Journal of Universal Computer Science, vol. 13, no. 2 (2007), 329-344
submitted: 31/7/06, accepted: 15/1/07, appeared: 28/2/07

Consensus Determining with Dependencies of Attributes with Interval Values

Michal Zgrzywa
(Technical University of Wroclaw, Poland
michal.zgrzywa@pwr.wroc.pl)

Abstract: In this paper the author considers some problems related to attribute dependencies in consensus determining. These problems concern the dependencies of attributes representing the content of conflicts, which cause that one may not treat the attributes independently in consensus determining. It is assumed that attribute values are represented by intervals. In the paper the author considers the choice of proper distance function. Next, the limitations guarantying determining a correct consensus despite treating the attributes independently are presented. Additionally, the algorithm of calculating the proper consensus in cases when these limitation are not met is introduced.

Keywords: Consensus theory, Conflict, Attribute dependency, Agent, Distributed system
Categories: E.1, H.2.1, I.2.4, I.2.11

1 Introduction

Conflict resolution is one of the most important aspects in distributed systems and multi-agent systems. The resources of conflicts in these kinds of systems come from the autonomy feature of their sites (nodes). This feature means that each site of a distributed or multi-agent system processes its tasks independently and may collect different knowledge [Helpern, 01]. There are several reasons to organize a system in such an architecture [Coulouris, 96]. First of all, information collected in the system is easier to obtain – some sites may be nearer or not as busy as others. Also the reliability of such systems is better – the failure of one node may be compensated by using others. Finally, the trustworthiness of the system may be increased when several agents are investigating the same issue. Unfortunately, there may arise such a situation that for the same task, different sites may generate different solutions. Thus, one deals with a conflict.

In distributed and multi-agent systems three origins of conflicts can be found: insufficient resources, differences of data models and differences of data semantic [Pawlak, 98]. For a semantic conflict one can distinguish the following three its components: *conflict body*, *conflict subject* and *conflict content.* Consensus models, among others, seem to be useful in semantic conflicts solving [Tessier, 01]. The oldest consensus model was worked out by such authors as Condorcet, Arrow and Kemeny [Arrow, 63]. This model serves to solving such conflicts in which the content may be represented by orders or rankings. Models of Barthelemy and Janowitz [Barthelemy, 91], Barthelemy and Leclerc [Barthelemy, 95] and Day [Day, 88] enable to solve such conflicts for which the structures of the conflict contents are n-trees, semillatices, partitions etc. These models are still being developed, e.g. in works

[Wang, 01] or [McMorris, 03]. Also determining consensus for fuzzy opinions is widely considered (among others in [Yager, 01] and [Lee, 02]). The common characteristic of these models is that they are one-attribute, it means that conflicts are considered only referring to one feature. Multi-feature conflicts have not been investigated.

In works [Nguyen, 02a] and [Nguyen, 02b] the author presents a consensus model, in which multi-attribute conflicts may be represented. Furthermore, in this model attributes are multi-valued, what means that for representing an opinion on some issue an agent may use not only one elementary value (such as +, –, or 0) [Pawlak, 98] but a set of elementary values. The author considers also conflicts with opinions represented by intervals of values. Intervals are simulated by sets of neighboring elements. This model enables to process multi-feature conflicts, but attributes are mainly treated as independent. However, in many practical conflict situations some attributes are dependent on others. For example, in a meteorological system attribute *Wind_power* (with values: *weak*, *medium*, *strong*) is dependent on attribute *Wind_speed*, the values of which are measured in unit *m/s* (of course, a set of values may be proposed for each attribute when a meteorological station is not sure about a forecast). This dependency follows that if the value of the first attribute is known, then the value of the second attribute is also known. It is natural that if a conflict includes these attributes then in the consensus the dependency should also take place. The question is: Is it enough to determine the consensus for the conflict referring to attribute *Wind_speed*? In other words, is it true that if some value is a consensus for the conflict referring to attribute *Wind_speed*, then its corresponding value of attribute *Wind_power* is also a consensus for the conflict referring to this attribute? And if it is not true, how one should calculate the proper consensus that fulfils the dependency?

In this paper we consider the answers for mentioned above questions. For this aim we assume some dependencies between attributes and show their influence on consensus determining. We will focus on conflicts where agents' opinions are represented by intervals of values. The original contribution of the article is: adjustment of distance function introduced in [Nguyen, 02a] to the interval structure; introduction of consensus calculating algorithm together with theorem considering its properties; introduction of theorem considering conditions allowing to calculate consensus for each attribute independently.

2 The Outline of Consensus Model

The consensus model which enables processing multi-attribute and multi-valued conflicts has been discussed in detail in works [Nguyen, 02a] and [Nguyen, 02b]. In this section we present only some of its elements with extensions needed for the consideration of attribute dependencies. We assume that some real world is commonly considered by a set of agents that are placed in different sites of a distributed system. The interest of the agents consists of events which occur (or have to occur) in this world. The task of the agents is based on determining the values of attributes describing these events. If several agents consider the same event then they may generate different descriptions (which consist of, for example, scenarios, timestamps etc.) for this event. Thus we say that a conflict takes place. Let us consider the simple example. Let the meteorological system introduced in Section 1 give information about temperature forecast for the next day. Every agent – meteorological station – foresees the weather for a few country regions. Consequently, many stations made a forecast for the central region. The following predictions were gathered: three stations claimed that the temperature will be 24 Celsius degrees and one forecast 25 Celsius degrees. Which prediction should be presented in an official communicate? The intuition suggests that 24 is the correct value, but how could it be formally chosen? In other words, how to calculate the consensus?

