How to efficiently decombine belief functions?

Daira Pinto Prieto[1], Ronald de Haan[1], and Sebastien Destercke[2]

[1] Institute for Logic, Language and Computation, University of Amsterdam
{d.pintoprieto,r.dehaan}@uva.nl
[2] Université de Technologie de Compiègne, CNRS, Heudiasyc
sebastien.destercke@hds.utc.fr

Abstract. Decombination is a useful operator in settings where one has to *forget* or *retract* previously given information. A typical case where this happens is when some inconsistency between information sources is observed, where removing potentially conflicting and unreliable information can restore consistency. Other cases where the removal operator is useful include inferences in graphical models and valuation-based systems, where the decombination operator is routinely used. In each case, repeated decombinations of different pieces of information has to be performed, which is conflicting with the fact that, in belief function theory, such an operator is computationally very costly. In this paper, we show that the decombination operation can actually be performed efficiently, at least as long as the number of focal elements remains limited.

Keywords: Information fusion · Belief functions · Decombination · Algorithm

1 Introduction

Belief functions are a generic language to model and reason with uncertainty, extending both set and probabilistic representations in a unifying framework. As such, belief functions offer a quite expressive language allowing to express uncertainties arising from variability and/or imprecision. It does so by assigning masses of beliefs not only to singletons, but also to subsets of a state space $\mathcal{X}$.

One caveat of this higher expressiveness is that many operations become computationally costly, limiting their applicability. Since the maximal size of the representations increases exponentially with the size of $\mathcal{X}$, a naive application of common operators quickly leads to intractable computations. This is why many works strive to simplify the complexity of operators such as fusion rules [2], computations of belief degrees [11], or inferences in graphical models [15].

One operator that has benefited from less attention, at least from a computational perspective, is the so-called decombination operator [17], where an element of evidence given as a mass function is removed from a global corpus, itself given as a mass function. Indeed, while some works deal with their existence [10] or with approximated decombinations [19], none deals with complexity issues of exact decombination. Such decomposition operators have been used, for example, in non-monotonic reasoning [8], graphical models [14], or more recently the modelling of prejudices [6].

The goal of this paper is to partly fill this gap by investigating the complexity of the decombination operator in specific yet practically relevant situations. More precisely, we will assume that the evidence to retract from the available evidence was one of the pieces of evidence previously combined, therefore ensuring that the decombination operator will result in a standard mass function with positive weights (which is not ensured in the general case [17]). In such a case, we show that while a naive application of the decombination operator leads to exponential complexity, it can also be expressed in a way that sometimes leads to polynomial complexity in the number of focal elements, thereby offering a significant computational gain.

Section 2 gives the necessary basics. Section 3 provides the main results in this paper and presents, when possible, efficient approaches to perform decombination. Section 4 provides a small illustrative use case.

2 Reminders on belief functions and decombination

Belief functions are uncertainty models that generalize both classical probabilities and sets, providing one of the simplest common umbrella for both theories. As a model of uncertainty, belief functions have first emerged from statistical considerations [1], but their use extends to decision theory [7] or machine learning [3], to cite a few fields.

Let $\mathcal{X}$ be a state space, i.e., a set of elements that may form part of our observations. The basic building block to model these observations in the belief function framework is a positive mass function $m : 2^{\mathcal{X}} \to [0,1]$ defined on the power set of $\mathcal{X}$ and that sums up to one, i.e., $\sum_{A \subseteq \mathcal{X}} m(A) = 1$. Given a subset $A \subseteq \mathcal{X}$ and a mass function m such that $m(A) > 0$, the value $m(A)$ represents the degree of certainty of having observed A. From this mass function various uncertainty measures can be derived, the main ones being the belief, plausibility and commonality functions, respectively defined for any event $B \subseteq \mathcal{X}$ as

$$\mathrm{bel}_m(B) = \sum_{A \subseteq B} m(A); \quad \mathrm{plau}_m(B) = \sum_{A \cap B \neq \emptyset} m(A); \quad \mathrm{q}_m(B) = \sum_{B \subseteq C} m(C). \quad (1)$$

The belief and plausibility measures have a rather direct logical interpretation: bel(B) measures how much B is implied, or follows from the available evidence, as it sums up the weights of events that imply B; while plau(B) measures how much B is consistent with our information, as it sums up the weight of events that do not contradict B. On the other hand, commonality is harder to interpret, but could be considered as how common, or how unsurprising, B is. It also plays a very important role in the problem we will consider in this paper.

