**DemocraticOP: A Democratic way of aggregating Bayesian network parameters**

Kobra Etminani 1, Mahmoud Naghibzadeh 2, Jose M. Peña 3

1,2 Dept of computer engineering, Ferdowsi University of Mashhad, Iran

3 ADIT, Department of Computer and Information Science, Linköping University, Sweden

1 corresponding author: etminani@wali.um.ac.ir, tel & fax: 0098-511-8763306

**Abstract**

When there are several experts in a specific domain, each may believe in a different Bayesian network (BN) representation of the domain. In order to avoid having to work with several BNs, it is desirable to aggregate them into a single BN. One way of finding the aggregated BN is to start by finding the structure, and then find the parameters. In this paper, we focus on the second step, assuming that the structure has been found by some previous method.

DemocraticOP is a new way of combining experts’ parameters in a model. The logic behind this approach is borrowed from the concept of democracy in the real world. We assume that there is a ground truth and that each expert represents a deviation from it - the goal is to try to find the ground truth based on the experts’ opinions. If the experts do not agree, then taking a simple average of their opinions (as occurs in classical aggregation functions such as LinOP and LogOP) is flawed. Instead, we believe it is better to identify similar opinions through clustering, and then apply averaging, or any other aggregation function, over the cluster with the highest number of members to obtain the aggregated parameters that are closest to the ground truth. In other words, respect the majority as is done in democratic societies instead of averaging over all experts’ parameters. The new approach is implemented and tested over several BNs with different numbers of variables and parameters, and with different numbers of experts. The results show that DemocraticOP outperforms two commonly used methods, LinOP and LogOP, in three key metrics: the average of absolute value of the difference between the true probability distribution and the one corresponding to the aggregated parameters, Kullback-Leibler divergence, and running time.

**Key words**: aggregation function, Bayesian networks, parameter aggregation.

1. **Introduction**

Bayesian networks (BNs) are a popular graphical formalism for representing probability distributions. A BN consists of structure and parameters. The structure, a directed and acyclic graph (DAG), induces a set of independencies that the represented probability distribution satisfies. The parameters specify the conditional probability distribution of each node given its parents in the structure. The BN represents the probability distribution that results from the product of these conditional probability distributions. Typically, a single expert (or learning algorithm such as [1], [2], and [3]) is consulted to construct a BN of the domain at hand. Therefore, there is a risk that a BN constructed in this way is not as accurate as it could be, e.g. if the expert has a bias or overlooks certain details. One way to minimize this risk is to obtain multiple BNs of the domain from multiple experts and combine them into a single BN. This approach has received significant attention in the literature ([4], [5], [6], [7], [8], [9], [10], [11 [12], [13] and [14]). The most relevant of these references is probably [9], because it shows that even if the experts agree on the BN structure, no method for combining the experts' BNs produces a consensus BN that respects some reasonable assumptions and whose structure is the agreed BN structure. Unfortunately, this problem is often overlooked. To avoid it, we proposed combining the experts' BNs in two steps [12], [4] and [11], by first, finding the consensus BN structure and then finding the consensus parameters for the consensus BN structure.

The consensus Bayesian network structure can be obtained by any existing method. In particular, we recommend [12],[8],[11], because these methods discard the BN parameters provided by the experts and only combine the BN structures provided by the experts, which makes it possible to avoid the problem pointed out by Pennock and Wellman [9]. Even if the experts agree on the BN structure, no method for combining the experts’ BNs (structures + parameters) produces a consensus BN that respects some reasonable assumptions and whose structure is the agreed BN structureHereafter, we assume that the experts have adopted the consensus structure and thus they differ only in the parameters. In this paper, the second step of the BN combination strategy is studied. Specifically, we introduce a new way of pooling the experts’ parameters into one aggregated set of parameters.

In this study, we assume the existence of a ground truth and that experts represent deviations from the ground truth. It seems that this may be a reasonable assumption in many domains such as medicine, where a physical mechanism or system is being modeled. The experts may still disagree due to, for instance, personal beliefs or experiences. However, we do not claim that the assumption of a ground truth is valid in every domain. For instance, if the experts are sport experts that are consulted about the results of the next season, then the idea of the existence of a ground truth is open to discussion, to say the least.

Two commonly used methods for aggregating the BN parameters are linear opinion pool (LinOP) [15] and logarithmic opinion pool (LogOP) [16], which obtain the weighted arithmetic and geometric means respectively. Both methods suffer from two main problems. First, they are slow, since they have to compute the probability of each state of the world and there are exponentially many states. Second, they can be misled by outliers, i.e. non-experts, especially when there is a ground truth and experts believe in a deviation from this truth. One possible solution to the first problem is to do family aggregation, i.e. locally combine the opinions of the experts for each conditional probability distribution [4]. However, this solution still suffers from the second problem. In this paper, we try to address these problems by introducing a novel and smart family aggregation called DemocraticOP.

The main idea of DemocraticOP is to start by clustering the experts’ parameters, and then obtain the aggregated parameter by combining the parameters of those experts who are located in the largest cluster. The underlying logic is to respect the majority, i.e. experts in the largest cluster, and disregard others as they are outliers and would distort the result. Moreover, DemocraticOP is fast because it does not need to compute the probability of each state of the world.

We have implemented DemocraticOP and compared it to two previously well studied methods, LinOP and LogOP, in several BNs with different numbers of variables and parameters, and various numbers of experts as well. It shows superior performance in KL divergence, average of the absolute value of difference between true probability distribution and the one corresponding to the aggregated parameters, and particularly in running time.

The rest of this paper is organized as follows. First the preliminaries are discussed in Section 2. Related works in this area are described in Section 3. We introduce DemocraticOP, its time complexity and the properties it satisfies in Section 4. Section 5 includes the experimental results of implementing three aggregation functions DemocraticOP, LinOP and LogOP, and a comparison of the results with respect to different criteria. Finally, we conclude in Section 6.

1. **Preliminaries**

In this section, the notation used in this paper is introduced. Let *m* be the number of random variables in a BN and be the parent set of each variable. Each variable has possible states and possible parent configurations. Therefore the number of free parameters, *N*, can be calculated as . The set of states of the world is denoted by , where is the total number of states of the world.

The number of experts is denoted by *n*. The experts’ opinions can be aggregated by combining all of their opinions about each state of the world, , or each free parameter, (), where is the probability chosen by expert for event *x*, is one of the possible values of variable , and is one of the possible parent configurations of . The conditional probability is denoted by for short, where *z = 1,…,N.*

Note that is the aggregated parameter and the function *f* which aggregates the experts’ parameters is called the opinion pool:

.

