Acta Scientiarum. Technology
ISSN: 1806-2563
ISSN: 1807-8664
actatech@uem.br
Universidade Estadual de Maringá
Brasil

A framework for the consensus decision-making based on arguments and common knowledge formation

Possebom, Ayslan Trevizan; Morveli-Espinoza, Mariela; Tacla, Cesar Augusto

A framework for the consensus decision-making based on arguments and common knowledge formation
Acta Scientiarum. Technology, vol. 41, 2019
Universidade Estadual de Maringá, Brasil
Available in: https://www.redalyc.org/articulo.oa?id=303260200040
DOI: https://doi.org/10.4025/actascitechnol.v41i1.37955

Ciência da Computação

A framework for the consensus decision-making based on arguments and common knowledge formation

Ayslan Trevizan Possebom
Instituto Federal do Paraná / Universidade Tecnológica Federal do Paraná, Brasil
possebom@gmail.com

Mariela Morveli-Espinoza
Universidade Tecnológica Federal do Paraná, Brasil

Cesar Augusto Tacla
Universidade Tecnológica Federal do Paraná, Brasil

DOI: https://doi.org/10.4025/actascitechnol.v41i1.37955
Redalyc: https://www.redalyc.org/articulo.oa?id=303260200040

Received: 03 July 2017
Accepted: 16 February 2018

Abstract:

In an argumentative dialogue, agents exchange arguments to approve or disapprove a decision alternative. The problem arises when a group of agents with incomplete information needs to reach consensus on the decision. This study proposes a framework for decision-making where the argumentative agents can build the necessary common knowledge to make a decision based on every formula of the exchanged arguments. Each formula receives votes indicating either support or rejection. When a formula receives more supporting votes than rejection ones, that is, it is accepted by the majority, it becomes common knowledge. In this case, there is a consensus about that information, which influences the strength of the argument comprising the formula. The framework for decision-making permits to analyze the arguments related to every decision alternative and rank them based on the strength of the arguments. This framework is an alternative for existing ones based on preferences expressed by numerical values, allowing agents to explain the rationale behind the decisions.

Keywords: dialogue, argumentation, formal logic, agents.

Introduction

When a group of agents have to decide collectively about a problem, choosing one alternative among others, they usually use an aggregation procedure to combine the individual preferences into a collective one. The aggregation of opinions occurs especially when there are divergences of opinions. The most common techniques for group decision applied to intelligent agents are majority vote, Borda count, Condorcet criterion, approval voting, judgement aggregation (Gehrlein, 2006; Grossi & Pigozzi, 2014), and a number of variations of these techniques. All of them use voting as the basic principle to find the preferred result. Besides that, all agents must vote, even when they do not have enough information. Such mechanisms do not take the whole process of consensus achievement into consideration, which includes the possibility for the participants to change their minds before voting and the reasons for the pros and cons votes. Therefore, a dialogue protocol allowing all participants to express their opinions and to share knowledge seems an important phase for reaching consensus.

Author notes

possebom@gmail.com

Agreement about a decision implies common knowledge, it is a necessary requirement when a group of agents need to make decisions. In Fagin, Halpern, Moses, and Vardi (2004), common knowledge is defined as 'the state of knowledge where everyone knows, everyone knows that everyone knows, everyone knows that everyone knows that everyone knows, etc.'. The definition of common knowledge is in accordance with what is needed for consensus decision-making since consensus occurs when all opinions are considered and evaluated by the group (Neary & Winn, 2017).

Arguments sent in a dialogue express points of view about the decision alternative in discussion or other counter-arguments. The strength of arguments can be obtained through the degree of consensus in their formulas, indicating to what extent that argument is considered accepted by the group. The goal of this paper is to present a framework to build common knowledge from information presented by agents in a decision-making dialogue. Whenever an argument is sent to the group, it is analyzed to what extent its formulas are known by the group and the agents' belief sets are updated with the consented formula. In addition, it is proposed a method to analyze the arguments for or against a decision alternative, choosing the alternative preferred by the group.

In the next section, it is defined the framework for building common knowledge through the arguments in a dialogue, the argument strength calculation and the decision making. Next, it is presented a discussion about how the framework can be implemented and the results, following the conclusions and future work.

Material and methods

In this section, we introduce the fundamental background on knowledge representation and we focus on the agent architecture, the argumentative dialogue, and the process for (i) computing which piece of information in an argument should be accepted as common knowledge, (ii) computing the argument strength with the intrinsic and the overall strengths, and (iii) choosing the alternative preferred by the group.

Possible-world and knowledge representation

The classical model of reasoning about knowledge of a single agent is known as possible-world model. Possible-worlds represent the possible state of affairs (that is, there may be situations where a fact holds for a topic under discussion, but do not hold for another topic) (Fagin et al., 2004). An agent ag_i knows a fact Φ if Φ is necessarily true, i.e. it is true in all the possible worlds. The modal operators $K_1 ,..., K_n$ for n agents represent the knowledge of agent ag_i , i = 1, ..., n. The formula K_1p is read 'agent ag_1 knows p', K_1K_2p is read 'agent ag_1 knows that ag_2 knows p' and $\neg K_2K_1K_3r$ is read 'agent ag_2 does not know that ag_1 knows that ag_3 knows r'.

When the reasoning involves the knowledge of a set of agents AG, two modal operators can be defined (Fagin & Halpern, 1994): E_{AG} (i.e., $E_{AG}r$ represents that every agent in the group knows r) and C_{AG} (i.e., C_{AGr} represents that every agents in the group knows r and they know that every agent in the group knows r, that is, r is common knowledge in the group).

Agent architecture

Consensus is a dynamic and interactive group discussion process, generally coordinated by a moderator, that tries to help the group to make their opinions closer (Singh & Benyoucef, 2013). There are basically two types of agents involved: the argumentative and the mediator. An argumentative agent dialogues, builds arguments, and shares knowledge in some way so that the consensus can be achieved. The mediator agent is responsible for determining, by means of a consensus measure, which facts and rules should be accepted in the group.

