# Developing Software for Training Argumentation Skills

**Mare Koit**[1]

**Abstract.** The paper introduces a dialogue model and software we have been developing for training the user´s argumentation skills. We consider dialogues in natural language where one participant *A* is influencing his partner *B* to make a decision about performing an action *D*. In communication, *A* presents various arguments for *D* in order to direct *B*´s reasoning process; he stresses the positive and down-grades the negative aspects of *D*. When playing *B*´s role, the user can develop her skills - how to oppose, how to avert the partner´s arguments, and how to find suitable counter-arguments.

## 1 INTRODUCTION

When one person initiates communication with another (s)he proceeds from the fact that the partner is a human being who feels, reasons and has wishes and plans like every human being. In order to be able to foresee what processes will be triggered in the partner after a move and what will the outcome of these be, the agent must know the inner workings of the partner´s psychological mechanisms [10]. When aiming at a certain goal in communication, the agent must know how to direct the functioning of these mechanisms in order to bring about the intended result in the partner. This knowledge forms a necessary part of human communicative competence. Without it the intentional, goal-directed communication is impossible.

We are considering dialogues where the goal of one of the partners, *A*, is to get another partner, *B*, to carry out a certain action *D*. This type of dialogue constitutes one kind of so-called agreement negotiation dialogues [12]. Such a dialogue can be considered, on a more general level, as rational behaviour of conversational agents which is based on beliefs, desires and intentions of agents, at the same time being restricted by their resources [7].

Because of this, we have modeled the reasoning processes that people supposedly go through when working out a decision whether to perform an action or not. In a model of conversational agent its cognitive states as well as cognitive processes will be represented. One of the most well-known models of this type is the BDI (belief-desire-intention) model [1, 2].

The paper has the following structure. Section 2 introduces our dialogue model that includes a reasoning model. The model is implemented as a conversational agent, which interacts with a user in Estonian and can be used for training argumentation skills of the user. Section 3 represents interaction as update of information states of the conversational agent. Section 4 discusses some implementation details and section 5 draws conclusions.

[1] Institute of Computer Science, University of Tartu, Estonia, email: mare.koit@ut.ee

## 2 MODELLING ARGUMENTATION DIALOGUE

Let us consider dialogue between two participants (*A* and *B*) in a natural language. In the goal base of the initiator (let it be *A*) a certain goal $G^A$ related to *B*´s activities gets activated. In constructing his first turn (request the partner to perform some action *D*), *A* must plan the dialogue acts and determine their verbal form as a turn $tr_1^A$. The partner *B* interprets *A*´s turn and generates her own response $tr_1^B$. *B*´s response triggers in *A* the same kind of process in the course of which he has to evaluate how the realization of his goal $G^A$ has proceeded, and depending on this he may activate a new sub-goal of $G^A$, and the cycle is repeated: *A* builds a new turn $tr_2^A$. Dialogue comes to an end, if *A* has reached his goal or abandoned it.

After *A* has requested *B* to perform *D*, *B* can respond with agreement or rejection, depending on the result of her reasoning about the action. *B*´s rejection can be supported with an argument. These arguments can be used by *A* as giving information about the reasoning process that brought *B* to the (negative) decision.

### 2.1 Reasoning Model

Our reasoning model is based on the studies in the common-sense conception of how the human mind works in such situations, cf. [4] since in natural communication people depart from this conception, not from any scientific one.

In our model we try to reflect the main types of determinants that motivate humans to act. Thus the strategy used depends on which determinant is chosen as the target of influence.

In general lines our reasoning model follows the ideas realized in the BDI model. But it has a certain particular feature - we want to model a ´´naïve´´ theory of reasoning, a ´´theory´´ that people use when interacting with other people and trying to predict and influence their decisions [5].