1. **Previous work**

In this section, we explain the background of the various opinion pools and their problems. When combining the parameters of different experts, experts may agree or disagree on the parameters. Experts may all be knowledgeable in the area, i.e. arrive at very similar conclusions. In this situation, combining their parameters is straightforward. Problems arise when there are non-experts, i.e. when there are different values for the parameters. In this case, some problems arise in the existing combination functions as we will see below. Famous combination functions are linear opinion pool (LinOP) [15], logarithmic opinion pool (LogOP) [16], and supra Bayesian approach [17], [18], [19]. For more opinion pools we refer the reader to [20].

LinOp is a weighted arithmetic mean with the following formula:

where is a non-negative weight assigned to each expert and , and . It is also possible to apply LinOP locally or separately within each conditional probability distribution [4]. Pennock and Wellman in [9] called it, in a general case, family aggregation (FA):

=f()

LogOP is a weighted geometric mean with the following formula:

where, again, is a non-negative weight assigned to each expert and , and .

A third pooling method, called the supra Bayesian approach, identifies one distinguished expert *h* (real or fictitious) as the supra Bayesian. The consensus is then defined as the posterior distribution of *h* with Bayes’ rule as the pooling operator:

where is the *h*’s prior probability of state and is the appropriate likelihood of expert *h* for the other experts’ opinions. The problem with the supra Bayesian approach is that finding and is not a simple task, especially when *h* is virtual [20].

Due to the problems described above with the supra Bayesian approach, we focus primarily on LinOP and LogOP. These two commonly used methods suffer from some problems, especially when there is a ground truth and experts represent deviations from it, which we try to solve in this paper. First, the problem with LinOP is discussed. Consider a case where three experts do not agree. One believes the value for a parameter should be 0 and the others believe in 0.7 and 0.8 respectively, where the first expert seems to be an outlier with respect to the ground truth, which we assume is in the vicinity of 0.7-0.8 since most experts’ opinions lie in that vicinity. LinOP then combines them to 0.5 (assuming that the experts are given the same weight). The aggregated parameter is not a good representation of the experts. Note that if the experts believe the parameter should be 0.5, 0.45 and 0.55 respectively, the aggregated parameter also becomes 0.5. In the latter case the experts are satisfied because it is close to them. But when 0.5 is the aggregated value for 0, 0.7 and 0.8, the experts are unsatisfied, because 0.5 is far from the opinions of all of them. Therefore, the logic behind averaging to find a consensus based on several divergent opinions is flawed, since it may include outliers (non-experts). In that case, it seems risky to aggregate the opinions of all the experts, as LinOP does, because there may be non-experts/outliers among them.

This problem also occurs in LogOP, but there is also another problem with LogOP. Consider a case in which all experts believe the probability of visiting Asia is 0.1, except one who believes it is zero. The result of combining these probabilities through LogOP is zero. Again, this does not reflect the aggregation of the experts’ opinions. In other words, a single zero forces the resulting parameter to zero and this result does not reflect the opinions of other experts.

Briefly, the previous methods suffer from two problems. One is that they are slow, since they need to compute the probability of each state of the world. The other is that outliers, which we consider to be non-experts, throw off the results when there is a ground truth. In this paper, a new method is proposed to solve both problems.

The underlying principle of democratic combination can be explained as follows. When there are experts with different parameters, we believe the best way of combining them is to respect the majority. In other words, when most of the experts generally agree on the value of a parameter, there is no need to include the few experts that do not agree with the majority in the final combination. We should only combine those who are in the majority. We propose to first find the experts that generally agree by means of clustering. Then, the parameters of the experts in the largest cluster can be combined. This method of combination is called DemocraticOP in this paper. DemocraticOP is fast because it combines parameters of experts and not states of the world. Furthermore, it is not thrown off track by outliers because it considers only the parameters of experts in the largest cluster.

For instance, in the example for LinOP (three experts with 0, 0.7 and 0.8 values for a parameter), because we believe in a ground truth, we believe that we should trust the last two experts and discard the first one. Then, we output an aggregated parameter of 0.75, whereas LinOP outputs 0.5 and LogOP results in 0. Our output is representative of the experts that we think represent the ground truth. LinOP and LogOP treats experts and non-experts equally while DemocraticOP disregards non-experts, because we assume some experts are not good since they deviate too much from the ground truth.

In [21] Greene and Luger propose a democratic way of combining divergent ideas. This is a work that may look similar to ours, but there are clear differences. Like us, they cluster experts for each parameter. But, in the end, they have several cluster means for each parameter instead of one consensus parameter. Therefore, doing inference in such a consensus Bayesian network requires propagating different cluster means for each parameter throughout the network. In fact, they believe in divergent opinions, not in a fundamental truth as we assume. Therefore, they include all the opinions while we disregard some experts. We also provide a kind of vote for each expert (explained later as the *reliability* concept). Moreover, Greene and Luger did not evaluate their method in different aspects and no comparison to the previous methods is reported, as we do in this paper.

1. **DemocraticOP**

In this section, the democratic combination method is discussed. As stated in Section 3, the principle behind DemocraticOP is to pay attention to the majority as in a democracy.

The algorithm in Figure 1 illustrates DemocraticOP. DemocraticOP combines each parameter separately. To do this, it first builds the similarity graph for each free parameter separately. A similarity graph is a fully connected undirected graph (with edges) in which nodes are experts. Weights of the edges, , for *z*th parameter between expert *l* and *h*, are obtained through the similarity function , which takes and and returns their similarity.

We use the following equation to obtain the similarity:

(1)

From equation (1), if two experts think alike, i.e. , then they are absolutely similar, i.e. becomes 1. On the other hand, if they think totally differently, i.e. one is zero and the other is one, then they are absolutely dissimilar, i.e. becomes 0.

The second step of DemocraticOP is to cluster the experts with the help of the similarity graph for each parameter, which is done by the *Cluster(z)*function. It returns clusters, with *j*=1,…,. We considered the line-island algorithm [22]. It simply clusters the graph by removing edges with weights below a threshold, , and extracts the connected components. Thresholds could be fixed for all parameters or adaptive. Higher thresholds mean we want to be conservative in clustering i.e. prevent very dissimilar ideas from being combined. On the other hand, lower thresholds mean we want to allow more experts to be involved in the aggregation, even some dissimilar ones. If threshold is set to 0, then DemocraticOP will result in family aggregation (FA). In this paper, the average of the weights in the corresponding similarity graph, i.e. , is used as the threshold in each parameter *z*. This means that *t* is set in an adaptive manner with respect to each parameter and the opinions of experts for that parameter. This is not the best value for *t*. There are better thresholds for different cases. However, we observed that the average value works well in all of the domains in our experiments.

|  |
| --- |
| **Inputs**  : experts’ parameters  : experts’ *reliabilities* (it is not compulsory, default value is zero)  **Outputs**  : aggregated parameters |
| **for** *z =* 1 **to** *N*  // build similarity graph for each parameter separately  **for** *l =* 1 **to** *n*  **for** *h = l+*1 **to** *n*    **end for**  **end for**  // then cluster experts for each parameter separately with respect to their similarity graph  = *Cluster(z)*  **end for**  Sort parameters with respect to some criterion.  **for** *z =* 1 **to** *N*  *j* = cluster with maximum and    **for** **each** expert *l* **in**    **end for**  **end for** |

Figure 1. DemocraticOP algorithm.