Definition 1: Let *AG* be the set of argumentative agents. An agent $ag_i \in AG$ is a tuple $\langle \Sigma_i, CS_i \rangle$ where: Σ_i is the knowledge base with $\Sigma_i = K_i \cup KO_i$ (K_i is a set of formulas representing beliefs about the environment, and KO_i is a set of formulas gathered through communication of agi with $AG \backslash ag_i$); $CS_i = CS_i.S \cup CS_i.A$ is the commitment store ($CS_i.S$ has arguments to be sent and $CS_i.A$ arguments to be analyzed when looking for counter-arguments).

The proposed framework to build common knowledge on the agents' beliefs[1] is based on argument exchange and voting. When an agent sends an argument, it represents his opinion, point of view or a justification about the alternative under discussion. In this paper, Σ has formulas in propositional language and the arguments are built based on that formulas (Besnard & Hunter, 2014). Besides that, is the classical inference, $\equiv$ represents logical equivalence, and _|_ represents contradiction. An argument is formed by a pair $\langle \Phi, a \rangle$ where Φ represents the support (premises) and a the claim of the argument, such that: (1) a is a formula, (2) $\Phi \subseteq \Sigma$, (3) $\Phi \nvdash \perp$ (4) $\Phi \quad a$, (5) $\acute{o}\Phi' \subseteq \Phi$ such that $\Phi' \quad a$. Arguments are created to justify a position against the decision alternative or other arguments. The most common attack relations between arguments are undercut and rebuttal (Parsons & McBurney, 2003; Besnard & Hunter, 2014). Let $arg_1 = \langle \Phi_1, a_1 \rangle$ and $arg_2 = \langle \Phi_2, a_2 \rangle$ be two distinct arguments: arg_1 undercuts: arg_2 iff $\exists \varphi \in \Phi_2$ such that $a_1 \equiv \neg \varphi$, and arg_1 rebuts arg_2 iff $a_1 \equiv \neg a_2$.

When an argumentative agent is requested to speak, he has to choose one move among the following: propose (send set of arguments), 'vote' for support and 'vote' for rejection (in the formulas of the arguments). Whenever an argument is exposed, it is stored in CSi.A and every agent votes supporting or rejecting each argument's formula and tries to find counter-arguments, storing them in the $CS_i.S$ to present them on his turn. One supporting vote for a formula *r* in an argument means that ag_i knows *r* ($K_i r$), conversely, one rejection vote means that $K_i \neg r$ If the $\neg K_i r$ and $\neg K_i \neg r$, he does not vote.

Let $A(\Sigma_i)$ be the set of all arguments that can be built from Σ_i, $ARG \subseteq A(\Sigma_i)$, be the subset of arguments sent by an agent in a dialogue, and $\arg \in ARG$, an argument. The following functions allow us to decompose an argument into its formulas and atoms: $premise : ARG \rightarrow \Phi$ is a function that returns the set of formulas in the premise of an argument arg, $claim : ARG \rightarrow \Upsilon$ is a function that returns the claim of arg, $split : ARG \rightarrow \mathrm{H}$ is a function that returns the set of formulas in arg with $\mathrm{H} = premise(\arg) \cup claim(\arg)$, and $atoms : \mathrm{H} \rightarrow \Pi$ is a function that returns the set of atoms of a formula η with $\eta \in \mathrm{H}$. Each formula η of an argument can receive votes of support and rejection from the participating agents. We refer to Support[η] (and Reject[η]) as the set of agents that support (and reject) formula η during the voting stage.

Definition 2: A formula η in an argument sent by ag_i is supported by ag_j if: (1) $\exists \arg_2 \in A(\Sigma_j) \mid claim(\arg_2) = \eta$, or (2) $\exists \mu \in \Sigma_j \mid atoms(\mu) = atoms(\eta)$ and $\mu \leftrightarrow \eta$ is a tautology. A formula η is rejected if: $\exists \mu \in \Sigma_j \mid atoms(\mu) = atoms(\eta)$ and $\mu \leftrightarrow \eta$ is not a tautology.

The mediator is a special agent in charge of controlling the sequence of messages sent by the argumentative agents, including arguments and votes in a waiting time *t*, access the software artifacts used to synchronize the dialogues and the sequence of speech moves, and inform the agents about the actual state of the dialogue.

Definition 3: A mediator agent *med* is a tuple $\langle wb, Agenda, DT, \sigma \rangle$ where *wb* is a list of argumentative agents registered to send new arguments, *Agenda* is a list that stores all arguments sent by an agent in a round of dialogue, *DT* is a set of dialogue tables (dt_1, dt_2, ...) with a dialogue table for each decision alternative, and σ is a threshold value used to determine when a formula in an argument should be common knowledge.

The artifact *wb* is a coordination mechanism that emulates what happens in face-to-face meetings, when an agent wants to send an argument, he needs to ask for registration in a queue and wait for its turn. This queue guarantees that only registered agents receive the right to send arguments in a coordinated manner. The *Agenda* is a structure implemented as a queue, responsible for storing all arguments sent by the argumentative agents in each round of speak. Every argument in the *Agenda* needs to be verified (i.e., check if it is a valid counter-argument and check if this argument has not been already sent in the dialogue). Each $dt_1 \in DT$ is a table that is populated with the arguments provided in the dialogue moves. Every tuple of this table contains a sequential number *y*, the sender, the argument, and the list of formulas of the argument with the support/rejection votes.

In the proposed framework, every argumentative agent has a value that represents his degree of expertise in the group. This value represents in what extent each information in an argument should be considered as accepted by the group.

Definition 4: A Framework for Common Knowledge formation is a tuple $FCK = \langle AG, TS, D, med, t \rangle$ where: $AG = \{ag_1, ..., ag_n\}$ with $n > 1$ is the set of argumentative agents; $TS = \{ts_1, ..., ts_n\}$ with $\{ts_i \in □ \mid 0 \leq ts_i \leq 1\}$ and $ts_1 + ... + ts_n = 1$ is the expertise of the agents about the alternative under discussion where ag_i has expertise value ts_i; $D = \{d_1, ..., d_m\}$ with $m \geqslant 1$ is the set of decision alternatives to be discussed; *med* is the mediator agent; and *t* is the waiting time used to coordinate the votes and send arguments in a dialogue.