For representing ontologies of potential conflicts we use a finite set $\boldsymbol{A}$ of attributes and a set $\boldsymbol{V}$ of attribute elementary values, where $\boldsymbol{V} = \bigcup_{a\in \boldsymbol{A}} V_a$ (V_a is the domain of attribute a). Let $\Pi(V_a)$ denote the set of all possible intervals of elements from set V_a (or single elements when elements in V_a can not be ordered) and $\Pi(V_B) = \bigcup_{b\in B}\Pi(V_b)$. Let $B\subseteq \boldsymbol{A}$, a tuple r_B of type B is a function r_B: $B \rightarrow \Pi(V_B)$ where $r_B(b) \subseteq V_b$ for each $b\in B$. Empty tuple is denoted by symbol ϕ. The set of all tuples of type B is denoted by $TYPE(B)$. The conflict ontology is defined as a quadruple:

$$Conflict\_ontology = <\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{P}, \boldsymbol{F}>,$$

where:

- $\boldsymbol{A}$ is a finite set of attributes, which includes a special attribute *Agent*; a value of attribute a where $a \neq Agent$ is an interval of elements from V_a (or single elements when elements in V_a can not be ordered); values of attribute *Agent* are singletons which identify the agents.
- $\boldsymbol{X}=\{\Pi(V_a)\text{: } a\in \boldsymbol{A}\}$ is a finite set of conflict carriers.
- $\boldsymbol{P}$ is a finite set of relations on carriers from $\boldsymbol{X}$, each relation $P\in \boldsymbol{P}$ is of some type T_P (for $T_P \subseteq \boldsymbol{A}$ and $Agent \in T_P$). Relations belonging to set $\boldsymbol{P}$ are classified into groups of two, identified by symbols "$^+$" and "$^-$" as the upper index to the relation names. For example, if R is the name of a group, then relation R^+ is called the positive relation (contains positive knowledge) and R^- is the negative relation (contains negative knowledge). Positive relations contain tuples representing such descriptions which are possible for events. Negative relations, on the other hand, contain tuples representing such descriptions which are not expected for events. When there is only a positive relation, the upper index may be omitted.
- Finally, $\boldsymbol{F}$ is a set of function dependencies between sets of attributes (further described in Section 3).

The structures of the conflict carriers are defined by means of a distance function between tuples of the same type. In this chapter we will use distance functions that measure the distance between two values (intervals or singleton elements) as the minimal costs of the operation which transforms the first value into the second value. The symbol δ will be used for distance functions.

A consensus is considered within a conflict situation, which is defined as a pair $s = <\{P^+,P^-\}, (A,B)>$ where $A,B\subseteq \boldsymbol{A}$, $A\cap B=\emptyset$, and $r_A\neq\phi$ holds for any tuple $r\in P^+\cup P^-$ (P^+,P^- are relations of $TYPE(\{Agent\}\cup A\cup B)$). The first element of a conflict situation (i.e. set of relations $\{P^+,P^-\}$) includes the domain from which consensus should be chosen, and the second element (i.e. pair (A,B)) presents the schemas of consensus. For a subject e (as a tuple of type A, included in P^+ or P^-) there should be assigned only one tuple of type B. A conflict situation yields a set *Subject*(s) of conflict subjects which are represented by tuples of type A. For each subject e two conflict profiles, i.e. *profile*$(e)^+$ and *profile*$(e)^-$, as relations of $TYPE(\{Agent\}\cup B)$ may be determined. Profile *profile*$(e)^+$ contains the positive opinions of the agents on the subject e, while profile *profile*$(e)^-$ contains agents' negative opinions on this subject.

Definition 1. *Consensus on a subject* $e\in Subject(s)$ *is a pair* $(C(s,e)^+,C(s,e)^-)$ *of 2 tuples of type* $A\cup B$ *which fulfill the following conditions:*

a) $C(s,e)^+_A=C(s,e)^-_A=e$ *and* $C(s,e)^+_B\cap C(s,e)^-_B=\phi$,

b) *The sums* $\sum_{r\in profile(e)^+}\delta(r_B,C(s,e)^+_B)$ *and* $\sum_{r\in profile(e)^-}\delta(r_B,C(s,e)^-_B)$ *are minimal.*

Any tuples $C(s,e)^+$ and $C(s,e)^-$ satisfying the conditions of Definition 1 are called consensuses of profiles *profile*$(e)^+$ and *profile*$(e)^-$, respectively.

Example 1. Let us consider the meteorological system from the beginning of the first section. The ontology of that conflict is the quadruple: <{Agent, Region, Temperature}, {Π(Silesia, Great Poland, Little Poland, Pomerania), Π(-30,-29,…,34,35)}, {Weather$^+$}, ϕ>. We can distinguish one conflict situation: <{Weather$^+$,ϕ}, {Region}→ {Temperature}>. Information about the conflict for the subject Silesia is gathered in Table 1.

Agent	Region	Temperature
station$_1$	Silesia	25
station$_2$	Silesia	25
station$_3$	Silesia	24

Table 1: Relation Weather$^+$.