After the thresholds are identified, the aggregated parameters are calculated. In the previous step, experts were clustered with respect to their similarity in each parameter separately. Therefore, one cluster is the winner, the parameters of the experts in that cluster are combined with a combination function, *f*, and the aggregated parameter is achieved. There are a number of possible combination functions that could be applied. We used the arithmetic average with equal weights in our experiments.

At this point, one question to consider is: from which parameter should one begin, i.e. what is the best parameter ordering to start the aggregation process? One possibility would be to set the order arbitrarily. However, this could create problems. For example, if there is not a single largest cluster, i.e. at least two have the same number of experts in each, as shown in Figure 2, there would be more than one winning cluster. In order to choose one of the largest clusters as the winner, the concept of *reliability* is introduced for each expert (denoted by ) which shows how reliable each expert is. Therefore, in such a case, the cluster with the highest *reliability*, i.e. cluster j with largest , will win. Obviously, when there is only one cluster with the largest number of members, it is most likely going to be the winner. *Reliability* helps in choosing a winner when there is more than one cluster with the largest number of members. One rare thing that may happen is that the largest clusters also have equal *reliabilities*. This means that two or more clusters have the same number of members and the same *reliabilities*. From our point of view, this means that there is not a ground truth in this case, since we assume that there is a ground truth and the experts represent deviations from that truth. Therefore, this case would not happen with our assumption. One solution would be to introduce another metric, but this situation could also occur for this metric. Another solution would be to aggregate the parameters of the “tied” clusters to obtain the resulting parameter. We even tested the latter solution, but it did not lead to a better result. We are working on other solutions to prevent such cases.

Now, we want to find the best parameter ordering to start the aggregation process. Every arbitrary ordering works, but starting from the parameters that most of the experts agree on leads to better and more correct *reliability* values, i.e. values that unambiguously identify which experts deviate from the ground truth and which do not. Therefore, the following heuristic criterion is proposed, and is calculated for each parameter. First, parameters are sorted in a descending order with respect to this criterion, and then DemocraticOP is applied.

(3)

We want to first treat the parameters about which most experts agree and for which there is no need to use *reliability* to find the winning cluster. The criterion returns higher values for such parameters: larger differences between number of members in the largest and the smallest clusters, and a smaller number of clusters, which means more agreement.

Figure 2. More than one cluster wins.

Calculating the *reliability* for each expert is the next matter for discussion. In each step, i.e. for each parameter, after finding the largest cluster and calculating the aggregated parameter, *reliabilities* are then computed. When we have the aggregated parameter, then we can simply compute the expert *reliabilities* with respect to the difference between the aggregated parameter and the expert’s parameter. Function in the algorithm performs the *reliability* computation. We propose the following formula, but other functions are also applicable. Experiments show that this equation performs acceptably.

(2)

When the expert’s parameter is the same as the aggregated one, the newly computed *reliability*, i.e. , will be 1 which means this expert is totally reliable. On the other hand, when an expert’s parameter is 0 but the aggregated parameter is 1 (or vice versa), the newly computed *reliability* becomes 0 which means this expert is non-reliable.

As seen in equation (2), the previously computed *reliability*, , is added to the newly computed one. We suggest only calculating the *reliabilities* of the experts in the largest cluster, but one might update *reliabilities* of all experts in each step or even set some threshold.

If there is previous knowledge about experts’ *reliabilities*, it can be entered into the algorithm. Otherwise, all experts are treated the same initially, i.e. default value is zero which means non-reliable.

* 1. **Time complexity**

The time complexity of DemocraticOP is discussed in this section. Building the similarity graph is of order and it is done for all parameters, hence . Clustering the experts using the line-island algorithm is of order and is done for all parameters, hence . Sorting the parameters with respect to the criteria explained in the previous section, using algorithms like merge sort which behaves the same in all cases, requires *O(N.logN)*. Each step of DemocraticOP to find the aggregated parameter is of *O(n)* order. Since it is done for all parameters, *O(N.n)*. Therefore, the complexity of the whole algorithm is .

This is the time complexity of finding all the aggregated parameters. For comparison, we will discuss the time complexity of LinOP and LogOP. Since both LinOP and LogOP compute the probability of states of the world, all states are traversed. Moreover, in each step, probabilities of all variables for each expert are multiplied. Therefore, the time complexity becomes . In the best case, if all variables are binary, i.e. , then the complexity becomes . Time complexity of FA (LinOP when applied locally on parameters as explained in Section 3) is *O(N.n)*.

Thus, the time complexity of both DemocraticOP and FA is of polynomial order of the number of parameters and experts. However, for LinOP and LogOP, it is of exponential order of the number of variables, because both need to traverse all states of the world. Recall that FA is vulnerable to interference from non-experts (i.e. outliers).

* 1. **Properties of DemocraticOP**

Several researchers propose different properties for the opinion pool [23], [24], [25], [26], [27], [28], [20], [9]. The two incontrovertible properties are Unanimity (UNAM) and Nondictatorship (ND):

*Property 1 (UNAM)*: If for all experts *h* and *l*, and for all states , then .

*Property 2 (ND)*: There is no single expert *h* such that for all , and regardless of the other experts’ opinions.

UNAM says that if all experts agree, then the consensus agrees as well. ND states that no single expert will ever override the others’ opinions. There are several other properties proposed, [9] is a good reference for further reading. It is worth mentioning that there is no aggregation function that can simultaneously satisfy the marginalization property (MP), externally Bayesian (EB), UNAM and ND [25], [9].

Now, we will show that DemocraticOP satisfies both UNAM and ND. It also satisfies other properties such as proportional dependence on states (PDS) [9].

*Proposition 1*. DemocraticOP with average threshold satisfies UNAM.

*Proof*. UNAM says that if all experts agree then the aggregated result must agree. In DemocraticOP, if all experts agree, then they will all be clustered together. Furthermore, the average of n similar opinions will then be the same as one of them, hence the aggregation agrees on the same opinion: .

*Proposition 2*. DemocraticOP with average threshold satisfies ND.

*Proof.* Because in DemocraticOP, the cluster with the largest number of members determines the aggregated parameter, no single expert overrides the others’ opinions. The parameters of the experts in the largest cluster are combined to form the aggregated parameter, not a single expert. Therefore there is no dictatorship. Moreover, there is no winning cluster with only one dictatorial member. Since we consider the average threshold for clustering, the case in which the winning cluster contains only one member will never happen.

This paper is a proof of concept and, as such, we have focused on showing that it works well in practice. In the future, we will study the theoretical properties of DemocraticOP and compare them with those of LinOP. In this paper, we have only proved the two properties that are incontrovertible. Proving the rest of the properties in the literature is beyond the scope of this paper, which was to prove that DemocraticOP works better than LinOP under the assumption of ground truth.