The reasoning model consists of two parts [9]: (1) a model of human motivational sphere; (2) reasoning procedures. In the motivational sphere three basic factors that regulate reasoning of a subject concerning *D* are differentiated. First, subject may *wish* to do *D*, if pleasant aspects of *D* for him/her overweight unpleasant ones; second, subject may find reasonable to do *D*, if *D* is *needed* to reach some higher goal, and useful aspects of *D* overweight harmful ones; and third, subject can be in a situation where (s)he *must* (is obliged) to do *D* - if not doing *D* will lead to some kind of punishment. We call these factors WISH-, NEEDED- and MUST-determinants, respectively.

We represent the model of motivational sphere of a subject by the following vector of weights:

*w = (w(are-resources), w(pleasant), w(unpleasant), w(useful), w(harmful), w(obligatory), w(prohibited), w(punishment-do), w(punishment-not))*.

In the description, *w(pleasant)*, etc. mean weight of pleasant, etc. aspects of *D*; *w(punishment-do)* - weight of punishment for doing *D* if it is prohibited and *w(punishment-not)* - weight of punishment for not doing *D* if it is obligatory. Here *w(are-resources)* = 1, if subject has resources necessary to do *D* (otherwise 0); *w(obligatory)* = 1, if *D* is obligatory for the reasoning subject (otherwise 0); *w(prohibited)* = 1, if *D* is prohibited (otherwise 0). The values of other weights are non-negative natural numbers.

Resources of the subject concerning *D* constitute any kinds of internal and external circumstances which create the possibility to perform *D* and which are under the control of the reasoning subject.

The values of the dimension obligatory/prohibited are in a sense absolute: something is obligatory or not, prohibited or not. On the other hand, the dimensions pleasant/unpleasant, useful/harmful have a scalar character: something is pleasant or useful, unpleasant or harmful to a certain degree. For simplicity, it is supposed here that these aspects have numerical values and that in the process of reasoning (weighing the pro- and counter-factors) these values can be summed up.

Of course, in reality people do not operate with numbers. Anyway, existence of certain scales also in human everyday reasoning is apparent. For instance, for the characterization of pleasant and unpleasant aspects of some action there are specific words: *enticing, delightful, acceptable, unattractive, displeasing, repulsive* etc. We may suppose that each of these adjectives can be expressed quantitatively.

The second part of the reasoning model consists of reasoning procedures that supposedly regulate human action-oriented reasoning. A reasoning procedure represents steps that the subject goes through in his/her reasoning process; these consist in computing and comparing the weights of different aspects of *D*; and the result is the decision to do *D* or not.

The reasoning depends on the determinant which triggers it (WISH, NEEDED or MUST). In addition, a reasoning model, as a naïve theory of mind, includes some principles which represent the interactions between determinants and the causal connection between determinants and the decision taken. For instance, the principles fix such concrete preferences as:

- people want pleasant states and do not want the unpleasant ones
- people prefer more pleasant states to less pleasant ones.

We do not go into details concerning these principles here. Instead, we refer to [9].

As an example, let us present a reasoning procedure which is triggered by WISH-determinant, that is, if the subject believes that it would be pleasant to do *D* (JSP diagram in Fig. 1). WISH-determinant gets activated when a reasoning subject finds that the action *D* itself or some of its consequences would be pleasurable to him/her, i.e. *w(pleasant) > w(unpleasant)*.

In the case of other input determinants (NEEDED, MUST) the general structure of the algorithm is analogous, but there are differences in certain steps.