Example 1: Let $FCK = \langle \{ag_1, ag_2, ag_3, ag_4, ag_5, ag_6\}, \{0.22, 0.39, 0.09, 0.09, 0.04, 0.17\}, \{d_1\}, med, 10 \rangle$, $K_1 = \{c, c \rightarrow a\}$, $K_2 = \{\neg a, \neg b\}$, $K_3 = \{a, \neg(a \rightarrow b), \neg b\}$, $K_4 = \{\neg a \rightarrow b, \neg b\}$, $K_5 = \{a, a \rightarrow b\}$, $K_6 = \{d, d \rightarrow a\}$. At some point in the dialogue, agent ag_5 sends argument $\langle \{a, a \rightarrow b\}, b \rangle$. We have that *Support*[a] = $\{ag_1, ag_3, ag_6\}$, *Reject*[a] = $\{ag_2\}$, $Support[a \rightarrow b] = \{\}$, $Reject[a \rightarrow b] = \{ag_3, ag_4\}$, *Support*[b] = {}, and *Reject*[b] = $\{ag_2, ag_3, ag_4\}$.

Building common knowledge

In a group decision-making, support and rejection votes for a formula are weighted by the agents' expertise, so if it is the case that the supporting score is greater than the rejection score, that formula is taken as true by the group and it is included in each Σ_i by means of a belief update function. If rejection score is greater than supporting score, the negation of the formula is then included in Σ_i. The belief update function *buf* computes the consensus level and detect which formulas should be accepted. Equation 1 shows how the *buf* value is obtained. We refer to agi as the issuing agent.

$$buf(\eta) = ts_i + \sum_{ag_j \in Support[\eta]} ts_j - \sum_{ag_j \in Reject[\eta]} ts_j \quad (1) \qquad (1)$$

The result of the function *buf* indicates when the group should consider η or $\neg\eta$ accepted based on a threshold σ value[2]. This result can be obtained with: $buf(\eta)\begin{cases} > \sigma & \Sigma = \Sigma \cup \eta \\ < -\sigma & \Sigma = \Sigma \cup \neg\eta \end{cases}$.

Having the consensus theory in mind, whenever $buf(\eta) > \sigma$, formula η should be accepted and consented by the group. When $-\sigma \leq buf(\eta) \leq \sigma$, there is no consensus about η and this formula is not considered. For $buf(\eta) < -\sigma$, it stands that the group rejects η, thus accepting its negation. Whenever a formula η is consensually accepted, the group updates the Σ_i with that formula and an annotation containing the decision alternative in discussion with the consensus degree, like η[*di(buf)*]. If there is already a formula η in the K_i base of the argumentative agents, only the annotation is updated for that formula. Otherwise, the formula with its annotation has to be included in the KO_i. When there is no consensus, only the agents that knows η or $\neg\eta$ update the formula with the annotation $[d_i(0)]$.

In the dialogue, for some next alternative, consider $K_i = \{a, b[d_2\,(0.3)]\}$ and $KO_i = \{\neg a[d_1(0.6)], \neg b[d_1(0.5)], c[d_2(0.3)]\}$. Agent ag_i always believes in a, but for decision d_1 he considers $\neg a$. The same occurs for *b*. As ag_i knows nothing about *c*, he considers it for every possible dialogue since it is common knowledge. So, if η is presented in another argument and the knowledge base of an argumentative agent is inconsistent, this agent firstly checks if η has any annotation for the current alternative. Otherwise, the voting for support or rejection on η uses preferentially the *K* base. Again, a new annotation has to be added in the formula for the new current alternative with its consensus level.

Example 1 (cont.): Let $\sigma = 0.1$, we have in Formula 1: *buf*(a) = 0.13 and $buf(a) > \sigma$ denoting that formula η = a should be accepted by the group for issue d_1. Agents with no knowledge or rejecting η (ag_2, ag_4) must now accept $a; buf(a \to b) = -0.14$ denoting that $\neg(a \to b)$ should be accepted for issue d_1; *buf*(b) = -0.53 denoting that $\neg b$ should be accepted for issue d_1. The group of agents should update their knowledge bases *K* or *KO* with the formulas annotated $a[d_1(0.13)]$, $\neg(a \to b)[d_1(0.14)]$ and $\neg b[d_1(0.53)]$.

Proposition 1: For every $\Sigma \in ag_i$ and $ag_i \in AG$, we can represent the knowledge of the agents or ignorance using the modal operators Ki, E_{AG} and C_{AG}.

For two knowledge bases $\Sigma_1 = \{\{a, a \to b, c \to \neg b, a \to c\}, \{a \to \neg b[d(0.6)]\}\}$ and $\Sigma_2 = \{\{a, a \to \neg b[d(0.6)], \neg c, \neg c \to \neg b\}, \{\}\}$, the modal operator $K_1 a \to c$ indicates that agent ag_1 knows formula $a \to c$, that is, $a \to c \in K_1$. The modal operator $E_{AG} a$ indicates that agents ag_1 and ag_2 knows formula a, that is, $a \in \Sigma_1$ and $a \in \Sigma_2$. The modal operator $C_{AG} a \to \neg b$ indicates that formula $a \to \neg b$ has already been presented and discussed, becoming a common knowledge. The modal operators $\neg K_1 \neg c$, $\neg E_{AG} g$ and $\neg C_{AG} a \to b$ represent ignorance or missing information, where ag1 does not know formula $\neg c$, the group is unaware of *g* and formula $a \to b$ is not common knowledge for the group of agents.

Proposition 2: Formulas with no annotation are considered acceptable for an agent in all the possible worlds. The annotation $\eta\ [d_i\ (buf)]$ represents common knowledge in a possible world, that is, the issue under discussion.

