Player_i, Arg_i)$, $(i \geq 1)$ such that

1. $Player_i = P$ $(Proponent) \iff i$ is odd; and $Player_i = O$ $(Opponent) \iff i$ is even.
2. If $Player_i = Player_j = P$ $(i \neq j)$ then $Arg_i \neq Arg_j$.
3. If $Player_i = P$ $(i \geq 3)$ then $(Arg_i, Arg_{i-1}) \in su$; and if $Player_i = O$ $(i \geq 2)$ then $(Arg_i, Arg_{i-1}) \in d$.

In this definition, it is permitted that $P = O$, that is, a dialogue is done by only one agent. Then, we say such an argument is a self-argument (monologue).

Definition 16 (**Dialogue Tree** (Prakken and Sartor, 1997)). A dialogue tree is a tree of moves such that every branch is a dialogue, and for all moves $move_i =$

(P, Arg_i), the children of $move_i$ are all those moves (O, Arg_j) $(j \geq 1)$ such that $(Arg_j, Arg_i) \in d$.

Definition 17 (**Provably** x/y**-justified**). Let x be d(efeat) and y su(strictly undercut). An x/y-dialogue D is a winning x/y-dialogue $\iff$ the termination of D is a move of proponent. An x/y-dialogue tree T is a winning x/y-dialogue tree $\iff$ every branch of T is a winning x/y-dialogue. An argument Arg is a provably x/y-justified argument $\iff$ there exists a winning x/y-dialogue tree with Arg as its root.

We have the sound and complete dialectical proof theory for the argumentation semantics $J_{Args,x/y}$ (Takahashi and Sawamura, 2004).

4 ASYNCHRONOUS ARGUMENTATION

Our former PIRIKA (Tannai et al., 2013) is a synchronous argumentation in the sense that every agent who wants to participate in argumentation prepare its own knowledge base once prior to argumentation. Then, it produces an outcome of argumentation and displays it, as in many argumentation systems developed so far (Rahwan and Simari, 2009). In this section, we turn such a synchronous argumentation to a more flexible asynchronous one, by redesigning PIRIKA on the top of the pervasive personal communication tool, iPad, toward a new futuristic communication tool.

In this section, we illustrate a series of use of PIRIKA on top of iPad which allows for asynchronous argumentation, like a live argument using typical screenshots appearing in the argument process. This is because PIRIKA is the first implementation of the argumentation system on pervasive personal information equipments as far as we know, and we think that in order for readers to understand both our argumentation process and its realization on iPad, it would be necessary to describe the overall story of PIRIKA step by step from beginning to end without omitting any details, even if it contains one of the standard display of iPad.

We take up a so-called schedule management problem which is a typical target for which agent systems have been developing their capabilities such as interaction, negotiation, cooperation and so on. We demonstrate a new approach to realizing the schedule management system by the asynchronous argumentation on iPad. Then participating agents use not only calendar information but also preferential knowledge base of their own in EALP. Following Definition 1in Section2, the annotation employed is a complete lattice of the power set of the monthly

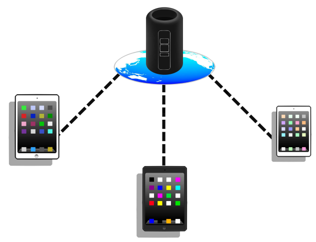

Figure 1: Asynchronous argumentation system PIRIKA on iPad as a client-server system.