For easiness, we will also assume that all combined masses are non-dogmatic, i.e., that the mass $m(\mathcal{X}) > 0$ (our results still hold if we assume it only for the mass to be decombined). Note that while this is theoretically restrictive, this is practically not that restrictive and also relevant, as it ensures that $\text{plau}(\{\omega\}) > 0$ for any $\omega \in \mathcal{X}$, meaning that all possible observations are plausible to some degree.

When having several mass functions, the belief function framework provides several ways to merge them and get a combined mass function. The most renowned one is the combination method known as normalized Dempster's rule of combination [13]. Given two mass functions m_1 and m_2, and a set $A \subseteq \mathcal{X}$, this rule can be applied to get their combination $m = m_1 \oplus_K m_2$ as follows:

$$(m_1 \oplus_K m_2)(A) = \begin{cases} 0 & \text{if } A = \emptyset, \\ \frac{1}{K} \sum_{B \cap C = A} m_1(B) \cdot m_2(C) & \text{otherwise} \end{cases} \tag{2}$$

where

$$K = \sum_{B \cap C \neq \emptyset} m_1(B) \cdot m_2(C). \tag{3}$$

Another well-known combination rule of this framework is the *unnormalized* Dempster's rule of combination [18]. In this case, given two mass functions m_1 and m_2, their combination $m = m_1 \oplus m_2$ will be computed following Equation (4).

$$(m_1 \oplus m_2)(A) = \sum_{B \cap C = A} m_1(B) \cdot m_2(C). \tag{4}$$

As it is observed, the main difference between Equation (2) and Equation (4) is the lack of a normalization factor. This allows the mass functions obtained by the unnormalized Dempster's rule of combination to give some positive value to the empty set while satisfying $\sum_{A \subseteq \mathcal{X}} m(A) = 1$. The mass $m(\emptyset)$ can then be interpreted as an estimation of the amount of conflict [4].

Once we obtain a combined mass function by any of the previous methods, we can use Equation (1) to get the correspondent functions. However, the commonality function associated with both $m_1 \oplus_K m_2$ and $m_1 \oplus m_2$ can be computed directly from m_1 and m_2 (Equation (5) and Equation (6) respectively).

$$q_{m_{\oplus_K}}(A) = \frac{1}{K} \cdot q_{m_1}(A) \cdot q_{m_2}(A) \tag{5}$$

where K is given by Equation (3)[3]. In the unnormalised case, we have

$$q_{m_\oplus}(A) = q_{m_1}(A) \cdot q_{m_2}(A). \tag{6}$$

From the commonality function of a mass function m, we can compute the belief function of m (Equation (7)) and also the mass function m (Equation (8)). Plausibility is easily computed from belief, since $\text{plau}(B) = 1 - \text{bel}(\mathcal{X} \setminus B)$.

[3] Note that K can be expressed using commonalities, using Equation (10) below.

$$\text{bel}_m(A) = \sum_{B \subseteq \mathcal{X} \setminus A} (-1)^{|B|} \cdot q_m(B) \tag{7}$$

$$m(A) = \sum_{A \subseteq B} (-1)^{|B-A|} \cdot q_m(B) \tag{8}$$

Therefore, mass functions and commonality functions are equivalent representations in this framework. Although the interpretation of commonality functions is less straightforward, they can be used to define a *decombination operator*.

Definition 1 (Decombination operator (a.k.a. removal operator [14])). *Given a set of possible states $\mathcal{X}$, two mass functions m_1 and m_2, and the combination of them $m = m_1 \oplus m_2$, the* decombination operator $\ominus$ *is defined as follows:*

$$q_{(m \ominus m_2)}(A) = \frac{q_m(A)}{q_{m_2}(A)} \tag{9}$$

Or equivalently,

$$(m \ominus m_2)(A) = \sum_{A \subseteq B} (-1)^{|B-A|} \cdot \frac{q_m(B)}{q_{m_2}(B)} \tag{10}$$

Multiplying Equation (9) and Equation (10) by K, we get the decombination operator for the normalized Dempster's rule of combination.

Another well-established variation of the normalized Dempster's rule of combination is the *disjunctive rule of combination* [16]. While the normalized and unnormalized Dempster's rules of combination assume that every source of evidence is reliable, the disjunctive rule of combination only assumes that at least one of the sources is reliable. To capture this notion, this rule takes into account the union of focal elements instead of the intersection:

$$(m_1 \circledcirc m_2)(A) = \sum_{B \cup C = A} m_1(B) \cdot m_2(C) \tag{11}$$

In [2], a decombination operator for the disjunctive rule of combination is presented. However, it considers stronger assumptions than ours, as it is defined for m_2 such that $\text{bel}_{m_2}(A) > 0$ for every A, including the empty set.