1. **Experimental results**

In this section, DemocraticOP is compared with two well-studied previous methods, LinOP and LogOP, using various different criteria. To create datasets, several BNs are used with different numbers of variables and parameters and various experts. Table 1 shows the properties of the BNs used. All are downloaded from the Norsys website[[1]](#footnote-1). For any given value of , there are free parameters. Since we believe in a ground truth, the downloaded BNs are considered as the ground truth here. We need different experts which believe in a deviation from the ground truth. To create each free parameter produced by the experts, we apply a normal distribution with mean equal to the original parameter and standard deviation (SD) uniformly selected in the interval [0,0.5]. Values below 0 and above 1, created by sampling normal distribution, are truncated to 0 and 1 respectively. Since the result may not be a probability distribution, we normalize the parameters. This process is repeated to create all the free parameters for all the experts.

Two main scenarios are considered to create experts’ parameters. In the first scenario, the same SD is considered for all parameters of each expert. In the second, each parameter for n given expert can have a different SD. The rationale behind these two scenarios is to test the aggregation methods both in cases where experts are equally reliable for all their parameters, and cases when they are not, as in the real world. In reality, sometimes experts are always reliable (as in the first scenario), but there are some experts whose *reliability* varies because they have more experience in some parameters than in others (as in the second scenario). Therefore, we considered both of these cases in our experiments. The number of experts is chosen uniformly in the range [10,20]. All programs are written in Java. Experiments are run on Windows XP with a 2.8 GHz CPU and 1.0 GB of RAM.

Table 1. BN properties (*m* is the number of variables, *N* denotes the number of parameters, and *A* is the number of whole states of the world).

|  |  |  |  |
| --- | --- | --- | --- |
| **BN Name** | ***m*** | ***N*** | ***A*** |
| Animals | 7 | 43 |  |
| Bat habitat basinscale | 8 | 582 |  |
| Asia | 8 | 18 |  |
| forEx consideration | 8 | 475 |  |
| Diabetes learned | 9 | 85 |  |
| B mesenterica alpha | 10 | 46 |  |
| Extending credibility | 12 | 170 |  |
| Bat habitat finescale | 13 | 64 |  |
| Car diagnosis | 18 | 164 |  |
| Midway | 26 | 151 |  |

We used three criteria to compare the performance of DemocraticOP, LinOP and LogOP: the average of the absolute difference between the true probability of each state of the world and the one computed by the aggregated parameters (Diff), the Kullback–Leibler divergence between the true probability distribution and the one corresponding to the aggregated parameters (KL), and running time. All the results were averaged over 10 runs. Diff and KL are calculated using the following formulas:

where is the aggregated probability of the ith state of the world, is the original probability of *i*th state of the world in the downloaded BN, and *A* is the number of the whole states of the world.

* 1. **Number of experts**

We investigated the effect of the number of experts on the KL criterion in LinOP and DemocraticOP. Since LinOP calculates the weighted average, when the number of experts is large, outliers do not affect the result much. Therefore, it is expected that KL for LinOP and DemocraticOP should become more similar as the number of experts grows. On the other hand, when there are fewer experts there is no clear winner. Therefore, it is most interesting to look at cases with an intermediate number of experts. In this case, DemocraticOP seems more effective than LinOP.

The results for the first and the second scenarios (with SDs that are the same and different, respectively) in Asia BN are shown in Figures 3 and 4 for quantities of experts between 2 and 50, with each run once. For the sake of readability, only the figure for this domain is shown. The figures for the other domains indicate the same qualitative conclusions. DemocraticOP is run with two clustering thresholds: 0 and .

As can be seen from both figures, with a smaller number of experts (less than 5) there is not always a clear winner. When the number of experts is between 10 and 30, DemocraticOP wins. As the number of experts grows, LinOP and DemocraticOP converge[[2]](#footnote-2). Recall that LinOP takes much longer to reach such results compared to DemocraticOP, which only takes a few seconds (running times are discussed later).

Figure 3. KL for varying numbers of experts from 2 to 50 in Asia BN for the first scenario (*n* denotes the number of experts).

The figures show that when few or many experts are available, there is no preferred method (except in running time, in which DemocraticOP is the winner). However, for an intermediate number of experts (especially between 10 and 20), DemocraticOP outperforms LinOP. We focus on this range of experts in the coming sections.

Figure 4. KL for varying numbers of experts from 2 to 50 in Asia BN for the second scenario.

* 1. **Diff and KL**

Table 2 illustrates Diff and Table 3 shows KL criteria in the first scenario for DemocraticOP, LinOP and LogOP. Values shown are average ± SD after 10 runs. The best result across the different aggregation methods appears in bold. DemocraticOP is run with two different clustering thresholds, *t*: the average of weights in the corresponding similarity graph () and 0. Recall that *t*=0 results in FA. Other thresholds were also tested that worked better in some cases, hence one can find thresholds that work better than the average of weights. However, this seems to be problem dependent. The average of weights is a threshold that seems to perform well in most of the domains we have considered.

One point that should be mentioned about KL is that when the original probability is not zero but the aggregated probability is zero, then KL becomes infinity because the denominator is zero. To avoid such a case, a tiny integer is used instead of zero. In LogOP, due to the problem discussed earlier (one zero forces the aggregated probability to become zero) cases in which the original probability is not zero but the aggregated probability is zero are more likely to happen. Therefore, values for KL are large in LogOP.

Table 2. Diff for the first scenario averaged after 10 runs (\*p<0.05, \*\*p<0.01).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BN Name** | **LinOP** | **LogOP** | **DemocraticOP (*t*=** ) | **DemocraticOP (*t*=0)** |
| Animals | 9.74e-4±1.02e-4 \* | 1.00e-3±1.70e-4 | **9.24e-4±1.51e-4** | 1.06e-3±9.73e-5 |
| Bat habitat basinscale | 2.58e-5±4.30e-6 \*\* | 2.93e-5±7.88e-6 \*\* | **2.18e-5±4.70e-6** | 2.61e-5±3.52e-6 |
| Asia | 2.96e-3±3.81e-4 \*\* | 3.90e-3±1.14e-3 \*\* | **2.14e-3±5.90e-4** | 3.06e-3±3.84e-4 |
| forEx consideration | 2.73e-6±1.99e-7 | **2.10e-6±4.24e-22** \*\* | 2.65e-6±2.44e-7 | 2.83e-6±1.80e-7 |
| Diabetes learned | 1.63e-4±3.27e-5 \*\* | 2.79e-4±6.55e-5 \*\* | **1.25e-4±3.29e-5** | 1.48e-4±3.09e-5 |
| B mesenterica alpha | 6.75e-5±1.17e-5 \*\* | 1.04e-4±2.31e-5 \*\* | **5.59e-5±1.44e-5** | 5.72e-5±8.72e-6 |
| Extending credibility | 2.03e-7±2.64e-8 \*\* | 1.88e-7±0 \*\* | 1.83e-7±2.12e-8 | **1.72e-7±2.21e-8** |
| Bat habitat finescale | 3.66e-5±6.50e-6 \*\* | 5.84e-5±1.24e-5 \*\* | **2.37e-5±2.79e-6** | 2.93e-5±5.40e-6 |
| Car diagnosis | 2.28e-7±2.53e-8 \* | 2.52e-7±5.72e-8 | **2.04e-7±4.39e-8** | 2.42e-7±2.52e-8 |
| Midway | 6.77e-9±8.84e-10 \*\* | 7.45e-9±8.27e-25 \*\* | **3.87e-9±6.70e-10** | 4.30e-9±6.66e-10 |