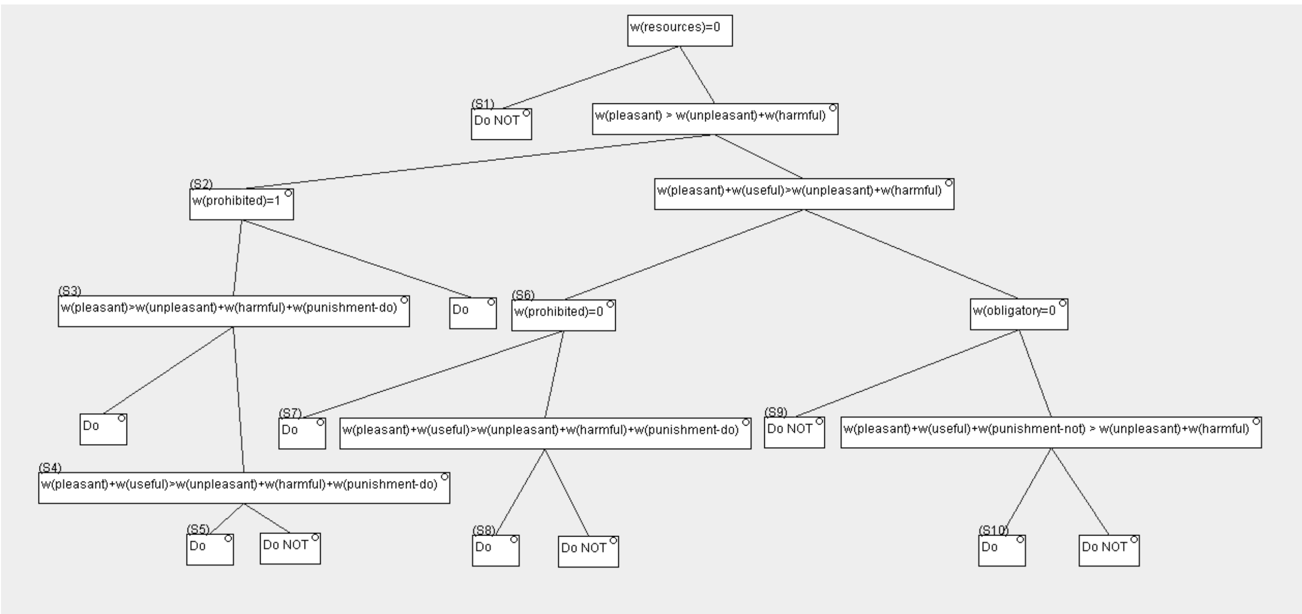


Figure 1. Reasoning procedure WISH.

When comparing our model with BDI model, then beliefs are represented by knowledge of the reasoning subject with reliability less than 1; desires are generated by the vector of weights *w*; and intentions correspond to the goals in the goal base. In addition to desires, from the vector of weights we also can derive some parameters of the motivational sphere that are not explicitly conveyed by the basic BDI model: needs, obligations and prohibitions.

The vector(s) $w^{AB}$ (*A*´s beliefs concerning *B*´s evaluations, where *B* denotes agent(s) *A* may communicate with) are used as partner model(s).

## 2.2 Communicative Strategies and Tactics

A communicative strategy is an algorithm used by a participant for achieving his/her goal in interaction. The initiator of communication (the participant *A*) can realize his communicative strategy in different ways: stress pleasant aspects of *D* (i.e. *entice* the partner *B*), stress usefulness of *D* for *B* (i.e. *persuade B*), stress punishment for not doing *D* if it is obligatory (*threaten B*), etc. We call communicative tactics these concrete ways of realization of a communicative strategy. *A*, trying to direct *B*´s reasoning to the positive decision (to perform *D*), proposes arguments for doing *D* while *B*, when opposing, proposes counter-arguments.

The simplest tactic, which *A* can use in order to achieve his communicative goal is so-called *defense* - here he does not stress any positive aspects of performing *D* for *B* but only averts (downgrades) counter-arguments presented by *B*. For example, in the following dialogue excerpt (1), *B* repeatedly points to missing resources while *A* tries to indicate how the resources can be obtained [8].

(1)
*A: Please prepare a potato salad.*
*B: I do not have enough time.*
*A: I will help you.*
*B: My mother is waiting for me.*
*A: Call home.*

# 3 CONVERSATION AS UPDATE OF INFORMATION STATES

Several dialogue management architectures have been implemented in dialogue systems [6]. The most powerful are information-state dialogue managers [11]. Information state represents cumulative additions from previous actions in the dialogue, motivating future actions of the conversational agent. The functions of the dialogue manager can be formalized in terms of information state update. In our software, we use information state architecture.