Column Temperature is in fact the *profile*(Silesia)$^+$. To compare the values of different opinions for Temperature attribute function δ_{Temp} will be used: $\delta_{Temp}(t_1,t_2) = |t_1 - t_2|$. Now, after calculating all the necessary distances, we can use the Definition 1b to determine the consensus for this subject (25 Celsius degrees).

3 Determining Consensus for Attributes with Interval Values

In this article we will focus on determining consensus with interval values. Such a problem may occur when agents disagree about the scope of some object property or some phenomenon. A good example is a conflict where agents argue about the timing of some action. Agents describe their knowledge showing the moment of the action beginning and the moment of action ending. The result of such a conflict should be a time interval, which is the most similar to all agents' propositions.

To allow construction of intervals, an attribute domain must fulfill an important condition. There must exist relation ≤ which orders elements of the domain. Additionally, in this article we impose another condition: all domain elements must be located on one axis. This condition guarantee that the following equation is true: $|v_i - v_j| + |v_j - v_k| = |v_i - v_k|$, when $v_i \leq v_j \leq v_k$. It will help us to focus on interesting domains, for example numbers, which are the most commonly used for intervals.

Intervals will be represented by ordered pairs $[i_*,i^*]$, where i_* is the beginning of the interval and i^* is the ending of the interval, $i_* \leq i^*$. For example, [3,5] represents an interval with scope from 3 to 5, and [5,5] represents a single value 5.

Attributes which domains can not be ordered will only have single values: $[i,i]$, or shortly: i.

Let's see a case of conflict with interval-value attributes.

Example 2. Suppose that out meteorological system gives also a forecast for rain. Stations predict when it is going to rain and propose intervals of time. Again, information about the conflict for the subject Silesia is gathered in Table 2.

Agent	Region	RainingTime
$station_1$	Silesia	[10:00, 12:00]
$station_2$	Silesia	[11:00, 14:00]
$station_3$	Silesia	[11:30, 13:00]

Table 2: Relation $Rain^+$ with intervals of values.

Having a conflict situation, we would like to calculate an optimal consensus. Unfortunately, before determining consensus, there is always an important issue to consider: the choice of a distance function. A distance function calculates how different (or similar) are values proposed by agents. For the same conflict situation, algorithms facilitating different distance functions may return completely different results. It is crucial to choose such a function than is not only easy to calculate but also always returns intuitive results.

Calculating consensus for interval-value attributes is not different in this aspect. Let's consider function $\delta_{Sym\text{-}Dif}(interval_1,interval_2)$ defined as length of span covered by only one of the intervals. It works just fine when intervals are intersecting, e.g.: $\delta_{Sym\text{-}Dif}([1,4],[2,5])=2 < \delta_{Sym\text{-}Dif}([1,4],[3,5])=3$ or $\delta_{Sym\text{-}Dif}([1,4],[2,5])=2 < \delta_{Sym\text{-}Dif}([1,4],[3,6])=4$. But it starts to work unintuitive when intervals has no common span. It does not notice the distance between intervals: $\delta_{Sym\text{-}Dif}([1,3],[4,5]) = \delta_{Sym\text{-}Dif}([1,3],[5,6]) = \delta_{Sym\text{-}Dif}([1,3],[6,7]) = 3$. Because it is wrong to assume that all propositions in conflict will always have common span, we have to use a little more complex distance function.

Let's consider function δ_R, defined as follows:

Definition 2. *Distance δ_R between two intervals r and q equals the sum of:*

- *half of the length of the part of r which is outside of q,*
- *half of the length of the part of q which is outside of r,*
- *the length of span between r and q.*

Function δ_R suits very well for calculating difference between intervals. Let's consider some of its properties (considered also in [Nguyen, 02a] and [Nguyen, 02b], but for intervals simulated by sets of neighboring elements):

- the distance increases while one of the intervals expands on the direction opposite to the second interval: $\delta_R([1,3], [5,7]) < \delta_R([1,3], [5,8])$,
- the distance decreases while one of the intervals expands on the direction to the second interval: $\delta_R([1,3], [5,7]) > \delta_R([1,3], [4,7])$,

- the distance increases while span between the intervals increases: δ_R ([1,3], [5,7]) < δ_R ([1,3], [6,8]),
- the distance between intervals with length 0 – points – equals the regular distance between numbers: δ_R ([3,3], [5,5]) = |3-5| = 2,
- the function keeps its properties for both intersecting and separated intervals.

As one can see, the function returns very intuitive results. Fortunately, it also is very easy to calculate. Let's introduce the following theorem:

Theorem 1. *Let set V contain elements which can be ordered. Let the elements from set V fulfill the equation:* $|v_i - v_j| + |v_j - v_k| = |v_i - v_k|$*, when* $v_i \le v_j \le v_k$*. Let* $r=[r_*,r^*]$ *and* $q=[q_*,q^*]$ *be intervals of elements from set V. The distance* δ_R *between intervals r and q equals:*

$$\delta_R(r,q) = \frac{|q^* - r^*| + |q_* - r_*|}{2}.$$

Proof. In such conditions, there are three possible variants of locating two intervals:

a) intervals are separated: $r_* \le r^* \le q_* \le q^*$,
b) intervals have a common span $r_* \le q_* \le r^* \le q^*$,
c) one interval contains the other $r_* \le q_* \le q^* \le r^*$.