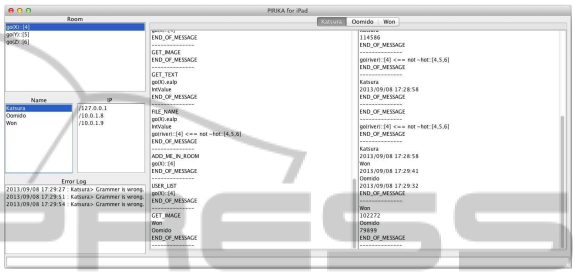

Figure 2: Screenshot of the PIRIKA server.

dates, i.e., $\langle \mathcal{P}(\{1, \ldots, 31\}), \subseteq \rangle$ with the set inclusion $\subseteq$ as the lattice ordering. This type of annotation may be somewhat deviant, but allows for representing temporal information, and hence works well conveniently with the schedule management problem, such as $visit(niigata) : \{3, 4\}$ representing 'we visit Niigata on 3rd and 4th'. Furthermore, PIRKA allows for representing and inquiring indefinite issues such as questions or problems satisfying certain conditions such as $visit(X) : Y$, where X and Y are variables. Such an expressivity is a desideratum particularly for the schedule management system since we naturally inquire 'When and where we should visit?' for schedule coordination (Oomidou et al., 2013)

4.1 PIRIKA on iPad as a Client-server System

Figure 1 shows an overall look-and-feel of PIRIKA on iPad as a client-server system. Figure 2 is a screenshot of the PIRIKA server where the presently connected clients on the leftmost pane, and the communication log among the server and clients on the right pane, are listed, so that users can monitor argument processes generated by the dialectical proof theory (Definition 15-17) in Section 3.

4.2 Invoking PIRIKA on iPad

Figure 3 is an initial screenshot on the standard screen of iPad which includes the PIRIKA icon. By tapping it, we proceed to the user/agent registration page.

Figure 3: Pirika icon on the top page of iPad.

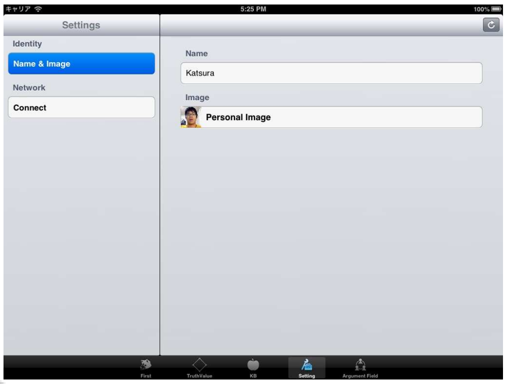

Figure 5: Setting up agent information

Figure 4: Connecting to PIRIKA server.

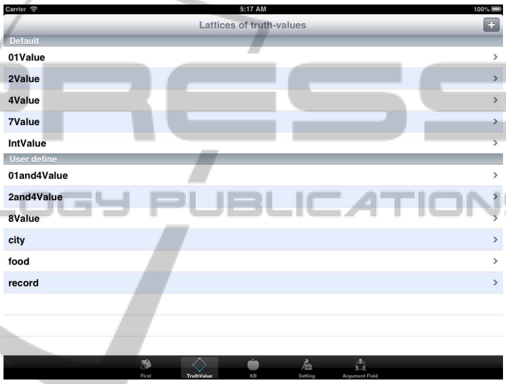

Figure 6: Truth values editor.

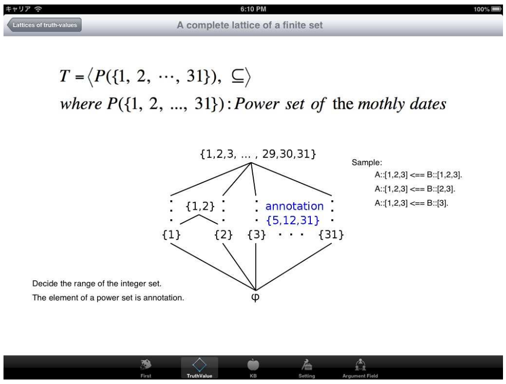

Figure 7: Lattice of the power set of the monthly dates.

4.3 Registering Agents with the Argument Server

Agent (as avatar of human) who wants to commit to argumentation has to register its name and image with the argument server which predesignates its IP and PORT numbers. Figure 4 shows a successful connection to the server, and Figure 5 shows a screenshot for registering agent's name and image with the argument server.

