Multiple Expert Knowledge Acquisition: Experimental Investigation of Three Voting Strategies

Boris Omelaenko, Vagan Terziyan (*), Seppo Puuronen (**)

(*) Department of Artificial Intelligence and Information Systems, Kharkov State Technical University of Radioelectronics, Ukraine, e-mail: vagan@milab.kharkov.ua

(**) Department of Computer Science and Information Systems, University of Jyvaskyla, Finland, e-mail: sepi@jytko.jyu.fi

Abstract

Paper deals with voting strategies of knowledge acquisition from multiple experts. It presents part of research, aimed to develop multiple expert knowledge acquisition system, able to produce opinion, most supported by group of experts, possibly remote. Paper presents experimental investigation of three voting strategies based on the dynamic expert ranking technique. Four main components determine the technique's behavior: the way to derive the most supported opinion; the way to evaluate quality of the most supported opinion; the way to recalculate experts' ranks; and the way to organize expert voting process. Last component determines the order of problems to vote about and the way of applying other components of the technique. In this paper, we fix first three of the above components and investigate knowledge acquisition process with three different voting strategies. The first is a real-time one. It proceeds expert opinions in the natural order of their occurrence, and forms most supported opinion in the real time scale. Second one is a batch strategy. It allows experts to repeat their opinions few times to make flexible rank evaluation. Both strategies demand experts' ranks recalculation just after each vote. Third one is a parallel voting strategy. It assumes that experts vote on parts of one compound task and an expert rank is being recalculated only after processing his opinions on all parts of the problem. We investigated how different voting strategies effect the following characteristics of voting process: rank dynamics and rank convergence; most supported opinion and its quality evaluation dynamics; task ordering. Each strategy is illustrated with an appropriate example. It was noticed that the real-time strategy is the simplest one, it takes minimum computer resources to run. Voting results however greatly depend on the sequence of the domain description. Batch strategy gives more precise results than the real-time one. It weakly depends on domain description sequence, but demands more computer resources to work. Parallel strategy does not depend on the order of domain description, but demands maximum resources to produce precise results.

1. Introduction

The area of knowledge management includes the problem of eliciting expertise from more than one expert. The significance of this subject deals with fast development of telecommunications, Internet, WWW that connects people together and gives possibilities to collect knowledge from different remote sources. The problems, how to collect different opinions, handle inconsistent and incomplete knowledge taken from them, find consensus, support interface between individual and collective knowledge, are discussed in [4].

Appeared in: Proccedings of *STeP'98 Human and Artificial Information Processing*, Jyvaskyla, Finland, 7–9 September 1998, Publ. of Finnish AI Society, p. 88-97.

Could the overlapping knowledge from multiple sources be described in such a way that it is context or even process independent? In [7], the negative answer was given. If more than one expert is used, one must either select the opinion of the best expert or pool experts' judgements [7, 3]. It is assumed that when experts' judgements are pooled, collectively they offer sufficient cues to lead to the building of a comprehensive theory.

In practice, one of the following three strategies may be used for knowledge acquisition: use opinion of only one expert; collect opinions of multiple experts, but use them one at a time, or integrate these opinions. Research described in [6] deals mainly with the strategy of integrating opinions. It is assumed that acquired knowledge has more validity if it forms a consensus among the experts. In paper [5], five techniques are discussed and compared for aggregating expertise. In this study, elicited knowledge is aggregated using classical statistical methods (regression and discriminant analysis), the ID3 pattern classification method, the k-NN technique, and neural networks. In aggregating knowledge, the authors try to identify the significance of each extracted factor and the functional inter-relationship among the relevant factors.

Present paper continues research [4]. We present a flexible model for voting-type techniques to work within. Model is presented in Basic concepts section. Four strategies determine techniques behavior and course it's flexibility: strategy of deriving opinion, most supported by experts, strategy of it's quality evaluation, experts rank refinement strategy and voting strategy. All of them are described in Developing techniques section.

We fix first three strategies to investigate how different voting strategies influence voting process and expertise results. For this we proceed real-life experts opinions with two voting strategies, as presented in Experimental investigation section. Voting strategies are compared and paper results are discussed in Conclusions.