## 3.1 Representation of Information States

The key of an information state is the partner model, which is changing during the interaction.

There are two parts of an information state of a conversational agent *A* - private (information accessible only for the agent) and shared (accessible for both participants). The private part consists of the following information slots [8]:

- current partner model (vector $w^{AB}$ of weights - *A*´s picture about *B*)
- a tactic $t_i^A$ which *A* has chosen for influencing *B*
- a reasoning procedure $r_j$ which *A* is trying to trigger in *B* and bring it to a positive decision (it is determined by the chosen tactic, e.g. when enticing, *A* triggers the reasoning procedure *wish* in *B*)
- stack of (sub-)goals under consideration. At the beginning, *A* puts its initial goal into the stack (*B* decides to do *D*). In every information state, the stack contains an aspect of *D* under consideration (e.g. while *A* is enticing *B* then pleasantness is on the top)
- set of dialogue acts $DA=\{d_1^A, d_2^A,..., d_n^A\}$. There are the following DA-s for *A*: proposal, assessments for increasing or decreasing weights of different aspects of *D* for *B*, etc.
- (finite) set of utterances as verbal forms of DA-s, incl. utterances for increasing or decreasing the weights (arguments for/against) $U=\{u_{i1}^A, u_{i2}^A,..., u_{iki}^A\}$. Every utterance has its own weight - numerical value: $V=\{v_{i1}^A, v_{i2}^A,..., v_{iki}^A\}$ where $v_{i1}^A$, etc. is the value of $u_{i1}^A$, etc., respectively. Every argument can be chosen by *A* only once.

The shared part of an information state contains

- set of reasoning models $R=\{r_1,..., r_k\}$
- set of tactics $T=\{t_1, t_2,..., t_p\}$
- dialogue history - the utterances together with participants´ signs and dialogue acts $p_1{:}u_1[d_1]$, $p_2{:}u_2[d_2]$, ..., $p_i{:}u_i[d_i]$ where $p_1=A$, $p_2$, etc. is *A* or *B*.

## 3.2 Update Rules

There are different categories of update rules which will be used for moving from the current information state into the next one:

I. rules used by *A* in order to interpret *B*´s turns
II. rules used by *A* in order to generate its own turns:

1) for the case if the ´´title´´ aspect $a^*(t_i)$ of the current tactic $t_i$ is located on top of the goal stack (e.g. if the tactic is enticing then the ´´title´´ aspect is pleasantness)
2) for the case if another aspect $a_j$ is located on the ´´title´´ aspect of the current tactic $t_i$ (e.g. if *A* is trying to increase the pleasantness of *D* for *B* but *B* argues for usefulness, then the usefulness lies over the pleasantness)
3) for the case if there are no more utterances for continuing the current tactic (and a new tactic should be chosen if possible)
4) for the case if *A* has to abandon its goal
5) for the case if *B* has made the positive decision and therefore, *A* has reached the goal.

Specific rules of the category II exist for updating the initial information state.

For example, the rules of category I have the following general form:

IF the current tactic is $t_i$ AND B´s last utterance was about D´s aspect $a_j$ THEN **put**, firstly, the ´´title´´ aspect $a^*(t_i)$ and secondly, $a_j$ into the goal stack.

Generating a response turn, *A* has, firstly, to attack *B*´s argument concerning the aspect $a_j$ and secondly, to stress the ´´title´´ aspect $a^*(t_i)$ of the current tactic $t_i$ taking them off from the stack in the reverse order.