4.4 Preparing a Lattice of Truth Values for Dealing with Uncertainty

In this stage, agents prepare a lattice of truth values for dealing with uncertainty, depending on application domains, following Definition 1. There is prepared an editor for specifying truth values as a complete lattice. Actually this is a standard text editor with which a complete lattice of truth values are stipulated in terms of Prolog.

Users can either use the built-in truth values or specify a user-defined truth values by using the truth values editor as shown in Figure 6, in which the upper pane includes the built-in truth values such as two values $\mathcal{TWO} = \{\mathbf{t}, \mathbf{f}\}$, four values $\mathcal{FOUR} = \{\perp, \mathbf{t}, \mathbf{f}, \top\}$, the power set of dates $\mathcal{P}(\{1, \ldots, 31\})$, the unit interval of reals $\Re[0,1]$, its product $\Re[0,1]^2$, and Jaina's seven values $\mathcal{JAINA} = \{\mathfrak{t}, \mathfrak{f}, \mathfrak{i}, \mathfrak{ft}, \mathfrak{fi}, \mathfrak{it}, \mathfrak{fit}\}$, and in the lower pane user-defined ones. Figure 7 depicts a built-in lattice of the power set of the monthly dates.

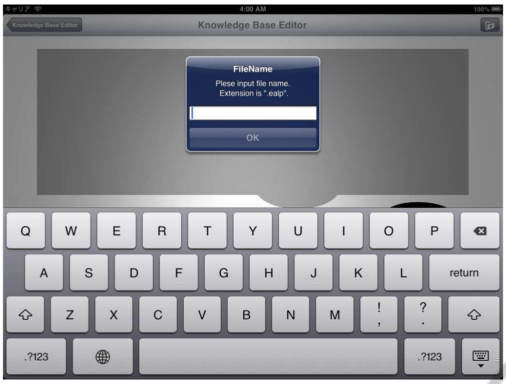

Figure 8: Knowledge base editor of PIRIKA on top of iPad.

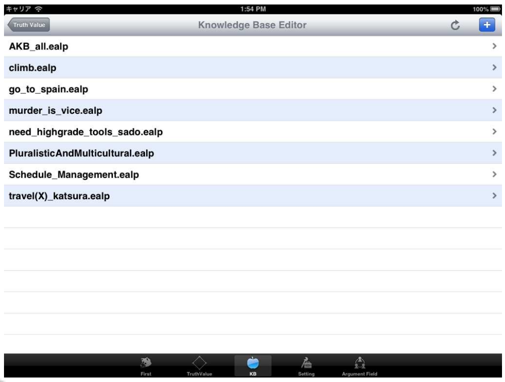

Figure 10: List of various knowledge bases.

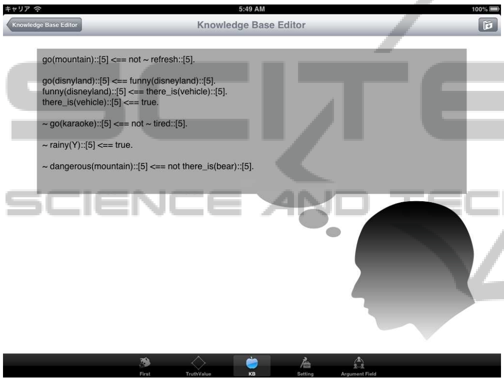

Figure 9: Knowledge base for argumentation.

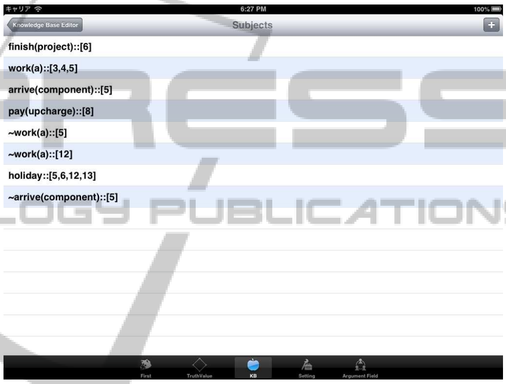

Figure 11: Suggested agendas by PIRIKA.