Proposition 3: Formulas that have been accepted as common knowledge and have not been rejected by the agent do not cause inconsistency of the Knowledge Base. Those formulas are then considered true for all the possible worlds.

A consensually accepted formula is stored in the knowledge base in every agent, so that formulas with annotations are known by every agent and every agent knows that the other agents know that formula, that is, this formula is common knowledge. Thus, the framework for building common knowledge satisfies the complete axiomatization for language of knowledge and common knowledge.

The dialogue process

The dialogue process is presented in Table 1. Agent *med* conducts and synchronizes the speech order, as well as computes the preferences of the group. The functions defined for the *med* with their relations in the argumentative agents are:

In the beginning of the dialogue, *med* initializes the structures needed to conduct the messages exchange, as the *wb* [createWhiteboard()], the dialogue table for the current decision alternative [createDialogueTable(di)], the agenda [createAgenda()], and the initial argument $\langle\{d_i\}, d_i\rangle$ [createInitialArgument(di)].

The messages are broadcasted to all the argumentative agents, as to inform the current argument to be discussed with its corresponding line in the dialogue table [broadcastArgument(arg,y)], ask the agents to find counter-arguments considering the current decision alternative [broadcastAttack(di)], ask for voting supporting and rejecting a formula [broadcastAskVotingAgreement(f), broadcastAskVotingRejection(f)], inform that a formula must be learned or unconsented [broadcastLearn(f,b,di), broadcastInform(f,0,di)], and ask the agents with arguments to be sent [broadcastAskSpeak()].

TABLE 1.
Dialogue process model for building common knowledge.

Agent: *med*	Agent: ag_i
procedure DIALOGUE(di)	**procedure** Argument(initArg,y)
wb ← createWhiteboard()	currentArgument(CS.A,initArg,y)
dtm ← createDialogueTable(di)	**end procedure**
agenda ← createAgenda()	**procedure** Attack(di)
initArg ← createInitialArgument(di)	form ← splitArgument(getArgument(CS.A))
y ← dtAddArgument(dtm,null,initArg,null)	**for each** f **in** form **do**
broadcastArgument(initArg,y)	**if** hasAttack(f,di) **then**
broadcastAttack(di)	att ← getAttacks(f,di)
wait(t)	addAttack(CS.S,att)
addWhiteboard(wb, broadcastAskSpeak())	**end if**
while(!emptyWhiteboard(wb))	**end for**
currentAgent ← getWhiteboard(wb,0)	CS.A ← ∅
listArgs ← askSendArguments(currentAgent)	**end procedure**
listArgs ← validateArguments(listArgs, dtm)	**function** AskSpeak()
fillAgenda(agenda,listArgs)	**if** CS.S ≠ ∅ **then** return true
for each a **in** agenda **do**	return false
y ←dtAddArgument(dtm,currentAgent,a,getAttack(a))	**end function**
broadcastArgument(a,y)	**function** SendArguments()
formulasArgument ← split(a)	args = CS.S
for each f **in** formulasArgument **do**	CS.S ← ∅
f1 ← broadcastAskVotingAgreement(f)	return args
f2 ← broadcastAskVotingRejection(f)	**end function**
b ← buf(f1,f2)	**function** AskVotingAgreement(f)
dtUpdate(dtm,y_new,f,f1,f2,b)	**if** support(f) **then** return true
if b > σ **then** broadcastLearn(f,b,di)	return false
else if b < -σ **then** broadcastLearn(!f,b+1,di)	**end function**
else broadcastInform(f,0,di)	**function** AskVotingRejection(f)
end for	**if** reject(f) **then** return true
dtUpdate(dtm,y,is(a))	return false
broadcastAttack(di)	**end function**
wait(t)	**procedure** Learn(f,b,di)
end for	**if** Khas(f) **and** label(f,di)=∅ **then**
updateWhiteboard(wb)	updateK(f,b,di)
addWhiteboard(wb, broadcastAskSpeak())	**else if** not(KOhas(f)) **then**
end while	addKO(f,b,di)
end procedure	**else if** KOhas(f) **and** label(f,di)=∅ **then**
	updateKO(f,b,di)
	end procedure
	procedure Inform(f,b,di)
	if Khas(f) **then**
	updateK(f,b,di)
	end procedure

To manage the *wb*, the functions are: add argumentative agents that has arguments to be sent [addWhiteboard(wb,broadcastAskSpeak())], update the wb enabling the next agent in the list to send arguments [updateWhiteboard(wb)], get the first agent in the list [getWhiteboard(wb,0)], and check if the list is empty [emptyWhiteboard(wb)]. Other functions are: ask the enabled agent to send his arguments [askSendArguments(ag)], validate the arguments sent by the enabled agent to guarantee that they are counter-arguments, unique, and with formulas that are not rejected nor unconsented [validateArguments(listArgs, dtm)], and fill the agenda with only the valid arguments [fillAgenda(agenda,listArgs)].

The argumentative agents have some functions in response to the requests of med. Table 2 lists this correspondence.

Consensus decision-making process

During the dialogue, agents can send their arguments or counter-arguments (Besnard & Hunter, 2014). After the dialogue, a set of arguments *ARGS* = {arg_1, ..., arg_z} with $z > 0$ was presented. Hereupon, *ARGS* is then mapped to a graph of arguments. One of the best known abstract argumentation frameworks that uses graph of arguments was proposed by Dung (1995) and it is formed by a pair AF = ⟨ *ARGS, R* ⟩ where *ARGS* is the set of arguments and *R* is a binary relation that represents attacks between arguments with $R \subseteq ARGS \times ARGS$. We use notation $R(\arg_1, \arg_2)$ to represent that an argument $\arg_1$ attacks $\arg_2$.

Having an argumentation graph, the consensus decision-making process has two additional stages to suggest the decision alternative that is preferred by the group: compute the strength of arguments and determine to what extent one alternative is preferred to another.

Argument strength

Arguments sent in a dialogue about some decision alternative receive supporting and rejection votes in every formula. These votes are the base for determining the consensus level, which provides the argument intrinsic strength (Cayrol & Lagasquie-Schiex, 2005).