In variant a) we have:

$$\frac{|q^* - r^*| + |q_* - r_*|}{2} = \frac{[q^* - r^*] + [q_* - r_*]}{2} =$$

$$\frac{[(q^* - q_*) + (q_* - r^*)] + [(q_* - r^*) + (r^* - r_*)]}{2} =$$

$$\frac{(q^* - q_*) + 2*(q_* - r^*) + (r^* - r_*)}{2} =$$

$$\frac{r^* - r_*}{2} + \frac{q^* - q_*}{2} + (q_* - r^*).$$

The obtained result corresponds to the definition of function δ_R (Definition 4).

In variant b) we have:

$$\frac{|q^* - r^*| + |q_* - r_*|}{2} = \frac{[q^* - r^*] + [q_* - r_*]}{2} =$$

$$\frac{q_* - r_*}{2} + \frac{q^* - r^*}{2} + 0.$$

Again, the result matches the definition.

In variant c) we have:

$$\frac{|q^* - r^*| + |q_* - r_*|}{2} = \frac{[r^* - q^*] + [q_* - r_*]}{2} =$$

$$\frac{[r^* - q^*] + [q_* - r_*]}{2} + 0 + 0.$$

Finally, also in this variant the definition is fulfilled.

Another advantage of using function δ_R is very quick procedure of determining consensus for single attribute. Let's consider the following theorem:

Theorem 2. *Lets have a conflict situation described by at least one attribute x. Let attribute domain V_x contain elements which can be ordered. Let the elements from both set V_x fulfill the equation: $|v_i - v_j| + |v_j - v_k| = |v_i - v_k|$, when $v_i \leq v_j \leq v_k$. Let the propositions in the conflict be intervals of elements from set V_x. Let function δ_R be used for measuring distance between intervals.*

An interval which starts from a central element among propositions for interval opening and ends in a central element among propositions for interval closing is a consensus in the conflict situation.

Proof. Interval $[i_*,i^*]$ is a consensus if its distance to all propositions (profile X) is the smallest. When we use δ_R, the distance may be transformed to:

$$\delta_R([i_*,i^*],\mathrm{X}) = \sum_{[x_*,x^*]\in \mathrm{X}} \delta_R([i_*,i^*],[x_*,x^*]) = \sum_{[x_*,x^*]\in \mathrm{X}} \left(\frac{|x^* - i^*|}{2} + \frac{|x_* - i_*|}{2}\right) =$$

$$= \frac{1}{2} \sum_{[x_*,x^*]\in \mathrm{X}} |x_* - i_*| + \frac{1}{2} \sum_{[x_*,x^*]\in \mathrm{X}} |x^* - i^*| \,.$$

In work [Zgrzywa, 04] the author proved, that in the same conditions (ordered elements of V which can be located on one axis) consensus calculated for proposition that are singleton elements ($\sum_{x\in \mathrm{X}} |x - i|$) is always the central element. (If there is an even number of propositions, any element between both central propositions is a consensus). Now, our result sum is the smallest when both subsumes are the smallest. The first subsume is in fact the distance of the interval start to all the propositions for interval start. So it will be the smallest if we choose the central element. The second subsume is the distance of the interval end to all the propositions for interval end. Again - it will be the smallest if we choose the central element. Thus, it is true that if an interval starts from a central element among propositions for interval opening, and ends in a central element among propositions for interval closing then it is a consensus.

Now, since we have chosen a satisfying distance function, we can return to Example 2. Using Theorem 2, we can easily determine consensus for the conflict situation in relation Rain$^+$. The forecast for region Silesia should contain information about rain from 11:00 to 13:00.

4 Some Aspects of Attribute Dependencies

In Definition 1, condition b) is the most important. It requires the tuples $C(s,e)^+_B$ and $C(s,e)^-_B$ to be determined in such a way thus the sums $\sum_{r\in profile(e)^+} \partial(r_B, C(s,e)^+_B)$ and $\sum_{r\in profile(e)^-} \partial(r_B, C(s,e)^-_B)$ are minimal. These tuples could be calculated in the following way: for each attribute $b\in B$ one can determine sets $C(s,e)^+_b$ and $C(s,e)^-_b$, which minimize sums $\sum_{r\in profile(e)^+} \partial(r_b, C(s,e)^+_b)$ and $\sum_{r\in profile(e)^-} \partial(r_b, C(s,e)^-_b)$ respectively. This way is an effective one, but it is correct only if the attributes from set B are independent ($\boldsymbol{F}=\phi$). In this section we consider consensus choice assuming that some attributes from set B are dependent on some others. The definition of attribute dependency given below is consistent with those given in the information system model ([Pawlak, 93], [Skowron, 92]):

Definition 3. *Attribute b is dependent on attribute a if and only if there exists a surjective function* $f_b^a: V_a \to V_b$ *for which in conflict ontology* $<\boldsymbol{A},\boldsymbol{X},\boldsymbol{P},\boldsymbol{F}>$ $(f_b^a \in \boldsymbol{F})$ *for each relation* $P \in \boldsymbol{P}$ *of type* T_P *and* $a,b \in T_P$ *the formula* $(\forall r \in P)(\ (r_a=[a_*,a^*]) \Rightarrow (r_b=[f_b^a(a_*), f_b^a(a^*)])\ )$ *is true.*

The dependency of attribute b on attribute a means that in the real world if for some object the value of a is known then the value of b is also known. In practice, owing to this property for determining the values of attribute b it is enough to know the value of attribute a. Instead of $[f_b^a(a_*), f_b^a(a^*)]$ we will write shortly $f_b^a([a_*,a^*])$.