2. Basic Concepts

In this chapter, we define basic concepts of the method. We define knowledge about multi-expert knowledge acquisition as sixth $<S, D, Q, V, P, T>$. The concepts used are the following:

$S = \{S_1, S_2, \ldots, S_n\}$ — the set of n knowledge sources or experts. We assign a numeral rank to each expert with the set $r = \{r_1, r_2, \ldots, r_n\}$ of n expert's ranks to measure his authority in domain. Authority and rank is a subject to change during voting process.

$D = \{D_1, D_2, \ldots, D_d\}$ — the set of d domain concepts (domain relations). Domain is structured, and each domain concept consists of m component relations C_i: $D_i = (C_1, C_2, \ldots, C_m)$. Each component takes it's values from corresponding set E: $C_i \in E_i$, discrete or continuos. Usual domains have restrictions on validity of combinations $D_i = (C\ , C\ , \ldots, C\ )$. We define predicate D to describe these restrictions, as follows:

$$D(C_1, C_2, \ldots, C_m) = \begin{cases} 1, \text{ if combination } (C_1, C_2, \ldots, C_m) \text{ is valid in domain } D; \\ 0, \text{ otherwise.} \end{cases}$$

$Q = < Q_1, Q_2, \ldots, Q_q >$ — ordered set of q problems, or questions, asked to experts. It contains all tasks to solve (or questions to ask) during the expertise. Each task $Q_i \in Q$ is a problem, presented somehow.

$V = < V_1, V_2, \ldots, V_q >$ — ordered set of q solutions, or answers on correspondent questions from the set Q; Initially, it contains undefined solutions (answers), with the meaning «no solution found». Each solution $V_i \in V$ must be filled with only one opinion, constructed with technique T on the basis of n expert opinions, considered as the most supported opinion (MSUP). The resulting opinion must belong to the set of all possible domain concepts: $V_i \in D$.

P — semantic predicate, which defines piece of knowledge about temporal relationships in domain by the following relation between the sets Q, D and S:

$$P(Q_i, D_j, S_k) = \begin{cases} 1, \textit{ if the knowledge source } S_k \textit{ uses } D_j \textit{ to solve task (answer question) } Q_i; \\ 0, \textit{ otherwise.} \end{cases}$$

Definition of this predicate is updated just after every expert vote, and it presents all our knowledge about domain and the world.

$T = \{T_1, T_2, \ldots, T_t\}$ — the set of t techniques to proceed expert opinions. Every technique $T_i \in T$ is a fourth, as follows:

$$T_i = \langle MS, QS, RS, VS \rangle,\ T_i \in T$$

Four strategies determine technique behavior: the most supported opinion deriving strategy *MS* defines the way to elicit common knowledge from the set of knowledge sources. The most supported opinion's quality evaluation strategy *QS* defines the way to understand, whether derived common knowledge corresponds to real situation. Rank refinement strategy *RS* defines a way to change expert ranks (to prize or punish experts), according to their opinions, ranks and context. Voting strategy *VS* defines the way of asking questions or giving tasks to experts.

3. Allen Domain of Temporal Intervals

We use domain of Allen's relations between temporal intervals for illustrations. This domain is evidently structured, has restrictions on component combinations; it also has numerous practical implementations.

Component relations for domain concepts are as follows. We define $M = \{C_1, C_2, \ldots, C_m\}$ — the set of m (m=12) basic binary relations for temporal points, as shown in Table 1. In this Table, X^S is starting point and X^F is end point of temporal interval T_1; Y^S and Y^F are endpoints of temporal interval T_2. Component relations are binary, hence $E_i = \{1,0\}$, $i = \overline{1,m}$.

We define domain D according to Allen [1,2] as the set of 13 (d=13) basic relations for temporal intervals. Predicate D defines the correspondence between values $(C_1, C_2, \ldots, C_m)$ and domain concepts D_i, $i = \overline{1,13}$.