Definition 4: The intrinsic strength is a value obtained by the agent expertise and the concept of group majority knowledge, which express to what extent an argument is reliable based on its formulas.

Let *split*($\arg_i$) be a function that returns the formulas (premises and claim) of $\arg_i$ and *length*($\arg_i$) be a function that returns the size of argi (number of formulas). The intrinsic strength of an argument is obtained according to Equation 2.

$$is(\arg_i) = \left(\frac{\sum_{\eta \in split(\arg_i)} buf(\eta)}{length(\arg_i)} + 1 \right) * 0.5 \qquad (2)$$

(2)

Proposition 4: The total acceptance of the argument results in *is*($\arg_i$) = 1, while the total rejection[3] of the argument results in *is*($\arg_i$) = 0.

While the intrinsic strength considers the supporting and rejection votes for the formulas of one argument, the overall strength copes with attack relations among arguments. Both the rejections in the formulas and the counter-arguments portray a negative effect, weakening the arguments. The overall strength of an argument considers its attackers (the counter-arguments), attackers of their attackers (the defenders), and so on.

TABLE 2.
Relationship of requests (med) and actions performed (agi).

Request in *med*	Action in ag_i	Description
broadcastArgument(arg,y)	Argument(initArg,y)	*arg* is stored in $CS_i.A$
broadcastAttack(di)	Attack(di)	Attacks are stored in $CS_i.S$, cleaning $CS_i.A$
broadcastAskVotingAgreement(f)	AskVotingAgreement(f)	Agents vote for support in f
broadcastAskVotingRejection(f)	AskVotingRejection(f)	Agents vote for rejection in f
broadcastLearn(f,b,di)	Learn(f,b,di)	Formula f with label [di(b)] is added to Σ
broadcastInform(f,0,di)	Inform(f,b,di)	Agents update label for formula f
broadcastAskSpeak()	AskSpeak()	Agents with $CS_i.S \neq \varnothing$ have arguments to send
askSendArguments(ag)	SendArguments()	Arguments in $CS_i.S$ are sent and $CS_i.S$ is cleaned;

Definition 5: The overall strength of an argument is a score that represents its importance when compared to the other arguments in an argumentation graph.

Let *attack*($\arg_i$) be a function that returns the set of arguments that attack $\arg_i$ in *R*, that is, $attack(\arg_i) = \arg_j \in ARGS \mid R(\arg_j, \arg_i)$. To compute the overall strength, it is considered the intrinsic strength of $\arg_i$ and the overall strength of each $\arg_j \in attack(\arg_i)$, according to Equation 3.

$$os(\arg_i) = \frac{is(\arg_i)}{1+\sum_{\arg_j \in attack(\arg_i)} os(\arg_j)} \quad (3)$$

(3)

To solve the entire system of argumentation with strength in the arguments, we resort to the iterative method[4]. We denote by $time_0$ the initial overall strength calculation for each argument, and $time_s$ the overall strength calculation obtained after the s^{th} iteration. An iteration in times is carried out by computing a new overall strength $time_{s+1}$ for all arguments in *ARGS*. We refer to $os(arg_i)^s$ the process of computing the overall strength of arg_i in iteration *s*. The solution to the iterative method is obtained when $os(arg_i)^s = os(arg_i)^{s+1}$ for all arguments and the result is independent of argument processing order.

Example 2: Let an argumentation framework AF = ⟨ {arg_1,arg_2,arg_3,arg_4,arg_5,arg_6,arg_7}, {(arg_1, arg_3), (arg_2, arg_3), (arg_5, arg_3), (arg_3, arg_4), (arg_4, arg_5), (arg_5, arg_6), (arg_6, arg_7), (arg_7, arg_6)} ⟩ with arguments having intrinsic strengths as in Figure 1. The iterations to obtain the overall strength for every argument is presented in Table 3. The ordering used to calculate follows from arg1 to arg7 and the initial overall strength is 0 for all arguments in *time* $_0$.

Making decisions

An argumentation graph represents the dialogue about every alternative in *FCK*. To establish the preference order on the set of alternatives, we count the arguments that defend the alternative (PRO arguments) and those that reject it (CON arguments).

Let $PRO : D \rightarrow 2^{Arg+}$ be a function that returns all arguments with an even distance in a simple path to the node that represents the decision alternative, and $CON : D > 2^{Arg-}$ a function that returns all arguments with an odd distance in a simple path to the node that represents the decision alternative. To compute the level of preference for a decision alternative, we use the Equation 4.

$$pref(d_i) = \sum_{\arg_j \in PRO(d_i)} os(\arg_j) - \sum_{\arg_j \in CON(d_i)} os(\arg_j) \quad (4)$$

(4)

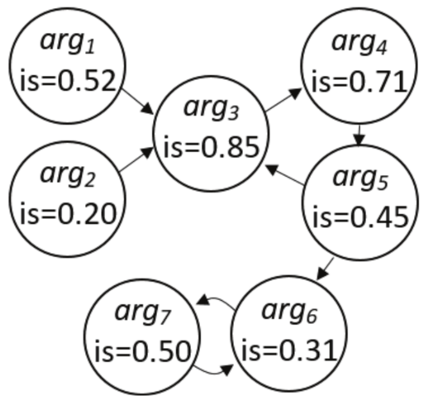

FIGURE 1.
Argumentation framework with strength in arguments.

TABLE 3.
Overall strength calculation.

$time_s$	arg_1	arg_2	arg_3	arg_4	arg_5	arg_6	arg_7
0	0	0	0	0	0	0	0
1	0.52	0.20	0.49	0.48	0.31	0.24	0.40
2	0.52	0.20	0.42	0.50	0.30	0.18	0.42
3	0.52	0.20	0.42	0.50	0.30	0.18	0.42

Having two decision alternatives d_1 and d_2 , and their respective level of preference *pref*(d_1) and *pref*(d_2): (i) if *pref*(d_1) = *pref*(d_2) then d_1 is as preferable as d_2; (ii) if *pref*(d_1) > *pref*(d_2) then d_1 is preferable to d_2; (iii) if *pref*(d_1) < *pref*(d_2) then d_2 is preferable to d_1.