The given definition of attribute dependency is very similar to the concept of function dependencies in the relational model. However, there is an important difference that should be emphasized. The dependency function from Definition 3 transforms single element from the domain of attribute a into one element from the domain of attribute b. The dependency function in relational model transforms an interval of elements from the domain of attribute a (whole proposition r_a) into an interval of elements from the domain of attribute a (whole proposition r_b). Additionally, it is not possible to decompose a relation P into two relations according to a dependency function (like in relation model) because it is not even in the First Normal Form.

Consider now a conflict situation $s=<\{P^+,P^-\},A \to B>$, in which attribute b is dependent on attribute a where $a,b \in B$. Let $profile(e)^+$ be the positive profile for given conflict subject $e \in Subject(s)$. The problem relies on determining consensus for this profile. We can solve this problem using two approaches:

1. Notice that $profile(e)^+$ is a relation of type $B \cup \{Agent\}$. There exists a function from set $TYPE(B \cup \{Agent\})$ to set $TYPE(B \cup \{Agent\} \setminus \{b\})$ such that for each profile $profile(e)^+$ one can assign exactly one set $profile'(e)^+ = \{r_{B \cup \{Agent\} \setminus \{b\}}: r \in profile(e)^+\}$. Set $profile'(e)^+$ can be treated as a profile for subject e in the following conflict situation $s' = <\{P^+,P^-\}, A \to B \setminus \{b\}>$. Notice that the difference between profiles $profile(e)^+$ and $profile'(e)^+$ relies only on the lack of attribute b and its values in profile $profile(e)^+$. Thus one can expect that the consensus $C(s,e)^+$ for profile $profile(e)^+$ can be determined from the consensus $C(s,e)'^+$ for profile $profile(e)'^+$ after adding to tuple $C(s,e)'^+$ attribute b and its value which is equal to $f_b^a(C(s,e)'^+_a)$. In the similar way one can determine the consensus for profile $profile(e)^-$.
2. In the second approach attributes a and b are treated independently. That means they play the same role in consensus determining for profiles $profile(e)^+$ and $profile(e)^-$.

The consensus for profiles $profile(e)^+$ and $profile(e)^-$ are defined as follows:

Definition 4. *The consensus for subject* $e \in Subject(s)$ *considered in situation* $s=<\{P^+,P^-\},A \to B>$ *is a pair of tuples* $(C(s,e)^+, C(s,e)^-)$ *of type* $A \cup B$*, which satisfy the following conditions:*

a) $C(s,e)^{+}_{A}=C(s,e)^{-}_{A}=e$ *and* $C(s,e)^{+}_{B}\cap C(s,e)^{-}_{B}=\phi$,

b) $C(s,e)^{+}_{b}=f^{a}_{b}\,(C(s,e)^{+}_{a})$ *and* $C(s,e)^{-}_{b}=f^{a}_{b}\,(C(s,e)^{-}_{a})$,

c) *The sums* $\sum_{r\in profile(e)^{+}}\partial(r_B, C(s,e)^{+}_{B})$ *and* $\sum_{r\in profile(e)^{-}}\partial(r_B, C(s,e)^{-}_{B})$ *are minimal.*

We are interested in the cases when conditions b) and c) of Definition 4 can be satisfied simultaneously. The question is: Is it true that if set $C(s,e)^{+}_{a}$ is a consensus for profile $profile(e)^{+}_{a}$ (as the projection of profile $profile(e)^{+}$ on attribute a) then set $f^{a}_{b}\,(C(s,e)^{+}_{a})$ will be a consensus for profile $profile(e)^{+}_{b}$ (as the projection of profile $profile(e)^{+}$ on attribute)?

5 Problems of Determining Consensus with Attribute Dependencies

It has been shown that consensus for profiles consisting of independent attributes may be determined separately for each attribute. But is this approach also suitable when there are some dependencies between attributes? Let us consider the following example.

Example 3. Let the meteorological system described in Section 1 give also some information about air humidity and related forecast of comfort level for meteopaths (people who are very sensitive to weather conditions). There is a possibility of conflict in knowledge about air (Air^{+}) which is described using attributes: humidity (from 0 to 100%) and comfort level (bad, good). Indeed, a conflict situation takes place referring to the country region Silesia at 10 AM. Table 3 contains data collected by meteorological stations.

Agent	Region	Time	Humidity	Comfort level
$station_1$	Silesia	10:00 AM	25	Bad
$station_2$	Silesia	10:00 AM	55	Good
$station_3$	Silesia	10:00 AM	85	Bad

Table 3: Relation Air^{+}.

Additionally, attribute Comfort level depends on attribute Humidity:

Humidity	Comfort level
$0 \le h \le 30$	Bad
$30 < h \le 80$	Good
$80 < h \le 100$	Bad

Table 4: Dependency function between Comfort level and Humidity.