In addition, to show that these numbers are not always achieved randomly, p-values of Wilcoxon signed-rank test [29] for this scenario are also computed and shown in Tables 2 and 3 in comparison with DemocraticOP with *t***=** .

Table 3. KL for the first scenario averaged after 10 runs (\*p<0.05, \*\*p<0.01).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BN Name** | **LinOP** | **LogOP** | **DemocraticOP (*t*=** ) | **DemocraticOP (*t*=0)** |
| Animals | 1.11±1.82e-1 | 32.1±4.57 \*\* | **1.04±2.12e-1** | 1.26±1.9e-1 |
| Bat habitat basinscale | 1.07±2.74e-1 \*\* | 27.3±7.87e-2 \*\* | **8.60e-1±2.62e-1** | 1.06±2.22e-1 |
| Asia | 6.14e-1±1.1e-1 \*\* | 20.0±8.78 \*\* | **4.19e-1±1.50e-1** | 6.44e-1±1.11e-1 |
| forEx consideration | **1.72±2.65e-1** | 25.3±3.55e-15 \*\* | 1.75±3.3e-1 | 1.88±2.26e-1 |
| Diabetes learned | 5.10e-1±1.48e-1 \*\* | 28.0±7.60e-1 \*\* | **3.94e-1±1.95e-1** | 4.39e-1±1.31e-1 |
| B mesenterica alpha | 6.48e-1±1.87e-1 \*\* | 26.6±9.3e-1 \*\* | **4.73e-1±1.85e-1** | 5.18e-1±1.14e-1 |
| Extending credibility | 1.26±3.4e-1 \*\* | 20.8±0 \*\* | 1.10±2.9e-1 | **9.5e-1±2.5e-1** |
| Bat habitat finescale | 5.84e-1±1.58e-1 \*\* | 25.8±1.08 \*\* | **3.02e-1±7.51e-2** | 4.60e-1±1.14e-1 |
| Car diagnosis | 1.05±1.96e-1 | 27.2±2.65 \*\* | **9.50e-1±3.65e-1** | 1.18±2.21e-1 |
| Midway | 9.29e-1±2.12e-1 \*\* | 12.6±0 \*\* | **3.42e-1±1.06e-1** | 3.98e-1±1.18e-1 |

Tables 6 and 7 illustrate Diff and KL respectively for LinOP, LogOP and DemocraticOP with two thresholds in the second scenario. The Wilcoxon p-values for this scenario are also shown. From all of these tables, one can see that DemocraticOP with *t*= produces better results with respect to other methods in both of the scenarios considered. In the following subsections, running times are also shown, which emphasize the favorability of DemocraticOP over the others. Although there are a few cases in which the results of DemocraticOP vs. LinOP and LogOP aren’t significant, DemocraticOP still runs much faster (i.e. methods achieve solutions of similar quality but DemocraticOP does so faster).

Table 6. Diff for the second scenario averaged after 10 runs (\*p<0.05, \*\*p<0.01)..

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BN Name** | **LinOP** | **LogOP** | **DemocraticOP (*t*=** ) | **DemocraticOP (*t*=0)** |
| Animals | 1.03e-3±1.15e-4 \*\* | 9.81e-4±1.67e-4 \*\* | **9.18e-4±1.14e-4** | 1.03e-3±1.07e-4 |
| Bat habitat basinscale | 2.91e-5±2.18e-6 \*\* | 2.67e-5±3.39e-21 \* | **2.43e-5±3.22e-6** | 2.76e-5±2.28e-6 |
| Asia | 3.02e-3±5.93e-4 \*\* | 4.24e-3±1.25e-3 \*\* | **2.14e-3±6.76e-4** | 3.00e-3±5.76e-4 |
| forEx consideration | 3.16e-6±1.72e-7 \*\* | **2.31e-6±6.26e-7** \*\* | 2.94e-6±1.66e-7 | 3.05e-6±1.46e-7 |
| Diabetes learned | 1.72e-4±1.31e-5 \*\* | 2.79e-4±6.60e-5 \*\* | **1.49e-4±1.58e-5** | 1.55e-4±1.29e-5 |
| B mesenterica alpha | 6.99e-5±6.45e-6 \*\* | 9.65e-5±1.36e-20 \*\* | **5.26e-5±8.32e-6** | 5.83e-5±4.88e-6 |
| Extending credibility | 2.18e-7±1.16e-8 \*\* | 1.88e-7±0 \* | 1.74e-7±1.63e-8 | **1.68e-7±1.21e-8** |
| Bat habitat finescale | 3.76e-5±5.76e-6 \*\* | 5.43e-5±0 \*\* | **2.57e-5±7.24e-6** | 3.01e-5±4.93e-6 |
| Car diagnosis | 2.76e-7±1.64e-8 \*\* | **2.23e-7±5.29e-23** | 2.24e-7±1.65e-8 | 2.59e-7±1.70e-8 |
| Midway | 7.69e-9±8.84e-10 \*\* | 7.45e-9±8.27e-25 \*\* | **4.36e-9±6.70e-10** | 4.54e-9±6.66e-10 |