Results and discussion

In this section, a practical example applying the framework for consensus decision-making is presented to discuss the results obtained, followed by the related works.

Practical example

Let $FCK = \langle \{ag_1, ag_2, ag_3\}, \{0.4, 0.3, 0.3\}, \{x, y\}, med, 10 \rangle$ and *med* with $\sigma = 0.5$. The problem is a discussion among three agents trying to decide if a robot should rescue a human being in a disaster situation. The robot has a stretcher that can carry only one person at a time. There are two possible decision alternatives for the robot: recharge its battery (x) and make a rescue and take him to the hospital (y). The initial formulas for the agents before the dialogue are:

$$ag_1 = \{\{a, b, \neg c, c \rightarrow d, a \wedge b \rightarrow x, a \wedge b \rightarrow \neg y\}, \{\}\}$$
$$ag_2 = \{\{a, \neg b, c, c \rightarrow d, a \wedge b \rightarrow x, a \wedge b \rightarrow y, a \wedge b \rightarrow \neg d\}, \{\}\}$$
$$ag_3 = \{\{b, c, c \rightarrow d, d \rightarrow \neg x, a \wedge b \rightarrow x, a \wedge b \rightarrow \neg y, d \rightarrow y\}, \{\}\}$$

The atoms in the formulas represent the sentences: a = the battery has less than 70% charge; b = the person is very far from the robot; c = the risk of death is 9 ([0..10] where 10 the person is dead); d = the person is alive.

Agent *med* creates the initial argument to begin the dialogue for the first decision alternative $\text{arg}_1 = \langle \{x\}, x \rangle$. This argument is informed to the group in CS_i.A. For this argument, we have *Support*[a] = {ag_1} where ag_1 has the argument $\langle \{a, b, a \wedge b \rightarrow x\}, x \rangle$ supporting it with *buf*(a) = 0.4. Agent ag_3 has the counter-argument $\text{arg}_2 = \langle \{c, c \rightarrow d, d \rightarrow \neg x\}, \neg x \rangle$ in CS3.S and he is inserted into the *wb*. When enabled to speak, the arguments in CS_3.S is sent and then emptied. The mediator validates the arguments to ensure that all formulas are possibly accepted for the current dialogue (arguments are not repeated and formulas are not rejected nor unconsented). With a valid argument, the dialogue table is updated, the group is informed with the new argument, and the voting stage results in *Support*[c] = {ag_2}, *Reject*[c] = {ag_1}, $Support[c \rightarrow d] = \{ag_1, ag_2\}$, buf(c) = 0.2, $buf(c \rightarrow d) = 1$, $buf(d \rightarrow \neg x) = 0.3$, and $buf(\neg x) = 0.3$. All formulas are updated with the corresponding label.

Agent ag_1 has two counter-arguments[5] $\text{arg}_3 = \langle \{\neg c\}, \neg c \rangle$ and $\text{arg}_4 = \langle \{a, b, a \wedge b \rightarrow x\}, x \rangle$ and he is inserted into the *wb*. In arg_3 we have with $Reject[\neg c] = \{ag_2, ag_3\}$ with $buf(\neg c) = -0.2$. Agents ag_2 and ag_3 has a counter-argument $\langle \{c\}, c \rangle$ but c[x(0)] was presented in a previous argument and was not accepted, making this argument invalid. In arg_4, we have Support[a] = {ag_2} with *buf*(a) = 0.7 indicating that a was accepted by the group, where ag_3 updates his *KO* base with the unknown information.

Agent ag_2 has a valid counter-argument $\text{arg}_5 = \langle \{\neg b\}, \neg b \rangle$ with $Support[\neg b] = \{ag_3\}$ and $Reject[\neg b] = \{ag_1, ag_3\}$, with $buf(\neg b) = -0.1$. As *wb* and *Agenda* are empty, the dialogue is completed. Agent *med* starts the dialogue again for the next decision alternative y. Table 4 shows the dialogue table for decision alternatives x and y. The corresponding argumentation graphs are shown in Figure 2.

The strength of an argument is based on a social model of dialogue, where the more the argument is known, the stronger it becomes, and also the stronger the attacks are. The decision is made by the aggregation of the pros and cons arguments. We have *PRO*(x) = 1.34 and *CON*(x) = 0.85 with *pref*(x) = 0.49, whereas

PRO(y) = 0.72 and *CON*(y) = 2.15 with *pref*(y) = -1.43. With a negative result, the group, by consensus, has a position that is contrary to the choice of y. As *pref*(x) > *pref*(y), in the moment of the dialogue, recharge the battery is preferred by the group considering the current knowledge in each agent.

After all dialogues, the knowledge bases of the agents are:

$$ag_1 = \{\{a[x(0.7), y(1)], b[x(0), y(0)], \neg c[x(0), y(0)], c \to d[x(1), y(1)], a \wedge b \to x[x(1), y(1)],$$
$$a \wedge b \to \neg y[y(0)]\}, \{\neg y[y(0.7)]\}\}$$

TABLE 4.

Arguments, supporting and rejecting votes, and intrinsic strengths for alternatives x and y.