Now, let us determine the consensus separately for each attribute. The distance function for attribute Humidity returns the difference between two values, e.g. $\delta_{Humidity}(75\%,35\%) = 40$. The function for attribute "Comfort level" returns the number of ranks between comfort levels, e.g. $\delta_{Comfort\_level}$(bad, good) = 1 and $\delta_{Comfort\_level}$(bad, bad) = 0. After assembling the results, the following tuple may be constructed:

Region	**Time**	**Humidity**	**Comfort level**
Silesia	10:00 AM	55	Bad

Table 5: The result calculated for independent attributes.

Table 4 shows that when humidity is about 55% the comfort level should be good, not bad. Apparently, the obtained result does not fulfill the dependencies between attributes. Therefore it cannot be considered as a consensus for this situation. It would also be incorrect to determine the consensus for humidity attribute and to calculate the comfort level attribute using the dependency function.

As it was shown, dependencies between attributes are not always true in calculated consensus. The limitations guarantying determining a correct consensus for single-element values of attributes were considered in previous works: [Zgrzywa, 04], [Zgrzywa, 05]. In [Zgrzywa, 05_2] and [Zgrzywa, 06] the author found these limitations for values represented by sets. A few difference functions were considered there.

In this article we will focus only on determining consensus with dependencies of interval values of attributes. We will consider the following problems:

1. What is the general algorithm of calculating the optimal consensus.
2. What are the limitations guarantying that dependencies takes place in consensus calculated for separate attributes.

6 Consensus Determining with Dependencies of Interval-Value Attributes

The first issue to consider is if consensus calculated for every attribute separately will always fulfill attributes' dependencies. The following example proves that it is not true.

Example 4. We have a conflict situation in relation P, described in Table 6.

Agent	***a***	***b***
$agent_1$	[1,2]	[11,15]
$agent_2$	[1,3]	[10,11]
$agent_3$	[4,5]	[11,16]

Table 6: The conflict situation in relation P.

Additionally, attribute *b* depends on attribute *a*. The dependency is described in Table 7.

V_a	V_b

1	11
2	15
3	10
4	16
5	11

Table 7: Dependency function between b and a.

We will start from attribute *a*. According to Theorem 2, the consensus for separate attribute *a* is [1,3]. Unfortunately, when we repeat the procedure for attribute *b* it will result with interval [11,15]. As we can see, the dependency is not true ($f_b^a([1,3]) \neq [11,15]$) so our consensus is inconsistent.

The example showed, that it is wrong to determine consensus in the quickest way: to calculate consensus only for attribute *a* (C'_a) and to use $f_b^a(C'_a)$ as consensus for *b*. This pair is not necessary optimal – maybe pair $(f_b^a)^{-1}(C'_b)$ and C'_b would be better? Or some other pair?

Unfortunately, without imposing further conditions, a naive algorithm must be used to find an optimal consensus. Algorithm 1, presented below, tries every possible interval for attribute *a* and its image for attribute *b* and returns the best pair. But the algorithm is better than simple naive algorithm. It does not iterate through whole V_a – it narrows the set to elements which are possible to give the best results. It rejects elements for witch either the element and its image are before the earliest proposition or after the latest proposition.

Algorithm 1. Algorithm of calculating consensus.

Input:

- X – conflict profile,
- f_b^a – dependency function,
- δ_R – similarity function.

Output:

- C_a – consensus for attribute *a*,
- C_b – consensus for attribute *b*.

Step 1. *Sum* = maximum, $i = 1, j = 1$.

Step 2. Prepare set $PropSpan_b$ containing all the elements from V_b grater or inside the smallest proposition for attribute *b* ($\min(X_b)$) and lesser or inside the biggest proposition for attribute *b* ($\max(X_b)$):

$$PropSpan_b = \{x \mid x \in V_b \wedge \min(X_b) = [min_*, min^*] \wedge \max(X_b) = [max_*, max^*] \wedge \wedge min_* \leq x \leq max^*\}.$$

Step 3. Prepare set $Props_a$ containing all the elements from V_a that generates image from $PropSpan_b$:

$$Props_a = \{x \mid x \in V_a \wedge f_b^a(x) \in PropSpan_b\}.$$

Step 4. Prepare order $PropSpan_a$ containing ascending ordered elements from V_a grater or equal than $\min(Props_a)$ and lesser or equal than $\max(Props_a)$:

$$PropSpan_a = \{x \mid x \in V_a \wedge \min(Props_a) \leq x \leq \max(Props_a)\}.$$

Step 5. If j > *count*($PropSpan_a$) then

- $i = i + 1$,

- $j = i$.

Step 6. If i > *count*($PropSpan_a$) then go to step 11.
Step 7. $newSum = \delta_R([PropSpan_a(i), PropSpan_a(j)], X_a) +$
$+ \delta_R(f_b^a[PropSpan_a(i), PropSpan_a(j)], X_b)$
Step 8. If *newSum* < *Sum* then:

- $Sum = newSum$,
- $C_a = [PropSpan_a(i), PropSpan_a(j)]$,
- $C_b = f_b^a[PropSpan_a(i), PropSpan_a(j)]$.

Step 9. j = j + 1.
Step 10. Go to step 5.
Step 11. Return C_a, C_b.