Table 1. Set M of basic endpoints' relations

M					
Triad 1: {1,2,3}			*Triad 2: {4,5,6}*		
C_1	C_2	C_3	C_4	C_5	C_6
$X^S<Y^S$	$X^S>Y^S$	$X^S=Y^S$	$X^S<Y^F$	$X^S>Y^F$	$X^S=Y^F$
Triad 3: {7,8,9}			*Triad 4: {10,11,12}*		
C_7	C_8	C_9	C_{10}	C_{11}	C_{12}
$X^F<Y^S$	$X^F>Y^S$	$X^F=Y^S$	$X^F<Y^F$	$X^F>Y^F$	$X^F=Y^F$

4. Developing Strategies

Each technique T consists of four strategies. Each strategy determines one aspect of technique behavior: deriving the most supported opinion, it's quality evaluation, rank refinement and voting order.

4.1. Strategy for Deriving the Most Supported Expert Opinion

Method of deriving the most supported opinion concerning question, or task Q is the following. Experts give their votes about usage of each components of domain concept, seems to be an answer on question Q. Then we make the SC^Q matrix $n \times m$ which defines relationship between the set of knowledge sources S and their opinions about components C_i of answer V on question Q, as follows:

$$\forall S_i \in S,\ \forall D_i = (C_1, C_2, ..., C_m),\ D(C_1, C_2, ..., C_m) \& P(S, Q, D_i) \Rightarrow (SC^Q_{i,q} = C_q),\ q = \overline{1,m}.$$

The technique takes into account the rank of each expert which defines the weight of his vote among all other votes. Let r^v_i will be the rank of i-th expert before v-th voting.

We construct the vector $VOTE^Q$ which contains results of the current experts votings concerning question Q derived from the matrix SC^Q as follows:

$$VOTE^Q_q = \varphi^Q_q - \psi^Q_q, \forall q \in \overline{1,m}, \quad \text{where} \quad \varphi^Q_q = \sum^n_{\substack{i,\\ \forall i (SC^Q_{i,q} = 1)}} r^v_i, \quad \psi^Q_q = \sum^n_{\substack{i,\\ \forall i (SC^Q_{i,q} = 0)}} r^v_i.$$

Vector $VOTE$ can correspond to illegal domain concept due to inconsistent expert knowledge about domain and technique's knowledge about real expert authority in it. We must use domain-specific algorithm to produce correct most supported opinion. Such an algorithm for domain of Allen' temporal relations between pairs of temporal intervals is as presented below.

Temporal tasks Q are questions about endpoints' relations of two temporal intervals a and b. We denote triads of the vector $VOTE^{a,b}$, corresponding to endpoints' relations as $VOTE^{a,b}_{q_1}, VOTE^{a,b}_{q_2}, VOTE^{a,b}_{q_3}$. Then we derive $MSUP^{a,b}$ as the vector which contains the most supported opinion on task Q. Every triad gives one unity to MSUP on the same place with maximum of $VOTE^{a,b}$ in this triad as follows:

$$max_t (VOTE^{a,b}_{q_1, q_2, q_3}) \Rightarrow MSUP^{a,b}_q, t \in \overline{1,4}, \forall q \in \overline{1,m}.$$

If there are more than one maximal vote in a triad then:

(a) if no one of them corresponds to the relation of equivalence between temporal points then there is a conflict between two opinions and we set $MSUP^{a,b}$ as follows:

$$(VOTE^{a,b}_{q_1} = VOTE^{a,b}_{q_2}) \& (VOTE^{a,b}_{q_1} > VOTE^{a,b}_{q_3}) \Rightarrow MSUP *^{a,b}_{q_{1,2}}$$

(b) if one of them corresponds to the relation of equivalence between temporal points then set $MSUP^{a,b}$ as follows:

$$(VOTE^{a,b}_{q_1} = VOTE^{a,b}_{q_3}) OR (VOTE^{a,b}_{q_2} = VOTE^{a,b}_{q_3}) \Rightarrow MSUP^{a,b}_{q_3}.$$

Other domains require another specific algorithms to correct incorrect opinions.