For another example, the general form of the rules of category II-2 is as follows:

IF the current tactic is $t_i$ AND $a_j$ is on the top of the goal stack AND $a^*(t_i)$ lies under the top aspect in the goal stack AND there are utterances for decreasing $w(a_j)$ by $x$ units AND there are utterances for increasing $w(a^*(t_i))$ by $y$ units AND reasoning triggered `by the determinant` $a^*(t_i)$ `on the changed partner model gives a decision ´´do` *D*`´´ THEN` **`choose`** `these utterances (and the corresponding dialogue acts) AND` **`eject`** $a_j$ `and` $a^*(t_i)$ `from the goal stack.`

In such a case, *B* has presented the counter-argument against performing *D,* concerning the aspect $a_j$. *A* has, firstly, to attack this counter-argument and secondly, to stress the ´´title´´ aspect of the current tactic.

# 4 SOFTWARE DEVELOPMENT

Our software is implemented in two versions which differentiate from each other mainly by involvement of linguistic processing. In both variants, the computer plays *A*´s role.

In the first version, there are ready-made expressions for both the computer and the user, each of which represents an Estonian sentence. Consequently, the computer does not make any morphological and syntactic analysis or generation of texts and does not use any linguistic knowledge. Semantic analysis and generation are extremely simplified by classifying all the expressions. For example, sentences informing about the communicative goal (*The firm offers you a trip to Venice.*), sentences stressing/downgrading the pleasant/unpleasant/useful etc. aspects of an action (*The nature is very beatiful there*., *You must pay the travel costs yourself.*, etc.), affirming sentences (*OK, I shall go*.), etc. Semantic analysis/generation of an expression means only determining its semantic class. The files of Estonian sentences can easily be substituted with their translations and thus interaction can take place in another language.

In the second version of the software, there are ready-made expressions only for the computer. The user can put in free text which will be analysed by the computer. Speech recognition and speech synthesis of Estonian are not included.

## 4.1 Interaction with the User

At the beginning of a communication process the computer (*A*) chooses tactics (of enticing, convincing or threatening) and generates (randomly) a model of the partner, according to which the corresponding reasoning procedure (*wish, needed* or *must*) yields a positive decision, i.e. the computer presupposes that its partner (user *B*) can be influenced this way. A dialogue begins by an expresssion of the communicative goal (this is the first turn $tr_1^A$ of the computer). If the user refuses (after his reasoning: to perform *D* or not by implementing a normal human reasoning which we are trying to model here), the computer recognizes on the basis of the response ($tr_1^B$) the step in the reasoning procedure where the reasoning forked into the ´´negative branch´´, determines the aspect of *D* the weight of which does not match the reality and changes this weight so that a new model will give a negative result as before but it is an extreme case: if we increased this weight (in case of positive aspects of D) or decreased it (in case of negative ones) by one unit we should get a positive decision. We suppose that *A*´s each expression has a value (in the first version of software all the values are equal to 1) and will change a weight of the corresponding aspect of the action *D* by this value. The computer composes its turn $tr_2^A$ choosing a sentence from the set of sentences for increasing/decreasing this weight and at the same time it increases/decreases this weight in the partner model by the value of the chosen sentence. A reasoning procedure based on the new model will yield a positive decision (i.e. the computer supposes that the user´s decision will be positive). Now the user must choose (or put in as free text) his response and the process can continue in a similar way.

A dialogue will be generated jointly by a user and the computer. The computer uses its communicative tactics. Let us suppose that the user - after the computer´s proposal to perform an action *D* - will create a model of herself, i.e. she will attribute values to all aspects of *D* and will do reasoning on the basis of this model. Still, creating this model is certainly inexplicit. In reality the user does not think that the usefulness of *D* is 5 and its harmfulness is 7 but she figures out that doing *D* would be more harmful than useful. In principle, this reasoning procedure may as well be considered creating a model of oneself. The task of the computer is, by implementing its communicative tactics, to try to influence the partner model this way that on the basis of the changed model the partner would make a positive decision. The only problem is that the computer does not ´´know´´ the real weights attributed to different aspects of *D* by the user. It can only make guesses from the user´s negative responses.