4.5 Designing Knowledge Bases under the Specified Truth Values in Terms of EALP

The annotation in EALP plays an essential role in specifying argument knowledge since it allows agents to represent their epistemic or cognitive states for propositions that describe the argument world. Once the annotation has been specified, the next step is to provide the argument knowledge that agent conceives. Agent specifies its own argument knowledge in terms of EALP (Definition 3) by using the knowledge base editor as in Figure 8 with the keyboard, resulting in a bunch of knowledge listed in Figure 9. Figure 10 shows a list of knowledge bases that agent has accumulated with respect to every argument topic it has committed so far.

4.6 Starting Argumentation on Submitted Issues/Claims in LMA

The argumentation starts by submitting agendas or selecting possible agendas which PIRIKA helpfully suggests to agents envisaged from the knowledge base. Figure 11 shows a list of suggested agendas by PIRIKA.

4.7 Visualizing the Live Argumentation Process and Diagramming Arguments

At this stage, PIRIKA launches an argument on an issue or claim which has been submitted by the agent, and generates all the possible dialogue trees according to the dialectical proof theory of LMA (Definition 15-16). Figure 12 shows a dialogue tree which contains only one winning dialogue tree (Definition 17). Such a visualization or diagramming is the most important part of PIRIKA since we are not only concerned with argument results but also the overall structure and flow of an argument now developing. We further can see the structure of an argument itself in an argument tree form by long pressing the node on the dialogue tree (Figure 13). Then, we can of course use such physical features as pinch, swipe, tap, etc. actions of iPad that would be helpful to further support visualization and diagramation (Figure 14).

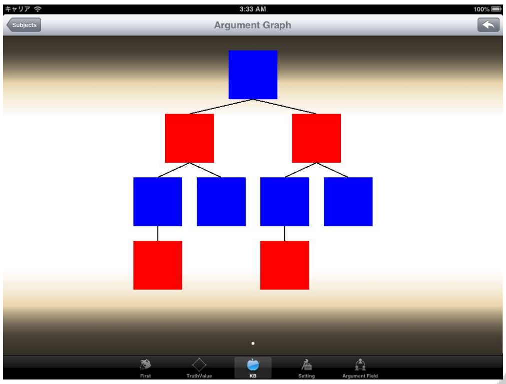

Figure 12: Dialogue tree.

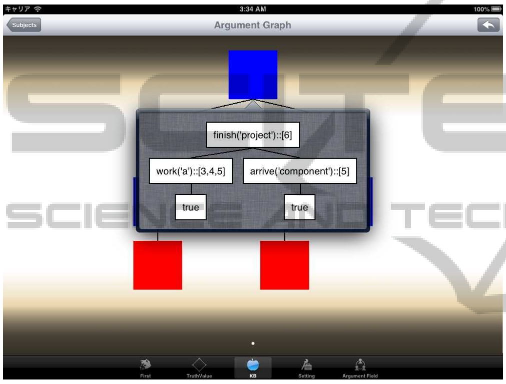

Figure 13: Argument tree at a node.

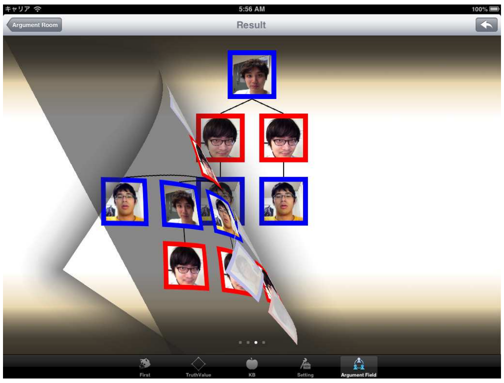

Figure 14: Swiping for looking the next dialogue tree.