Domain-independent number of conflicts con^v_i between opinion of i-th expert and the most supported opinion, used in rank refinement, is calculated through all set SC during the v-th voting:

$$con^v_i = \sum^m_k (SC^Q_{i,k} \neq MSUP^Q_k),\ \forall Q_i,\ i = \overline{1,q},\ \forall i \in \overline{1,n},\ \forall v \in \overline{1,\infty}$$

4.2. Strategy to Evaluate Quality of the Most Supported Opinion

We introduce parameter Quality to evaluate adequacy of the most supported opinion to real situation. The technique supposes that the quality of resulting opinion is better when the number of votes that are equal to the most supported opinion is large. We make the most supported opinion *quality evaluation* QE by the following way:

$$QE = \frac{\textit{Votes accepted as most supported opinion}}{\textit{All votes}}, \quad QE^Q_v = \frac{\sum^m_k abs(VOTE^Q_k)}{m \cdot \sum^n_i r^v_i}.$$

4.3. Rank Refinement Strategies

Mechanism of expert ranking is used to improve results of voting type processing of the multiple expert's knowledge. The main formula used to refine rank of each expert, as follows:

$$r_i^{v+1} = r_i^v + \Delta r_i^v,$$

where the value of Δr_i^v (punishment or prize value), is equal to

$$\Delta r_i^v = \delta_i^v \cdot \sigma_i^v \cdot \frac{\mu^v - con_i^v}{con}, \quad \sigma_i^v = \frac{v}{v+n-1}, \quad con = \frac{2}{3} \cdot m, \quad \mu^v = \frac{1}{n} \cdot \sum_j^n con_j^v .$$

The value δ_i^v depends on rank refinement strategy selected for an appropriate domain area. The above formulas are based on the following basic assumptions:

- All experts have the same initial rank equal to $\frac{n}{2}$.
- An expert's rank should always be more than zero and less than number of experts.
- After each vote the rank of each expert should be recalculated.
- An expert improves his rank after some vote if his opinion has less conflicts with the most supported one, than the average number of conflicts among all the experts. Otherwise, he loses some part of his rank.
- Expert's rank should not be changed after some vote if expert does not participate it or his opinion has as many conflicts with the most supported one as the average number of conflicts among all the experts.
- The value of expert responsibility (punishment or prize value) grows from one vote to another. [It means that expert cannot loose or improve his rank essentially during the first vote. However, his responsibility will grow accordingly to the multiplier σ_i further.]

We use the strategy «Equal requirements to leaders and outsiders». The main formula used to define the value of δ_i^v is as follows:

$$\delta_i^v = \frac{2 \cdot r_i^v \cdot (n - r_i^v)}{n}.$$

This formula provides the following requirements to the rank refinement strategy:

- The value of punishment (prize) for presence (absence) of each conflict should be maximal for expert with the rank, equal to $\frac{n}{2}$ (n - number of experts).
- The value of punishment (or prize) for presence (or absence) of each conflict should be aspire to zero for expert, whose rank is close to zero or to n.

4.4. Voting Strategies

Three voting strategies were developed during present research to implement three frequently used approaches.

4.4.1. Real-time voting strategy

Real-time voting strategy uses the natural way of questions (tasks) and forces experts to start with the problem Q_1, then continue with Q_2, up to Q_q. Technique produces corresponding common opinions to fill V_1 solution at first, then V_2, up to V_q.

Each problem Q_i forces technique to derive the most supported opinion, it's quality evaluation and to recalculate expert ranks. That is, q problems require q rank recalculations.

4.4.2. Batch strategy

Batch strategy lets experts to vote the same questions few times to repeat their correct or wrong answers. This helps to make more flexible rank evaluation.

We define the sets Q^B and V^B as follows:

$$Q^B = \underbrace{Q \cdot Q \cdot \ldots \cdot Q}_{k\ times},\ V^B = \underbrace{V \cdot V \cdot \ldots \cdot V}_{k\ times},$$

where operation « · » denotes concatenation of two ordered sets.

We use the sets Q^B and V^B instead of sets Q and V in real-time voting process. Technique produces k series of q common opinions. We take last most supported as a final common expert opinion: last q elements of the set V^B will form the resulting set V as follows:

$$V_i = V^B_{q\times(k-1)+i},\ i = \overline{1,q}\ .$$

That is, expert repeats the same answers on the same questions k times. Expert change his rank during such iterative discussion, according to relationship between his opinion and most supported one. This strategy require $k*q$ rank recalculations. The resulting common opinion V is a result of more flexible rank refinement, than in real-time technique.