As said before, when starting a dialogue the computer randomly generates a user model. At the beginning, we have set only one restriction: we required that the initial model should satisfy the presumption(s) underlying the corresponding reasoning procedure. Thus, for enticing *w(pleasant) > w(unpleasant)*, for convincing *w(useful) > w(harmful)* and for threatening *w(obligatory) = 1*. But the experiments have shown that such an initial model has given relatively bad results. 65% of the dialogues were hopeless because after three pairs of turns the computer had reached such values in the user model that the continuation of the dialogue became meaningless: the weights of negative aspects had reached such a level compared with the positive aspects that it was hopeless to try to reach a model where the reasoning would yield a positive decision by the partner.

The situation improved considerably when we added another restriction to the initial model: we required that the chosen reasoning procedure should aim at getting a positive decision in this model. In real life this restriction is also meaningful: while making a proposal or request we suppose that our partner will agree and only when counter-arguments are put forward shall we try to refute them.

## 4.2 Updating the User Model

The following example demonstrates in more details how the partner model is used in interaction.

Let us suppose that the computer has chosen the tactics of enticing and has generated the following user model:

$w^{AB}$ = *{w(are-resources)=1, w(pleasant)=9, w(unpleasant)=7, w(useful)=5, w(harmful)=0, w(obligatory)=1, w(prohibited)=0, w(punishment-do)=0, w(punishment-not)=1}*.

The reasoning procedure WISH in this model yields a positive decision since *w(are-resources)=1, w(pleasant) > w(unpleasant) + w(harmful), w(pleasant)+w(useful) > w(unpleasant) + w(harmful), w(prohibited)=0* (cf. Fig. 1). Let us suppose that the user chose a refusing sentence and indicated that *w(harm)* must be corrected. There are three possible negative outcomes when applying the procedure *wish* (Fig.1).

Let us suppose here that every sentence has the value 1. In this case: if *w(obligatory) = 1* we have *w(harmful) >= w(pleasant) - w(unpleasant) + w(useful) + w(punishment-not) = 8*. Thus, in the corrected model *w(harmful) = 8*. In this case the procedure *wish* will yield a negative decision as before but if we decreased the value of *w(harmful)* by 1 we should reach a positive decision soon.

If there is more than one possible non-empty domain of allowed values for correcting a weight we shall choose the domain with the greatest lower barrier (for negative aspects of *D*) or with the least upper barrier (for positive aspects), i.e. the worst case.

The following example (2) is an excerpt of an enticing dialogue where the goal of the agent is to reach the partner´s decision to travel to Venice (*A* - computer, *B* - user, ready-made sentences were used by both the computer and the user).

(2)

*A: Would you like to travel to Venice? Our firm needs to conclude a contract there.*

*B: Why me?*

*A: You look very smart, this is important for making contracts.*

*B: Why do I suit better than Mary?*

*A: You have a talent for making such contracts.*

*/---/*

*B: When I am abroad my husband will be unfaithful.*

*A: Sorry, I could not convince you.*

In the second version of the software, a database is used for identifying different key words and phrases in the user input (the input is checked against regular expressions). The database also includes an index of answer files and links to suitable answers, as well as files corresponding to different communicative tactics containing various arguments to present to the user.

The use of unrestricted natural language text as input is both an advantage and a disadvantage for the application as it helps in creating a more natural dialogue but at the same time, if the database is compiled poorly, it can turn the conversation unnatural in a few turns.

## 5 CONCLUSION

We are considering dialogues in natural language where one participant (initiator of interaction, *A*) has a communicative goal that the partner (*B*) will perform an action *D*. If *B* does not agree then in the following interaction, *A* tries to stress positive and down-grade negative aspects of *D* in order to direct *B*´s reasoning about performing *D* toward the positive decision. In the reasoning process, *B* is weighing different aspects of *D*. If the positive aspects weigh more than negative then the decision will be to do *D*. *A* can present different arguments for *D* in a systematic way, e.g. to stress time and again pleasantness of performing *D* (i.e. to entice *B*), to stress usefulness of *D* (i.e. to convince *B*), etc. *A* can also act passively, only averting the arguments presented by *B* and not stressing any positive aspects of performing *D*.