The presented algorithm has the following properties:

Theorem 3. *Lets have a conflict situation described by at least two attributes a and b. Let attribute b depend on attribute a according to function f_b^a. Let attributes' domains V_a and V_b both contain elements which can be ordered. Let the elements from both sets V_a and V_b fulfill the equation: $|v_i - v_j| + |v_j - v_k| = |v_i - v_k|$, when $v_i \le v_j \le v_k$. Let the propositions in the conflict be intervals of elements from sets V_a and V_b. Let function δ_R be used for measuring distance between intervals. It is true that:*

a) *The complexity of Algorithm 1 is $O(n^2)$, where n is the size of V_a.*
b) *The algorithm returns an optimal consensus.*

Proof.

a) The algorithm consists of two parts. In the first part, the algorithm prepares a span of elements which should contain the result. To accomplish this task, it iterates once through V_b and twice through V_a. The complexity of this part is then $O(n)$. In the second part, the algorithm iterates through prepared span, checking all the possible interval ends for all the possible interval starts. As the operation of calculating distance between two intervals is very simple, the complexity of steps 5-11 is $O(n^2)$. Thus, the complexity of whole algorithm is also $O(n^2)$.

b) The algorithm tries every possible interval in the prepared span. Thus, if the optimal consensus lays in the span, the algorithm will find it. Then, it is only necessary to prove that the optimal consensus does not lay outside the span. In steps 2, 3, 4 the algorithm rejects elements for witch both the element and its image are before the earliest proposition or after the latest proposition. If an interval and its image ($tuple_i$) start and end before the earliest proposition, then the distance from $tuple_i$ to all propositions is bigger than the distance from the earliest proposition to all propositions. The situation is analogical for tuples after the latest proposition – they generate bigger distances to all propositions than the latest proposition. As the earliest and the latest proposition lay in the span and will not be rejected, then it is certain that the algorithm will find better tuples than these outside the span. This proves that the algorithm gives the optimal results.

As one can see, the algorithm complexity is not very high – only $O(n^2)$. Unfortunately, *n* is not the count of agents' propositions but is the size of V_a (or its part). It will work very fast if all propositions in a conflict are close to each other. But what if the propositions are scattered through whole V_a? Or elements from V_a are not natural numbers but real numbers with high precision? In the next section we will try to find conditions, which allow to use less calculations during determining an optimal consensus.

7 Conditions Sufficient for Treating Attributes Independently

On the previous section we considered the example, where consensus calculated for each attribute separately did not fulfilled the dependency function. It brought us to the conclusion that generally determining optimal consensus needs time-consuming calculations for both attributes. But is it possible to reduce the complexity of the problem? Let's consider another example.

Example 5. We have a conflict situation in relation P, described in Table 8.

Agent	***a***	***b***
$agent_1$	[2,4]	[12,16]
$agent_2$	[3,3]	[14,14]
$agent_3$	[2,5]	[12,19]
$agent_4$	[3,4]	[14,16]
$agent_5$	[7,8]	[22,23]

Table 8: The conflict situation in relation P.

Additionally, attribute *b* depends on attribute *a*. The dependency is described in Table 9.

V_a	V_b
1	11
2	12
3	14
4	16
5	19
6	21
7	22
8	23

Table 9: Dependency function between b and a.

When we use Algorithm 1 for the conflict, we will get the result [3,4] for attribute *a* and [14,16] for attribute *b*. But when we calculate consensus for each attribute separately (using Theorem 2), the result will be the same. What makes the difference? Let's consider the following theorem.

Theorem 4. *Lets have a conflict situation described by at least two attributes a and b. Let attribute b depend on attribute a according to function* f_b^a*. Let attributes' domains* V_a *and* V_b *both contain elements which can be ordered. Let the elements from both sets* V_a *and* V_b *fulfill the equation:* $|v_i - v_j| + |v_j - v_k| = |v_i - v_k|$*, when* $v_i \leq v_j \leq v_k$*. Let the propositions in the conflict be intervals of elements from sets* V_a *and* V_b*. Let function* δ_R *be used for measuring distance between intervals.*

If function f_b^a *does not change the order of elements* $([v_i \leq v_j] \Rightarrow [f_b^a(v_i) \leq f_b^a(v_j)])$ *then the dependency takes place in consensus calculated for separate attributes.*

The theorem claims, that if a dependency function does not change the order of elements then it is not necessary to calculate consensus using Algorithm 1. In such a case it is enough to calculate consensus only for attribute a (C'_a) and use $f_b^a(C'_a)$ for attribute b.

Proof. Theorem 2 claims, that an interval which starts form the central proposition for interval openings and ends in the central proposition for interval closings is a consensus. This means that if $[a_*,a^*]$ is a consensus for attribute a than a_* is the central proposition for interval openings for attribute a and a^* is the central proposition for interval closings for attribute a. As the dependency function does not change the order of elements, it is true that $f_b^a(a_*)$ is the central proposition for interval openings for attribute b and $f_b^a(a^*)$ is the central proposition for interval closings for attribute b. Thus it is certain that $f_b^a([a_*,a^*])$ is a consensus for attribute b.

As it was shown, it is very easy to determine an optimal consensus when conditions from Theorem 4 are met. Not only it is necessary to calculate consensus for just one attribute, but thanks to Theorem 2 also the determining process itself is very quick. When the conditions from Theorem 4 are not met, Algorithm 1 must be used.