4.4.3. Parallel strategy

Parallel strategy is very similar to the batch. It differs only in rank recalculation. Experts pass every iteration of q questions without rank changing. Their individual prizes and punishments Δr are summarized, and their mean is final prize (or punishment) Δr, as follows:

$$\Delta r_i = \frac{1}{q}\sum_{j=1}^{q}\Delta r_{i\,j},\ i = \overline{1,n}\ .$$

Parallel voting strategy demands rank recalculations after each iteration. That is, k iterations of q questions each, require k rank recalculations.

This strategy can have another interpretation. We assume, that the whole set Q is one whole task, voted by it's parts. Expert has to vote about all subtasks Q_i from Q, to express his opinion on task Q. We must recalculate his rank on the basis of all q individual opinions, and q most supported opinions, made with constant rank. We cannot change his rank on the basis of one subtask Q_i.

5. Experimental Investigation

Let us consider three examples of three voting strategies. Four experts vote on three tasks from Allen temporal domain. Each expert expressed three opinions on three tasks (q=3), as shown in Table 2. We use three techniques with three different voting strategies to proceed expert opinions.

Table 2. Expert opinions

Expert	1st vote	2nd vote	3rd vote
S_1	T_1 during T_2	T_3 after T_4	T_5 includes T_6
S_2	T_1 overlaps T_2	T_3 meets T_4	T_5 finished by T_6
S_3	T_1 starts T_2.	T_3 overlapped by T_4	T_5 after T_6
S_4	T_1 finished by T_2	T_3 before T_4	T_5 starts T_6

5.1.1. Real-time strategy

Practical example with three first expert votes, taken from Table 2 and 12 other votes, is presented in Figure 1. Here we can see votes on relations from T (1st, 2nd and so on up to 15) on the abscissa axis. Ranks are marked on left Y-axis.

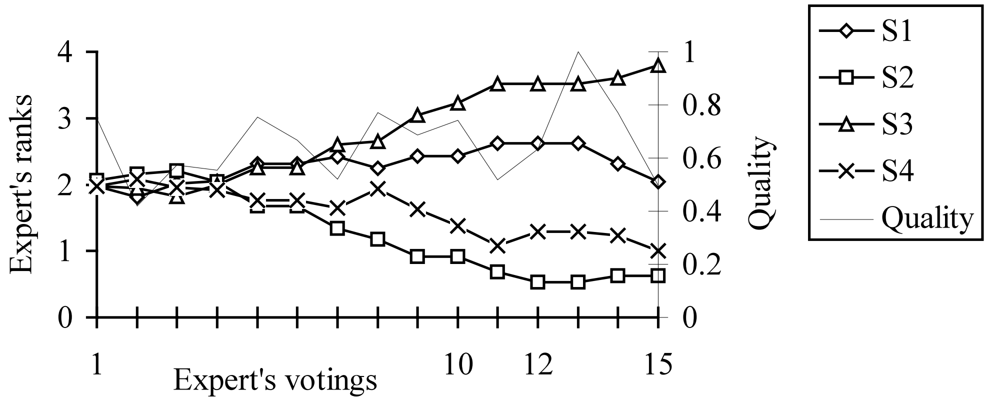

Figure 1. Ranks in real-time strategy, 15 votes

Experts vote on these three tasks one-by-one, corresponding to voting strategy. Experts start with initial ranks, equal to $n/2 = 2$, corresponding to rank refining strategy. Experts rank changes rather slowly, especially during few first votes. That is due to parameter σ in ranking formula. Other 12 votes do not influence on expertise results on three first tasks. They are added to see possible real-life rank changing.

After each vote every expert change his rank. Four lines in Figure 1, marked as S1, S2, S3 and S4 correspond to experts' ranks. One can see that after the 1^{st} vote all experts have rank, close to 2 ($n/2$) corresponding to rank refining principles. One can see that expert S4 seems to be more precise than other experts, so he loosed his rank. Other three experts are seems to be good specialists, but no leader is evident.

Three questions, on which experts tried to answer with their opinions, listed in Table 2, occur again on 10^{th} vote. That is, $Q_1 = Q_{10}$, $Q_2 = Q_{11}$, $Q_2 = Q_{12}$. Hence, all experts have the same opinions on these questions, and we can see how does position of a question in expert's life influence on knowledge, taken as most supported.