So far, one agent attempted to argue about his own issue in a monological way. From here, we will illustrate how agents can enter into the argumentation asynchronously. Any agent can see what issues are now being argued among agents by pressing the tab ‘argument field’ on the lower rightmost corner of the screen, and commit to it if it wishes to do so anytime and anywhere. In Figure 15, there can be seen many issues, now developed on the Internert.

Figure 15: Entering into the asynchronous argumentation.

Figure 16: Asynchronous argument process (1).

Figure 17: Asynchronous argument process (2).

4.8 Asynchronous Argumentation Process

In the argumentation field (agora), there is one agent sitting on the chair (Figure 16). His argument is justified since there is no attacking argument from other parties for now (Figure 17), subject to Definition 17.

Then, another agent comes to the agora on the Internet, wanting to commit the on-going argument

Figure 18: Asynchronous argument process (3).

Figure 19: Asynchronous argument process (4).

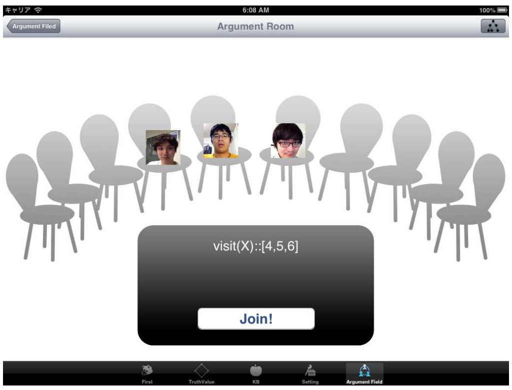

Figure 20: Asynchronous argument process (5).

with his knowledge through his own iPad. Actually it can join by clicking the 'Join' button on the agora. Now the augment agora consists of two agents (Figure 18). Thus, the argument on the common issue begins among two agents, and results in some justified arguments. Figure 19 shows one of them. Furthermore, the third agent appears in the agora, again wanting to participate in the on-going argumentation with his knowledge through his own iPad (Figure 20).

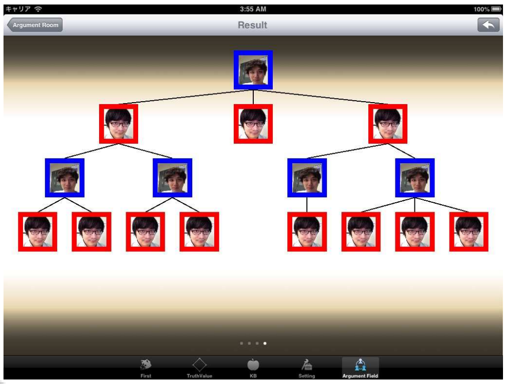

Figure 21: Argument graph (6).

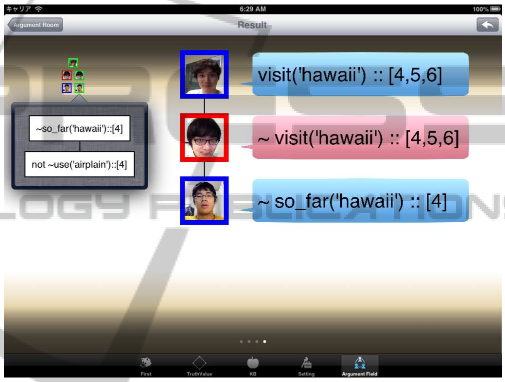

Figure 22: Winning argument tree (7).

Likewise, the argumentation on the common issue begins among three agents, and results in some dialogue trees. Figure 21 shows one of them.

4.9 Determining the Status of an Argument

Figure 22 is a winning dialogue tree (Definition 17), showing that the issue 'visit(hawaii)::[4,5,6]' is justified for now. It says that the travel destination Hawaii was definitely sought as a result of the schedule coordination through the LMA-based argumentation. This, however, is not a final justification status of the given issue. The advent of further new agents who may emerge from outside asynchronously may change the result. This phenomena evidences no less non-monotonicity in argumentation.