Table 7. KL for the second scenario averaged after 10 runs (\*p<0.05, \*\*p<0.01).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BN Name** | **LinOP** | **LogOP** | **DemocraticOP (*t*=** ) | **DemocraticOP (*t*=0)** |
| Animals | 1.24±2.41e-1 \*\* | 33.5±2.91 \*\* | **1.04±1.78e-1** | 1.22±2.17e-1 |
| Bat habitat basinscale | 1.41±1.86e-1 \*\* | 27.3±0 \*\* | **1.03±2.14e-1** | 1.19±1.23e-1 |
| Asia | 6.41e-1±1.80e-1 \*\* | 21.3±8.73 \*\* | **4.31e-1±1.86e-1** | 6.29e-1±1.72e-1 |
| forEx consideration | 2.76±6.26e-1 \*\* | 25.3±3.42e-2 \*\* | **2.13±2.89e-1** | 2.14±2.38e-1 |
| Diabetes learned | 5.57e-1±6.98e-2 \*\* | 28.0±7.29e-1 \*\* | **4.32e-1±9.43e-2** | 4.54e-1±6.28e-2 |
| B mesenterica alpha | 7.33e-1±1.50e-1 \*\* | 26.9±3.55e-15 \*\* | **4.30e-1±1.25e-1** | 5.32e-1±7.95e-2 |
| Extending credibility | 2.19±5.7e-1 \*\* | 20.8±0 \*\* | 1.01±1.6e-1 | **8.9e-1±1.10e-1** |
| Bat habitat finescale | 6.63e-1±1.78e-1 \*\* | 26.2±0 \*\* | **3.69e-1±1.53e-1** | 4.77e-1±1.01e-1 |
| Car diagnosis | 1.70±3.01e-1 | 28.5±0 \*\* | **1.08±1.53e-1** | 1.38±1.66e-1 |
| Midway | 1.54±2.12e-1 \*\* | 16.0±0 \*\* | **3.99e-1±1.06e-1** | 4.32e-1±1.18e-1 |

* 1. **Running Time**

To show the benefit DemocraticOP offers over the other methods, running times of the three methods are shown in Table 10. As discussed before, DemocraticOP finds the aggregated parameters directly, but LinOP and LogOP find the aggregated probability of states of the world. Therefore, the running time of LinOP and LogOP is considerably higher than DemocraticOP. The threshold in DemocraticOP does not affect the running time significantly, therefore it is not reported with different thresholds. Again, times are averaged over 10 runs. The lowest running times across the aggregation methods are shown in bold. As shown in this table, when the number of parameters, variables and the whole states of the world increases, running times of LinOP and LogOP rise exponentially, while DemocraticOP finishes in seconds. This is because of the polynomial order of number of experts and parameters in its time complexity, compared to the exponential order of number of variables for LinOP and LogOP.

The running time of LogOP is good in some cases. This is due to our implementation of it. When multiplying the probabilities to obtain the probability of each state of the world, when we encounter a zero, zero is reported for the result and no more computation is needed. Therefore, running time of LogOP is low in these cases.

Table 10. Running time of the methods averaged after 10 runs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **LinOP** | **LogOP** | **DemocraticOP** |
| Animals | 0.049 **s** | **0.009 s** | 0.018 s |
| Bat habitat basinscale | 13.3 s | 2.1 s | **0.199 s** |
| Asia | 0.009 s | **0.001 s** | 0.006 s |
| forEx consideration | 2.3 min | 18.5 s | **0.200 s** |
| Diabetes learned | 0.362 s | 0.084 s | **0.027 s** |
| B mesenterica alpha | 0.622 s | 0.108 s | **0.020 s** |
| Extending credibility | 15.0 min | 1.7 min | **0.082 s** |
| Bat habitat finescale | 1.5 s | 0.265 s | **0.023 s** |
| Car diagnosis | 12.5 min | 1.9 min | **0.064 s** |
| Midway | 6.3 h | 1.35 h | **4.5 s** |

* 1. **Parameter ordering**

As stated in section 4, before applying DemocraticOP, we sort the parameters according to a criterion. The rationale behind this idea is to first aggregate those parameters where most experts agree i.e. for which there is no need to use *reliability* to find the winning cluster. In this manner, experts’ *reliabilities* are calculated before they are needed. Then in parameters where there is no clear agreement between experts, i.e. there are more than one cluster with largest number of members, the winning cluster is chosen using the computed *reliabilities* of the experts.

If we choose a random ordering, then the experts’ *reliabilities* may not be computed by the time they are needed. In such cases, the winning cluster could be chosen erroneously. Table 11 shows the result of DemocraticOP applied to different BNs for two parameter orderings: the one we have suggested and a random ordering. As can be seen in this table, both Diff and KL in the random ordering are worse than the ones in the suggested ordering, which confirms the importance of good parameter ordering. It is worth mentioning that the two runs reported in the table have been selected because the ordering mattered. This was not always the case: In many runs, both orderings produced the same results.

Table 11. Diff and KL for two parameter orderings in DemocraticOP.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **DemocraticOP with suggested ordering** | | **DemocraticOP with random ordering** | |
| **Name** | **Diff** | **KL** | **Diff** | **KL** |
| Animals | 2.06e-19 | 9.61e-17 | 5.56e-4 | 5.12e-1 |
| Bat habitat basinscale | 1.29e-21 | 2.13e-17 | 4.48e-6 | 1.97 |
| Asia | 7.83e-20 | 2.18e-17 | 4.78e-4 | 1.92e-1 |
| forEx consideration | 1.30e-22 | 4.01e-17 | 2.53e-7 | 1.01 |
| Diabetes learned | 2.69e-20 | 1.28e-17 | 2.19e-5 | 7.29e-1 |
| B mesenterica alpha | 4.54e-21 | 7.36e-18 | 5.78e-6 | 5.14e-2 |
| Extending credibility | 3.25e-23 | 3.39e-17 | 1.50e-8 | 7.02e-2 |
| Bat habitat finescale | 2.02e-21 | 8089e-18 | 2.38e-6 | 2.57e-2 |
| Car diagnosis | 6.06e-23 | 6.64e-18 | 1.74e-8 | 6.62e-2 |
| Midway | 1.09e-24 | 3.92e-18 | 4.17e-10 | 2.83e-2 |

* 1. **DemocraticOP rank**

Now, we want to evaluate how well the combination works with respect to each expert solely in terms of the KL metric. In other words, if we consider the combination as a new expert, what is its rank with regard to the other experts in terms of KL metric? It is desirable that the combination should work better than each expert individually, or that it should at least fall into the top half. That way, we should on average outperform a randomly-chosen expert.

Tables 12 and 13 display the rank of the DemocraticOP expert and the total number of experts, both averaged after 10 runs for the first and second scenario respectively. Since DemocraticOP always works better than LinOP and LogOP in the KL metric, the ranks of those two approaches are worse than DemocraticOP and are not reported.

Table 12. Rank of the DemocraticOP resulting parameters with respect to the other experts in KL metric averaged after 10 runs, in the first scenario (same SD for all parameters of an expert).

|  |  |  |
| --- | --- | --- |
| **Name** | **DemocraticOP rank** | ***n*** |
| Animals | 5.3 | 15.8 |
| Bat habitat basinscale | 3.5 | 14.6 |
| Asia | 1.7 | 12.3 |
| forEx consideration | 2.4 | 14.2 |
| Diabetes learned | 2.4 | 14.9 |
| B mesenterica alpha | 3.0 | 14.5 |
| Extending credibility | 2.8 | 16.0 |
| Bat habitat finescale | 3.5 | 14.1 |
| Car diagnosis | 2.2 | 13.7 |
| Midway | 2.5 | 19.0 |

In the first scenario, each expert has the same SD for each of his parameters. Therefore, if he is reliable (i.e. has a small SD), then he is reliable for all of his parameters. The DemocraticOP expert is almost in the top third, which means it is more reliable than a randomly selected expert (ranked ), hence DemocraticOP can be seen as an expert that performs better than a randomly selected expert. Since there is no way to determine a priori which experts are reliable, the DemocraticOP expert yields good results. If there was a way to determine which expert(s) is the best, there would be no need to combine the opinions of experts.