We have worked out a model of conversational agent which includes a reasoning model and implemented it as a computer program, which can be used for training argumentation skills. The user can interact with the computer in Estonian, playing the role of the participant *B*, either choosing ready-made sentences as counter-arguments against performing the action or putting in free texts. In the last case, cue words are used by the computer in order to analyse user sentences. So far, a limited number of voluntary testers have worked with the software. However, we believe that such software is useful when training the skills of finding arguments and counter-arguments for or against performing an action. The program can establish certain restrictions on argument types, on the order in the use of arguments and counter-arguments, etc (cf. [3]).

Our future work includes implementing a conversational agent that can also play *B*´s role and software evaluation in user studies.


## ACKNOWLEDGEMENTS

This work is supported by the Estonian Science Foundation (grant 9124).


## REFERENCES


[1] J. Allen, *Natural Language Understanding*. 2nd ed. The Benjamin/Cummings Publ. Comp., Inc. 1995.

[2] G. Boella. and L. van der Torre. ´BDI and BOID Argumentation´, *Proc. of CMNA-03* http://www.computing.dundee.ac.uk/staff/creed/-research/previous/cmna/finals/boella-final.pdf (2003).

[3] S. Bringsjord, J. Taylor, A. Shilliday, M. Clark, K. Arkoudas, and S. Khemlani.´Slate: ´An Argument-Centered Intelligent Assistant to Human Reasoners´, *Proc. of CMNA-08* http://www.cmna.info/-CMNA8/ (2008).

[4] D´Andrade, R. ´A Folk Model of the Mind´, *Cultural models of Language and though,* Eds. D. Holland and A. Quinn, London: Cambridge University Press, 112-148, (1987).

[5] M. Davies and T. Stone, *Folk psychology: the theory of mind debate.* Oxford, Cambridge, Massachusetts: Blackwell. 1995.

[6] J. Ginzburg and R. Fernández, ´Computational Models of Dialogue´, Clark, Fox, and Lappin (eds.) *The Handbook of Computational Linguistics and Natural Language Processing*, Wiley Blackwell Publishers, 429-481, (2010).

[7] K. Jokinen, *Constructive Dialogue Modelling: Speech Interaction and Rational Agents*. John Wiley & Sons Ltd, 2009.

[8] M. .Koit, ´Conversational Agent in Argumentation: Updating of Information States´, *Proc. of the International Conference on Knowledge Engineering and Ontology Development: KEOD 2011, Paris, France, 26-29 October, 2011. (Eds.) Joaquim Filipe and Jan L. G. Dietz.* Paris: SciTEC Publications Ltd, 2011, 375-378, (2011).

[9] M. Koit and H. Õim, ´Dialogue management in the agreement negotiation process: a model that involves natural reasoning´, *The 1st SIGdial Workshop on Discourse and Dialogue.* Ed. L. Dybkjaer, K. Hasida, D. Traum. HongKong, 102-111, (2000).

[10] H. Õim, ´Naïve Theories and Communictive Competence: Reasoning in Communication´, *Estonian in the Changing World,* University of Tartu, 211-231 (1996).

[11] D.Traum and S. Larsson. ´The Information State Approach to Dialogue Management´, *Current and New Directions in Discourse and Dialogue*, J. van Kuppevelt and R. Smith (Eds.), Kluwer, 325-353, (2003).

[12] T. Yuan, D. Moore, and Alec Grierson, ´Human-Computer Debate: a Computational Dialectics Approach´, *Proc. of CMNA-02.* http://www.csc.liv.ac.uk/~floriana/CMNA/YuanMooreGrierson.pdf (2002).