8 Conclusions

In this paper we described how dependencies of interval attributes influence the possibilities of consensus determining. Assuming that δ_R distance function is used, the limitations of dependency functions were shown, guarantying determining a correct consensus despite of treating attributes independently. Using such functions provides the following profits. First of all, they enable determining a consensus for only a part of the attributes (the rest may be calculated using dependency functions). Secondly, they prevent from determining an incorrect consensus, which does not fulfill some of the dependencies of attributes. Additionally, the algorithm of consensus determining was shown, which may be used when the limitations are not met.

Algorithms of consensus determining for conflicts with attribute dependencies may be used to solve many practical problems. One good example is the problem of best book recommendation in internet bookstores. After a customer chooses one book, an algorithm should recommend another. The algorithm uses information about choices of other buyers of the first book. But it could also facilitate the fact, that there is an attribute dependency {book}->{domain}. Sometimes it is better to recommend a book from a very popular domain than a book with a slightly higher number of copies sold but from an unpopular domain. A bookstore with an algorithm facilitating attribute dependencies may generate higher income from cross-selling.

The theorems presented in this paper do not solve all of the problems of this area. The following issues, among other, need to be considered:

- which other limitations are necessary when many attributes may depend on many attributes?
- does using the square of distance in consensus determining influences on presented theorems?
- how to calculate correct consensus for different element structures (orders, divisions, trees)?

Works on these subjects are being continued. The results should enable construction of effective algorithms which will aid conflict resolution in distributed systems.

Acknowledgements

The work was supported by Polish MNiI grant No. N 5 16 033 31/3447.

References

[Arrow, 63] Arrow K.J., Social Choice and Individual Values, Wiley New York, 1963.

[Barthelemy, 91] Barthelemy J.P., Janowitz M.F., A Formal Theory of Consensus, in: SIAM J., Discrete Math vol. 4, 1991, 305–322.

[Barthelemy, 95] Barthelemy J.P., Leclerc B., The Median Procedure for Partitions, in: DIMACS Series in Discrete Mathematics and Theoretical Computer Science vol. 19, 1995, 3–33.

[Coulouris, 96] Coulouris G., Dollimore J., Kindberg T., Distributed systems, Concepts and design, Addison-Wesley, 1996.

[Day, 88] Day W.H.E., Consensus Methods as Tools for Data Analysis, in: Bock H.H. (ed.), Classification and Related Methods for Data Analysis, North Holland, 1988, 312–324.

[Helpern, 01] Helpern J.Y., Moses Y., Knowledge and common knowledge in distributed environment, In: Journal of the Association for Computing Machinery 37, 2001, 549-587.

[Lee, 02] Lee H-S., Optimal consensus of fuzzy opinions under group decision making environment, In: Fuzzy Sets and Systems Vol. 132, Issue 3, 2002, 303-315.

[McMorris, 03] McMorris F.R., Mulder H.M., Powers R.C., The median function on distributive semilattices, In: Discrete Applied Mathematics Vol. 127, Issue 2, 2003, 319-324.

[Nguyen, 02a] Nguyen N.T., Consensus System for Solving Conflicts in Distributed Systems. Information Sciences – An International Journal Vol. 147, 2002, 91-122

[Nguyen, 02b] Nguyen N.T., Methods for Consensus Choice and their Applications in Conflict Resolving in Distributed Systems. Wroclaw University of Technology Press 2002 (in Polish)

[Pawlak, 93] Pawlak Z., A. Skowron A., A Rough Set Approach to Decision Rules Generation, Reports of Institute of Computer Science, Warsaw University of Technology, 1993.

[Pawlak, 98] Pawlak Z., An Inquiry into Anatomy of Conflicts, Information Sciences vol. 108, 1998, 65–78.

[Skowron, 92] Skowron A., Rauszer C., The Discernibility Matrices and Functions in Information Systems, in: E. Słowiński (ed.), Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer Academic Publishers, 1992, 331–362.

[Tessier, 01] Tessier C., Chaudron L., Müller H.J., Conflicting agents: conflict management in multi-agent systems, Kluwer Academic Publishers, Boston, 2001.

[Wang, 01]. Wang J.T.L., Zhang K., Finding similar consensus between trees: an algorithm and a distance hierarchy, in: Pattern Recognition Vol. 34, Issue 1, 2001, 127-137.

[Yager, 01] Yager R.R., Fusion of multi-agent preference orderings, In: Fuzzy Sets and Systems Vol. 117, Issue 1, 2001, 1-12.

[Zgrzywa, 04] Zgrzywa M., Nguyen N.T., Determining Consensus with Simple Dependencies of Attributes, in: N.T. Nguyen (ed.), Intelligent Technologies for Inconsistent Knowledge Processing, Advanced Knowledge International, Australia, 2004, 57–72.

[Zgrzywa, 05] Zgrzywa M., Nguyen N.T., Estimating and Calculating Consensus with Simple Dependencies of Attributes, in: Proceedings of CORES 2005, Rydzyna, Poland, Advances in soft computing, Springer-Verlag, 2005, 319–328.

[Zgrzywa, 05_2] Zgrzywa M., Determining Consensus with Dependencies of Multi-element Attributes, In: Katarzyniak R. (ed.), Ontologies and Soft Methods in Knowledge Management, Advanced Knowledge International, Australia, 2005, 119–136.

[Zgrzywa, 06] Zgrzywa M., Determining Consensus with Dependencies of Multi-value Attributes using Jaccard's Index, in: Proceedings of Multimedia and Network Information Systems 2006, Wroclaw, Poland, 2006, 247–259.