In the second scenario, each parameter has its own SD. Therefore, all experts are more or less equally reliable. As expected, DemocraticOP almost always ranked first in this scenario. This means that combination of the experts with DemocraticOP works better than each expert individually.

Table 13. Rank of the DemocraticOP parameters with respect to the other experts in KL metric averaged after 10 runs, in the second scenario.

|  |  |  |
| --- | --- | --- |
| **Name** | **DemocraticOP rank** | ***n*** |
| Animals | 1.5 | 14.4 |
| Bat habitat basinscale | 1.0 | 14.3 |
| Asia | 1.0 | 14.7 |
| forEx consideration | 1.0 | 14.3 |
| Diabetes learned | 1.0 | 14.8 |
| B mesenterica alpha | 1.0 | 15.7 |
| Extending credibility | 1.0 | 15.8 |
| Bat habitat finescale | 1.0 | 13.9 |
| Car diagnosis | 1.0 | 13.9 |
| Midway | 1.0 | 12.0 |

* 1. **Experts’ reliabilities**

A secondary output of DemocraticOP is the specification of how reliable the original experts are, in other words, finding out how much they deviate from the ground truth. As discussed earlier, *reliabilities* of experts are computed gradually. *Reliability* is used here to select the winning cluster of parameters when there is more than one largest cluster. At the end of DemocraticOP, there is a value for each expert which shows how reliable that expert is in comparison to the other experts in his parameters. We show in this section that the *reliability* is a good representative of how reliable the expert’s parameters are i.e. how close they are to the ground truth. To prove it, the experts’ *reliabilities* are compared with the SDs (standard deviations) we used to produce their parameters. The higher the SD, the less reliable the expert.

In the second scenario, as discussed earlier, experts are more or less equally reliable. Therefore, no peak is observed in SDs or in *reliability*. Because we create each parameter with a different SD in the second scenario, there is not a single SD for each expert. Therefore, we take the average of SDs over each expert’s parameters and use it as the expert’s SD.

However, in the first scenario, the *reliabilities* are more meaningful, because each expert is stable in his opinion about the parameters, i.e. he has the same SD for all his parameters. Results for the Bat habitat basinscale BN are shown for one of the runs. For the sake of readability, only the figure for this domain is shown. The figures for the other domains lead to the same qualitative conclusions. The horizontal axis shows the experts (17 experts in Figure 5 and 18 experts in Figure 6). Experts’ SDs are shown, which, as explained before, are between 0 and 0.5. *Reliabilities* are normalized. *Reliabilit*y has the opposite meaning to SD. Therefore 1- is drawn for each expert vs. his SD in Figures 5 and 6. As seen, the trend (rise and falls) is the same in all cases. Therefore, DemocraticOP also can be used to find the experts’ *reliabilities* in addition to the aggregated parameters. Therefore, at the end of the algorithm, experts can be ranked with respect to their *reliabilities*.

Figures 5 and 6 also illustrate that a lower threshold (for instance t=0 in figures) leads to a milder trend, thus closer values in (milder ups and downs). On the other hand, a higher threshold (t=) is tougher in making difference between *reliabilities* of experts. We also tested even higher thresholds, which resulted in even larger values for *reliability*.

Figure 5. Expert *reliabilities* vs. experts’ SD in the first scenario for Bat habitat basinscale BN in one run.

Figure 6. Expert *reliabilities* vs. experts’ SD in the second scenario for Bat habitat basinscale BN in one run.

1. **Discussion**

The problem of finding aggregated parameters is studied in this paper. Let , …, denote the experts’ BNs, where is the BN structure and is the set of parameters that expert *l* believes in. It is assumed that the consensus BN structure, , has been found previously through our prior research or any other consensus structure finding algorithm, , where *g* can be any consensus BN structure finding function. Now, we want to find the aggregated parameters, .

In addition, we assume that there is a ground truth and that each expert believes in a deviation from it with respect to his/her experience/belief. This is a reasonable assumption in many domains such as medicine, law, etc, where some mechanism or system exists. However, we do not claim that the assumption of a ground truth is valid in every domain. For instance, if the experts are sport experts that are consulted about the results of the next season, then the idea of the existence of a ground truth is open to discussion, to say the least.

Several aggregation functions have been proposed previously such as LinOP, LogOP, and Supra Bayesian. Each of them suffers from some problems. LinOP and LogOP mainly suffer from averaging the opinions of the experts, which in divergent cases lead to meaningless results. In other words, they create an aggregated parameter that none of the experts believe is true. Moreover, they suffer from long running time, since they need to compute probability of each state of the world. Therefore, a new method of combining experts’ parameters, called DemocraticOP, is presented to address these problems.

The logic behind DemocraticOP is to simulate a real world democracy. When there are different ideas, averaging is not a good means of combination. Consider a case where 8 experts believe a particular treatment must be administered to a patient (i.e. the probability of it is 1) and 2 remaining experts believe the opposite (0). Averaging does not lead to a specific result: whether to administer the treatment or not (0.8). But in DemocraticOP, the opinion on which the majority agrees will win and produce the aggregation. In this example, because 8 experts, i.e. the majority, believe in treatment, the aggregation also believes in it (1).

Furthermore, DemocraticOP combines parameters of experts while LinOP and LogOP aggregate the probability of each state of the world. Therefore, DemocraticOP is fast. Moreover, it leads to one aggregated value for each parameter, thus doing inference in the consensus BN does not require propagation of several values for each parameter.

The concept of *reliability* was also introduced to provide a kind of rating of experts. By default all experts are non-reliable. Gradually their *reliabilities* are updated with respect to their similarity to the aggregated parameter calculated in each step of the algorithm. In the end, experts can be ranked with respect to this new concept. For instance, in the previous example, those 8 experts are more reliable than the two who disagreed.

We have also verified that DemocraticOP satisfies the proposed properties of combination functions. It satisfies UNAM and ND, which are the two incontrovertible properties.

To show that DemocraticOP outperforms LinOP and LogOP, several BNs were used. All results show the benefits of DemocraticOP over the others. This is especially true for running time, where DemocraticOP finishes in seconds while the others may require hours to complete.

We also investigated the effect of the number of experts on the KL metric in DemocraticOP and LinOP. With large and small numbers of experts, there is no difference except in running time. But DemocraticOP works better than LinOP when the number of experts is between 10 and 30. Moreover, DemocraticOP needs only a few seconds to reach the aggregated parameters compared to LinOP and LogOP.