γ	dt_x							dt_y						
	ag	arg	att	Support	Reject	buf	is	ag	arg	att	Support	Reject	buf	is
1	*med*	$\langle\{x\}, x\rangle$		$x[ag_1]$	$x[]$	0.4	0.70	*med*	$\langle\{y\}, y\rangle$		$x[ag_3]$	$x[]$	0.3	0.65
2	ag_3	$\langle\{c, c \to d$ $d \to \neg x\},$ $\neg x\rangle$	1	$c[ag_2]$ $c \to d[ag_1, ag_2]$ $d \to \neg x[]$ $\neg x[]$	$c[ag_1]$ $c \to d[]$ $d \to \neg x[]$ $\neg x[]$	0.2 1.0 0.3 0.3	0.73	ag_1	$\langle\{a, b,$ $a \wedge b \to$ $\neg y\}, \neg y\rangle$	1	$a[ag_2, ag_3]$ $b[ag_3]$ $a \wedge b \to$ $\neg y[ag_3]$ $\neg y[ag_3]$	$a[]$ $b[ag_2]$ $a \wedge b \to$ $\neg y[ag_2]$ $\neg y[]$	1.0 0.4 0.4 0.7	0.81
3	ag_1	$\langle\{\neg c\}, \neg c\rangle$	2	$\neg c[]$	$\neg c[ag_2, ag_3]$	-0.2	0.40	ag_3	$\langle\{c, c \to d,$ $a, d \to \neg x,$ $a \wedge b \to$ $x\}, \neg b\rangle$	2	$c[ag_2]$ $c \to d[ag1, ag_2]$ $a[ag1, ag_2]$ $d \to \neg x[]$ $a \wedge b \to$ $x[ag1, ag_2]$ $\neg b[ag_2]$	$c[ag_1]$ $c \to d[]$ $a[]$ $d \to \neg x[]$ $a \wedge b \to x[]$ $\neg b[ag_1]$	0.2 1.0 1.0 0.3 1.0 0.2	0.81
4	ag_1	$\langle\{a, b,$ $a \wedge b \to x\},$ $x\rangle$	2	$a[ag_2]$ $b[ag_3]$ $a \wedge b \to$ $x[ag_2, ag_3]$ $x[]$	$a[]$ $b[ag_2]$ $a \wedge b \to x[]$ $x[]$	0.7 0.4 1.0 0.4	0.81	ag_1	$\langle\{\neg c\}, \neg c\rangle$	3	$\neg c[ag_3]$	$\neg c[ag_3]$	0.4	0.70
5	ag_2	$\langle\{\neg b\}, \neg b\rangle$	4	$\neg b[ag_3]$	$\neg b[ag_1, ag_3]$	-0.1	0.45	ag_3	$\langle\{\neg y, d \to y,$ $c \to d\}, \neg c\rangle$	3	$\neg y[ag_1, ag_2]$ $d \to y[]$ $c \to d[ag_1, ag_2]$ $\neg c[ag_1]$	$\neg y[]$ $d \to y[]$ $c \to d[]$ $\neg c[ag_2]$	1.0 0.3 1.0 0.4	0.84

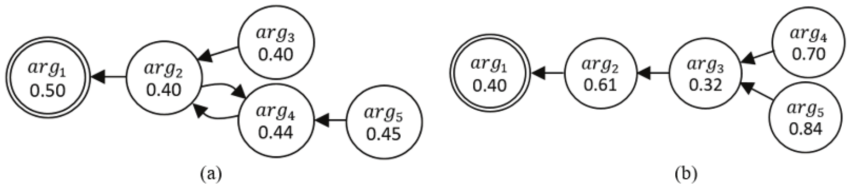

FIGURE 2.

Argumentation Frameworks mapped from dialogue tables dtx (a) and dty (b). Double line circle represents the decision alternative.

$$ag_2 = \{\{a[x(0.7), y(1)], \neg b[x(0), y(0)], c[x(0), y(0)], c \to d[x(1), y(1)], a \wedge b \to x[x(1), y(1)],$$
$$a \wedge b \to y, a \wedge b \to \neg d\}, \{\neg y[y(0.7)]\}\}$$
$$ag_3 = \{\{b[x(0), y(0)], c[x(0), y(0)], c \to d[x(1), y(1)], d \to \neg x[x(0), y(0)], a \wedge b \to x[x(1), y(1)],$$
$$a \wedge b \to \neg y[y(0)], d \to y[y(0)]\}, \{a[x(0.7), y(1)], \neg y[y(0.7)]\}\}$$

Considering the threshold σ, the formulas that were consensually accepted are: a, $c \to d$, $a \wedge b \to x$, and $\neg y$. There were no rejected formulas. Unconsented formulas are: b, $\neg b$, c, $\neg c$, $a \wedge b \to \neg y$, $d \to \neg x$, and $d \to y$. Formulas that were not presented in the dialogues are: $a \wedge b \to y$ and $a \wedge b \to \neg d$. From the consented formulas, we can observe that the battery does not have enough power to make a rescue, so it is preferred by the group to recharge the battery instead of making a rescue and taking the injured person to the hospital. It is important to note that the result shows the preference of the group over a decision alternative. By adopting one of them, a set of actions may be related. For example, when choosing x, some actions could be: choose the closest location to recharge, send a message to an emergency center, help other robots, among others.

Other characteristics of the framework

The proposed framework presents the essential characteristics for the language of knowledge and common knowledge, like those in Fagin et al. (2004):

Proposition 5: $\forall ag_i \in AG$, if $\eta \in \Sigma_i$, then the axiom '$E_{AG}r \leftrightarrow \wedge_{ag_i \in AG} K_i r$' holds for η = r.

Proposition 6: $\forall ag_i \in AG$, $\forall \eta_j \in \arg$, if $\eta_j[d_i(buf)] \in \Sigma_i$ with buf > 0, then the axiom '$(C_{AG}r \wedge C_{AG}(r \rightarrow q)) \rightarrow C_{AG}q$' holds for $\arg = \langle \{r, r \rightarrow q\}, q \rangle$.

Proposition 7: $\forall ag_i \in AG$, if $\eta_j[d_i(buf)] \in \Sigma_i$ with buf > 0, then the axiom '$C_{AG}r \leftrightarrow E_{AG}(r \wedge C_{AG}r)$' holds for η = r, indicating that η is consensually accepted.

Proposition 8: $\forall ag_i \in AG$, if $\eta \in \Sigma_i$, when η is sent in an argument during the dialogue, we get buf(η)= 1, then the axiom 'From $r \rightarrow E_{AG}r$ infer $r \rightarrow C_{AG}r$' holds.