5 LESSONS LEARNED FROM THE USES OF PIRIKA

We have attempted various experimental uses of

PIRIKA on iPad over the Internet. The assessment of its effectiveness should be done only through various empirical uses since we have no theoretical communication model for PIRIKA for now. This also can apply for other communication tools such as Twitter, Line, email etc. that have begun as brainchild of engineers. Someday in the future, however, psychologists or sociologists may be to analyze and explain how and what those communication modes and styles imply for people and society.

Here we summarize advantageous points at this stage as follows, which we have confirmed from participants in experimental and daily uses:

- Asynchrony in argumentation: This is our first intended goal of asynchrony like in communication with email, which was found potentially useful since it allows agents and people to make arguments at whim at any time but still in a logically disciplined manner. Actual usefulness for users will be verified through real or simulated ongoing arguments.
- Logical and critical thinking: Argumentation starting from grounds raises persuasive and critical power toward other parties. Put it differently, the asynchronous argumentation tends to bring us an attitude to take deeper thought all the time even for the little things.
- Visualization of argument structure : We can notice which part of the argumentative dialogue is weak or strong since PIRIKA visualizes resulting in dialogue structures. So we could put forward alternative arguments in no time, taking advantage of the feature of the asynchronous argumentation.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed the asynchronous argumentation as a new way to communication with a pervasive personal tool, iPad. We also have tried it out, resulting in a good reputation from users who participated in experimental uses in the daily life.

Somewhat philosophically speaking, the asynchronous argumentation may be as well reworded as asymptotic or incremental argumentation since agents could approach towards truth or justification every time argument is put forward by an agent. So the philosophy of the asynchronous argumentation might be said to be sort of gradualism or incrementalism. On the other hand, the asynchronous argumentation obviously practices so-called non-monotonicity in its pragmatic expansion of argumentation. We would also say that it implicitly includes the laws of the negation of the negation in the Engels' dialectics (Sawamura et al., 2000).

The next step is to port PIRIKA on iPad to mobile phones like iPhone, a more pervasive personal tool. It is expected that such an attempt will open up a new horizon for a more deliberate and logical human communication through computational argumentation research as well as for the social network service in the future.

The open source software and the video clip of PIRIKA on iPad are available at URL http://www.cs.ie.niigata-u.ac.jp/Paper/Research/aappct/visit(X).m4v, PIRIKA-ios.zip. And also PIRIKA will become available from the Apple store for free.

REFERENCES

Dung, P. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logics programming and n-person games. *Artificial Intelligence*, 77:321–357.

Kifer, M. and Subrahmanian, V. S. (1992). Theory of generalized annotated logic programming and its applications. *J. of Logic Programming*, 12:335–397.

Oomidou, Y., Sawamura, H., Hagiwara, T., and Riche, J. (2013). Non-standard uses of pirika: Pilot of the right knowledge and argument. volume 8291 of *Lecture Notes in Computer Science*, pages 229–244. Springer.

Prakken, H. and Sartor, G. (1997). Argument-based extended logic programming with defeasible priorities. *J. of Applied Non-Classical Logics*, 7(1):25–75.

Rahwan, I. and Simari, G. R. E. (2009). *Argumentation in Artificial Intelligence*. Springer.

Sawamura, H., Umeda, Y., and Meyer, R. K. (2000). Computational dialectics for argument-based agent systems. In *4th International Conference on Multi-Agent Systems (ICMAS 2000)*, pages 271–278. IEEE Computer Society.

Takahashi, T. and Sawamura, H. (2004). A logic of multiple-valued argumentation. In *Proceedings of the third international joint conference on Autonomous Agents and Multi Agent Systems (AAMAS'2004)*, pages 800–807. ACM.

Tannai, S., Ohta, S., Hagiwara, T., Sawamura, H., and Riche, J. (2013). The state of the art in the development of a versatile argumentation system based on the logic of multiple-valued argumentation. In *Proc. of International Conference on Agents and Artificial Intelligence (ICAART2013)*, pages 217–224. INSTICC.