Parameter Quality is also shown in Figure 1, marked on the right Y-axis.

Results of experts votes on these three subjects are graphically shown in Figure 2. In Figure 2 all temporal relations, supported by experts are presented in rows with Allen's temporal intervals. Experts opinions on the first vote are listed in column T1, on the second vote — in column T2 and on the third vote — in column T3. Each interval have it's pair, presented in the top row. That is, every opinion is a relation between interval in a corresponding cell and interval in the top of corresponding column. Most supported opinions on every vote are presented in the lowest row. Expert's most supported on questions 10-12 are also presented in Figure 2. New experts ranks course appearance of new opinion, derived and taken as the most supported.

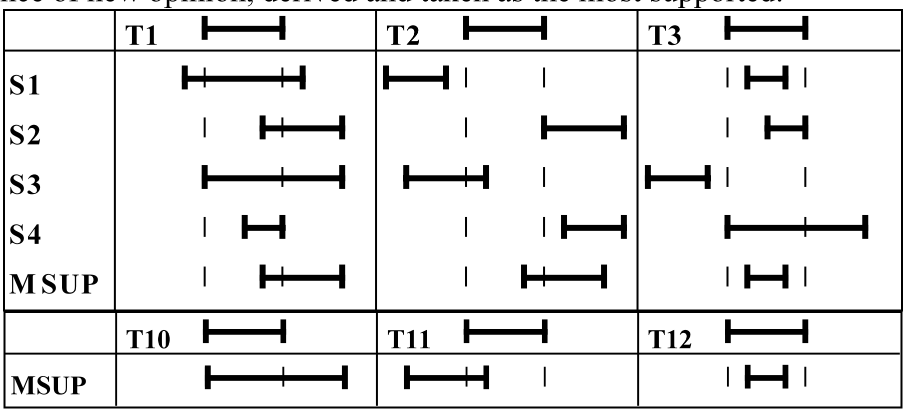

Figure 2. Illustration for real-time strategy

Real-time strategy is the simplest one. It needs minimum computer resources to run. Really, technique runs just one time on each problem, expert vote on. But it has poor rank precision. Experts had very close ranks on votes 1-3. That is due to multiplier σ in rank refinement. This multiplier is an attempt to eliminate strong dependency of experts ranks on tasks order, strategy has.

5.1.2. Batch strategy

Experts ranks changing is illustrated in Figure 3. Experts have close ranks after voting first series. Leaders and outsiders become evident next series. After 5th voting, S2 becomes a leader, his high rank forms Most supported opinion, which coincides his opinion. S4 is improving his rank slower. Experts S1 and S3 lose their ranks and their opinions can not significantly influence on Most supported.

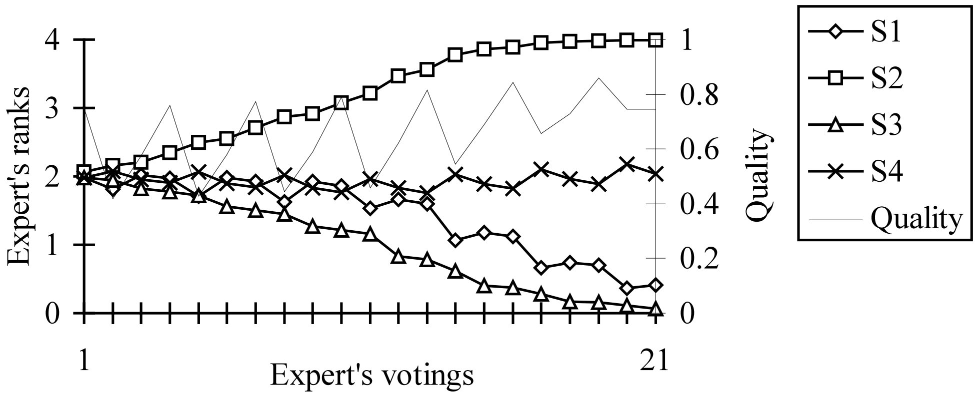

Figure 3. Ranks in batch strategy

Experts opinions and vote results are presented in Figure 4. Here relations are presented in the same manner as in Figure 2.