Furthermore, to determine whether the resulting combination performs well enough, DemocraticOP was treated as a fictitious expert, and its ranking of the aggregated parameters was compared to the experts in both of the scenarios in terms of KL metric. The results showed that in the first scenario it ranked in the top third and in the second its rank was nearly first, both better than a randomly selected expert which would fall at the halfway mark.

Additionally, experts’ *reliabilities* are also evaluated vs. their SDs. Specifically, when experts are consistent in their opinion about all their parameters (same SD), one can see a similar trend between their SD and their *reliabilities* obtained through DemocraticOP.

In the future, we are going to extend this idea to the first step of finding the consensus BN, i.e. finding the consensus structure that will allow us to get rid of unreliable experts, i.e. outliers, by finding independencies. The idea of democratic parameter combination works well; we also want to investigate it in the structure, i.e. in the first step of finding the consensus BN. Other future work includes applying the proposed method to a real application in order to show the advantages of DemocraticOP in real world applications.

This paper is a proof of concept and, as such, we have focused on showing that it works well in practice. We are still working on DemocraticOP, addressing issues like what happens in the case of more than one largest cluster with the same *reliability*, what happens with other thresholds in clustering, other clustering methods, the best parameter ordering, and the other properties of DemocraticOP. Our goal is to continue this study where DemocraticOP is presented from a more theoretical perspective, independently of any particular clustering algorithm. In the future, we will study the theoretical properties of DemocraticOP and compare them with those of LinOP. In this paper, we only proved the two properties that are incontrovertible. As previously stated, proving the rest of the properties in the literature is beyond the scope of this paper, which was to prove that DemocraticOP works better than LinOP under the assumption of a ground truth.

1. **Acknowledgements**

This research has been partially supported by ITRC (Iran Telecommunication Research Center) grant with the number 8740/500 and the researchers wish to express their appreciation. The third author is funded by the Center for Industrial Information Technology (CENIIT) and a so-called career contract at Linköping University, and by the Swedish Research Council (ref. 2010-4808).

1. **References**

[1] D. M. Chickering, Optimal Structure Identification with Greedy Search, Journal of Machine Learning Research, 3 (2002) 507-554.

[2] I. Tsamardinos, L. E. Brown, C. F. Aliferis, The max-min hill-climbing Bayesian network structure learning algorithm, Machine Learning 65 (1) (2006) 31-78.

[3] K. Etminani, M. Naghibzadeh, A.R. Razavi, Globally Optimal Structure Learning of Bayesian Networks from Data, Lecture Notes in Computer Science, 6352 (2010) 101-106.

[4] I. Matzkevich, B. Abramson, The Topological Fusion of Bayes Nets, Proceedings of the Eight Conference on Uncertainty in Artificial Intelligence, 191-198, 1992.

[5] I. Matzkevich, B. Abramson, Some Complexity Considerations in the Combination of Belief Networks, Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence, 152-158, 1993.

[6] I. Matzkevich, B. Abramson, Deriving a Minimal I-Map of a Belief Network Relative to a Target Ordering of its Nodes, Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence, 159-165, 1993.

[7] P. Maynard-Reid II, U. Chajewska, Aggregating Learned Probabilistic Beliefs, Proceedings of the Seventeenth Conference in Uncertainty in Artificial Intelligence, 354-361, 2001.

[8] S. H. Nielsen, S. Parsons, An Application of Formal Argumentation: Fusing Bayesian Networks in Multi-Agent Systems, Artificial Intelligence 171 (2007) 754-775.

[9] D. M. Pennock, M. P. Wellman, Graphical Representations of Consensus Belief, Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, 531-540, 1999.

[10] M. Richardson, P. Domingos, Learning with Knowledge from Multiple Experts, Proceedings of the Twentieth International Conference on Machine Learning, 624-631, 2003.

[11] J. del Sagrado, S. Moral, Qualitative Combination of Bayesian Networks, International Journal of Intelligent Systems, 18 (2003) 237-249.

[12] J. M. Peña, Finding Consensus Bayesian Network Structures, Journal of Artificial Intelligence Research, 42 (2011) 661-687.

[13] J. Kracik, Combining marginal probability distributions via minimization of weighted sum of Kullback-Leiblerdivergences, International Journal of Approximate Reasoning, 52(6) (2011) 659-671.

[14] E. Santos Jr., J. T. Wilkinson, and E.E. Santos, Fusing multiple Bayesian knowledge sources, International Journal of Approximate Reasoning, 52(7) (2011) 935-947.

[15] M. stone, The opinion pool, Ann. Math. Statist. 32 (1961) 1339-1342.

[16] M. Bacharach, Scientific disagreement, unpublished manuscript, Christ Church, Oxford, 1972.

[17] R. L. Winkler, The consensus of subjective probability distributions, Manag. Sci., 15 (1968) B61-B75.

[18] P. A. Morris, Decision analysis expert use, Manag. Sci., 20 (1974) 1233-1241.

[19] P. A. Morris, Combining expert judgments: a Bayesian approach, Manag. Sci., 23 (1977) 679-693.

[20] C. Genest, J. V. Zidek, Combining probability distributions: A critique and an annotated bibliography, Statistical Science, 1(1) (1986) 114–148.

[21] K. A. Greene, G. F. Luger, Agreeing to disagree: leveraging consensus and divergence in Bayesian belief aggregation, Proceedings of the AAAI 2009 Spring Symposium. Presented in the Techno-Social Predictive Analytics workshop, 2009.

[22] V. Batagelj, Analysis of large networks-islands, Presented at Dagstuhl seminar 03361: Algorithmic aspects of large and complex networks, 2003.

[23] N. C. Dalkey. Toward a theory of group estimation, In H. A. Linstone and M. Turoff, editors, The Delphi Method: Techniques and Applications, 236–261. Addison-Wesley, Reading, MA, 1975.

[24] C. Genest, Pooling operators with the marginalization property, Canadian Journal of Statistics, 12(2) (1984) 153–163.

[25] C. Genest, A conflict between two axioms for combining subjective distributions, Journal of the Royal Statistical Society, 46(3) (1984) 403–405, 1984.

[26] C. Genest, A characterization theorem for externally Bayesian groups, Annals of Statistics,12(3) (1984) 1100–1105.

[27] C. Genest, K. J. Mc-Conway, M. J. Schervish, Characterization of externally Bayesian pooling operators, Annals of Statistics, 14(2) (1986) 487–501.

[28] C.l Wagner, Aggregating subjective probabilities: Some limitative theorems, Notre Dame Journal of Formal Logic, 25(3) (1984) 233–240.

[29] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics Bulletin 1(6) (1945) 80–83.

1. www.norsys.com [↑](#footnote-ref-1)
2. One may expect both of the methods will converge to 0 in KL metric, but due to the way we create experts, i.e. normalization process and also truncating values less than 0 and above 1 to 0 and 1 respectively, they do not converge to 0, but to a small value. [↑](#footnote-ref-2)