As a result of our framework, we may present some comments:

· In a knowledge base $\Sigma = \{\{\beta, \pi[d(0.5)], \psi\}, \{\neg\psi[d(0.5)], \upsilon[d(0.6)]\}\}$, we have that an agent considers β as true for all possible worlds; $\pi[d(0.5)]$ is also true for all possible worlds, since there is not another π in $\sum$; ψ is true for all possible worlds, except for world *d* which has $buf(\neg\psi) = 0.5$; and $\upsilon[d(0.6)]$ is true for all possible worlds. The framework also represents information that is common knowledge in the group of agents, like π, $\neg\psi$ and v;

· The framework acts as a partial belief model, being able to represent the different belief states for every formula in every possible world;

· Arguments are built and stored in $CS_i.S$ to attack the current argument in the $CS_i.A$. It is important to consider only information the agents have in that moment;

· Every argument in the *Agenda* should be an attack relation (undercut or rebuttal) and it may not have already been presented in the current *DT*. It is necessary to ensure that all arguments are related to each other and are sent only once in a dialogue avoiding repetitions;

· The result of the framework for consensus decision-making indicates the decision alternative that is preferred by the group, not necessarily the optimal one;

· The framework does not allow the blocking of a decision alternative. Blocking is a situation where an agent is not sure to accept an argument, formula, or result of the decision, or then when to implement the decision, there are insufficient resource available;

· The initial knowledge base for each argumentative agent are defined by the application designers.

Related works

Dung (1995) proposed some semantics to determine the admissibility of the arguments, that is, a formal framework to identify conflict outcomes, like preferred or grounded semantics. The idea is to specify sets of acceptable arguments, or extensions. An extension is a set of arguments which can be accepted together. These semantics are used to select arguments without considering support or rejection about a decision alternative or group decision.

In Coste-Marquis, Konieczny, Marquis, and Ouali (2012), there is an argumentation graph with weights in the attack relations and apply Dung's semantics. The authors can determine the last attacked or best defended extensions. Our proposal deals with arguments that receive strengths represented as numeric values, and is applied when a group of agents intends to select the preferred alternative considering the opinions of all the agents.

Pereira et al. (2011) use a belief revision using argumentation that assigns a fuzzy labeling to every argument, permitting to change the agent's mind without removing the previous information forever, allowing recovery when this information turns out to be wrong. In our work, the evaluation is done based on the set of formulas of the argument, all arguments are evaluated, and the agents keep all the information that is acceptable to every possible-world in their knowledge base.

Conclusion

A framework to build common knowledge in a group of agents by using consensus on the formulas of an argument was proposed. The framework uses logical arguments in the message exchange, voting system to identify the consensus level and the argument strength. Semantics were proposed to determine the most preferred alternative. The next step is to compare our results with other frameworks that make use of weighted arguments, since they do not represent the consensus level of a group. Another direction is to work with the expertise value that reflects the subdomains of knowledge in which agents are experts in.

References

Besnard, P., & Hunter, A. (2014). Constructing argument graphs with deductive arguments: a tutorial. Argument & Computation, 5(1), 5-30. doi: 10.1080/19462166.2013.869765

Cayrol, C., & Lagasquie-Schiex, M. C. (2005). Graduality in argumentation. Journal Of Artificial Intelligence Research, 23(1), 245-297. doi: 10.1613/jair.1411

Coste-Marquis, S., Konieczny, S., Marquis, P., & Ouali, M. A. (2012). Selecting extensions in weighted argumentation frameworks. Frontiers in Artificial Intelligence and Applications, 245(1), 342-349. doi: 10.3233/978-1-61499-111-3-342

Dung, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence, 77(2), 321-357. doi: 10.1016/0004-3702(94)00041-X

Fagin, R., & Halpern, J. Y. (1994). Reasoning about knowledge and probability. Journal of the ACM, 41(2), 340-367. doi: 10.1145/174652.174658

Fagin, R., Halpern, J. Y., Moses, Y., & Vardi, M. Y. (2004). Reasoning about knowledge. London, GB: MIT Press. doi: 10.1023/A:1008224520671

Gehrlein, W. V. (2006). Condorcet's paradox. New York, NY: Springer-Verlag Berlin Heidelberg. doi: 10.1007/3-540-33799-7

Grossi, D., & Pigozzi, G. (2014). Judgment aggregation: a primer. In R. Brachman, F. Rossi, & P. Stone (Eds.), Synthesis lectures on artificial intelligence and machine learning (p. 1-151). Williston, Vermont: Morgan & Claypool. doi: 10.2200/S00559ED1V01Y201312AIM027

Neary, M., & Winn, J. (2017). Beyond Public and Private: A Framework for Co-operative Higher Education. Open Library of Humanities, 3(2), 1-36. doi: 10.16995/olh.195/

Parsons, S., & McBurney, P. (2003). Argumentation-based dialogues for agent co-ordination. Group Decision and Negotiation, 12(5), 415-439. doi: 10.1023/B:GRUP.0000003742.50038.d3

Pereira, C. C., Tettamanzi, A. G. B., & Villata, S. (2011). Changing one's mind: Erase or rewind? Possibilistic belief revision with fuzzy argumentation based on trust. IJCAI International Joint Conference on Artificial Intelligence, 1, 164-171. doi: 10.5591/978-1-57735-516-8/IJCAI11-039

Singh, R. K., & Benyoucef, L. (2013). A consensus based group decision making methodology for strategic selection problems of supply chain coordination. Engineering Applications of Artificial Intelligence, 26(1), 122-134. doi: 10.1016/j.engappai.2012.03.013

Notes

[1] Here, the word 'belief' represents a formula containing a fact or rule of inference.

[2] The higher s is, the more skeptical the agents are. The lower s is, the more credulous the agents are.

[3] In our framework, the maximum rejection of an argument results in is(argi) = tsi, provided that at least agi believes in the argument.

[4] The iterative method is inspired in Pereira, Tettamanzi, and Villata (2011).
[5] For simplicity, we omit the process to fulfill CSi.S and CSi.A.