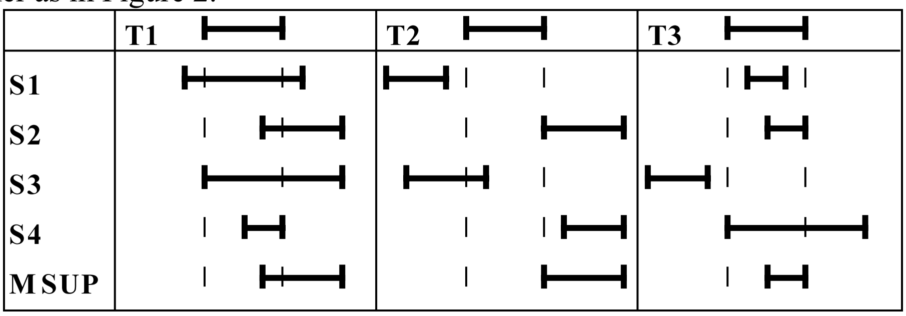

Figure 4. Illustration for batch strategy

Batch technique is more complicated technique, than real-time, and it needs *k* times more computer resources to run. Here *k* is the number of iterations, used to proceed expert's opinions. It has less dependency on domain description order, than real-time strategy. It makes more precise experts ranks — they are close to margin values just after 7 iterations, as shown in Figure 3.

5.1.3. Parallel strategy

We can proceed the same expert opinions with parallel voting strategy. Illustration for most supported opinion after 7 iterations in presented in Figure 5. Rank dynamics is presented in Figure 6. One can see, that expert S3 becomes a leader, under parallel strategy. It was an outsider in batch technique.

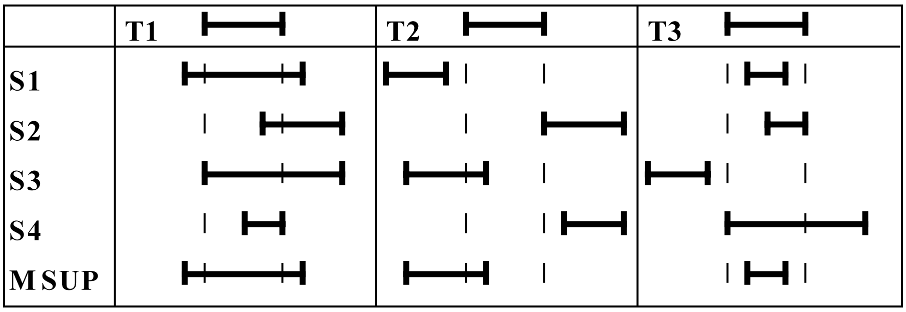

Figure 5. Illustration for parallel strategy

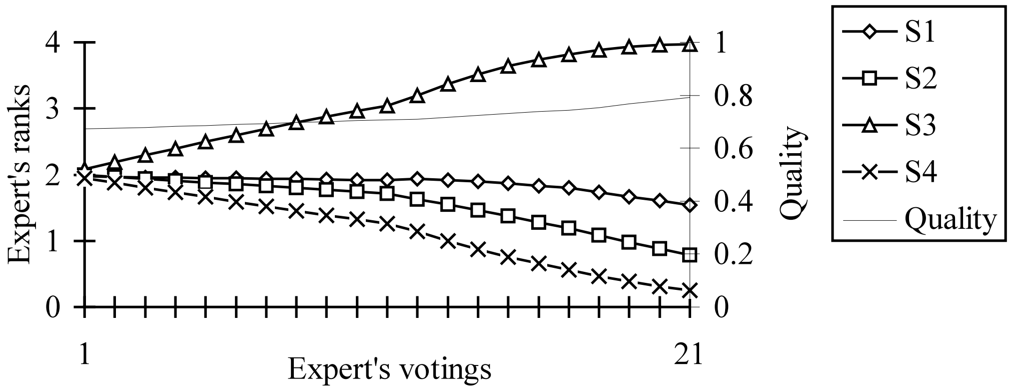

Figure 6. Ranks dynamics under parallel strategy

Parallel strategy does not depend on domain description order. It change ranks slower, than previous techniques. But it needs maximum computer resources for precise ranking. It needs q times more computer time, than batch strategy, and $q*k$ times more, than real-time strategy. Parallel technique allow experts to vote on complex problems, needed to vote by their parts. Expert will vote for all parts of the problem with the same rank, and strategy will recalculate his rank after voting all of problem parts.

5.1.4. Voting results comparison

Let us compare the most supported opinions, obtained after proceeding experts opinions, shown in Table 2 with all three techniques. Comparison results are presented in Table 3.

Table 3. Most supported opinions

Voting strategy	Most supported opinion in graphics		
Real-time	T1 (7) Overlaps	T2 (7) Overlaps	T3 (6) Includes
Batch, 7 iterations	T1 (7) Overlaps	T2 (3) Meets	T3 (12) Finished by
Parallel, 7 iterations	T1 (5) During	T2 (8) Overlapped by	T3 (6) Includes

One can see, that voting results are similar, but different. The greatest difference is between real time and batch technique from one side, and parallel strategy, from the other side. This is coursed by slow rank changing during voting process under parallel strategy. We need to run more iterations to see rank changing under this strategy.

We need to balance between strong rank refinement and available computer resources. Special method for selecting appropriate technique has to be developed in the future.

6. Conclusions

The method of deriving most supported knowledge from set of experts is developed, implemented and experimentally investigated. Flexible rank refinement system evaluate expert authority in domain with expert ranks. Ranks are used to derive most supported opinion, as well as expert opinions. Three voting strategies are developed.

Real-time strategy is the simplest one. It needs minimum computer resources to run. Real-time technique do deriving common opinion and rank recalculation on each problem, expert vote on. It has poor rank precision due to strong dependency of experts ranks on tasks order.

Batch technique uses more complicated batch strategy, than real-time. It needs k times more computer resources to run. Here k is the number of iterations, used to proceed expert's opinions. Greater k gives more precise voting results due to multiplication of expert successes and fails. Strategy has less dependency on domain description order, than real-time strategy and makes more precise experts ranking.

Parallel strategy does not depend on domain description order. It change ranks slower, than batch technique, and it needs maximum computer resources for precise ranking ($q*k$ times more, than real-time strategy). We give only estimated demands to computer resources. Exact demands depend on expert's opinions and other factors and vary greatly.

Voting results are compared. This shows that results depend on voting technique, and we have to develop a context-dependent method to select technique, appropriate to domain, expert stuff and other conditions.

References

[1] Allen, J.F., Maintaining Knowledge About Temporal Intervals, *Communications of the ACM*, V.26, No.11, 1983, pp. 832-843.

[2] Allen, J.F., Temporal Reasoning and Planning, In: Allen J.F. (ed.), *Reasoning About Plans*, Morgan Kaufmann, San Mateo, 1991, pp. 8-33.

[3] Barrett, A.R. and Edwards, J.S., Knowledge Elicitation and Knowledge Representation in a Large Domain with Multiple Experts, *Expert Systems with Applications*, Pergamon, V.8, No.1, 1995, pp. 169-176.

[4] Kaikova, H. and Terziyan, V., Temporal Knowledge Acquisition From Multiple Experts, In: Shoval P. & Silberschatz A. (Eds.), *Proceedings of NGITS'97,* Neve Ilan, Israel, June 30 - July 3, 1997, pp. 44 - 55.

[5] Mak, B., Bui, T. and Blanning, R., Aggregating and Updating Experts' Knowledge: An Experimental Evaluation of Five Classification Techniques, *Expert Systems with Applications*, Pergamon, V.10, No.2, 1996, pp. 233-241.

[6] Medsker, L., Tan, M. and Turban, E., Knowledge Acquisition from Multiple Experts: Problems and Issues, *Expert Systems with Applications*, Pergamon, V.9, No.1, 1995, pp. 35-40.

[7] Taylor, W.A., Weimann, D.H. and Martin, P.J., Knowledge Acquisition and Synthesis in a Multiple Source Multiple Domain Process Context, *Expert Systems with Applications*, Pergamon, V.8, No.2, 1995, pp. 295-302.