Being able to *forget* or *regret* information is key for an evidence model, and for reasoning in the presence of inconsistency in general. The decombination operator brings this feature to the belief function framework. However, in spite of its mathematical simplicity, Definition 1 cannot be widely used in real-life applications. In Section 3, we show why this is the case and propose an efficient alternative. In addition, we extend this alternative to get a *pseudo-forgetting* method for the disjunctive rule of combination.

3 On the efficiency of decombination

In this section, we will study the problem of forgetting evidence from a computational point of view. We will focus on two masses m_1 and m_2, yet our results extend directly to the case where we have ℓ masses $m_1, \ldots, m_\ell$, their combination $m_{\oplus_{1,\ldots,\ell}}$ and want to remove the mass m_i from it, as we are considering associative combination rules. In this situation, m_i and $m_{\oplus_{1,\ldots,\ell}}$ correspond to m_2 and m in the next results. We therefore assume that two mass functions, m_1 and m_2, have been combined by one of the methods previously introduced, producing a mass function m.

3.1 Conjunctive combination

Given a state space $\mathcal{X}$ and mass functions m_1 and m_2, the problem of forgetting evidence after applying the unnormalized Dempster's rule of combination (Equation (4)) is formalized as follows:

Problem. FORGETTING
Input: *two mass functions m and m_2, both given as a list of their focal elements together with the corresponding mass values.*
Output: *the mass function $m_1 = m \ominus m_2$, as a list of focal elements together with the corresponding mass values.*

FORGETTING can be solved by applying the decombination operator (Equation (10)). However, this computation requires exponential time in the size of the state space, as shown in the following proposition.

Proposition 1. *Given two mass functions m and m_2, given as a list of their focal elements together with the corresponding mass values, computing $m_1 = m \ominus m_2$ by (1) computing the commonality functions q and q_2 corresponding to m and m_2, respectively, (2) using these to compute the commonality function q_1 corresponding to m_1 via Equation* (10)*, and (3) computing m_1 from q_1, takes time exponential in the size n of the frame of discernment (in the worst case).*

Proof. Let us consider n the number of possible states in $\mathcal{X}$. From Equation (10), we know that computing $(m \ominus m_2)(A)$ by the decombination operator implies computing the commonality function associated with m and m_2 for every subset of $\mathcal{X}$ that contains the set A. If $A = \emptyset$ this implies computing q_m and q_{m_2} for every subset of $\mathcal{X}$, that is, computing at least $2 \cdot 2^n$ operations. □

The next result is a key element of our next developments, and give some insights as to why it is useful to assume that masses are non-dogmatic.

Proposition 2. *Let $\mathcal{X}$ be a state space, m_1 an arbitrary mass function, m_2 a non-dogmatic mass function, and $m = m_1 \oplus_K m_2$ or $m = m_1 \oplus m_2$ a combined mass function. Then, the set $\mathcal{F}_{m_1}$ of focal elements of m_1 is contained in the set $\mathcal{F}_m$ of focal elements of m. That is, $\mathcal{F}_{m_1} \subseteq \mathcal{F}_m$.*

Proof. Let us assume that $A \in \mathcal{F}_{m_1}$, i.e, $m_1(A) > 0$. By Equation (2), $m(A) = {}^1\!/_K \sum_{B \cap C = A} m_1(B) \cdot m_2(C)$. Since m_2 is non-dogmatic, $m_2(\mathcal{X}) \neq 0$. This means that $m_1(A) \cdot m_2(\mathcal{X}) \neq 0$. Writing out the definition of $m(A)$, we get:

$$m(A) = {}^1\!/_K \cdot \left(m_1(A) \cdot m_2(\mathcal{X}) + \sum_{\substack{B \cap C = A \\ (B,C) \neq (A,\mathcal{X})}} m_1(B) \cdot m_2(C) \right) \neq 0.$$

This means that, $A \in \mathcal{F}_m$ for $m = m_1 \oplus_K m_2$. A similar argument proves the result for $m = m_1 \oplus m_2$—only the factor ${}^1\!/_K$ gets removed from the equation. □

We can use the above property to find an efficient solution for Forgetting.

Proposition 3. Forgetting *can be computed in polynomial time.*

Proof. Let $\mathcal{F}_m$, $\mathcal{F}_{m_2}$ and $\mathcal{F}_{m_1}$ be the sets of focal elements of m, m_2 and m_1 respectively, and ℓ the number of focal elements of m. We will define a linear system with ℓ equations and ℓ variables. For each $A \in \mathcal{F}_m$, we introduce a variable x_A and add the equation

$$m(A) = \sum_{\substack{B \cap C = A, \\ B \in \mathcal{F}_{m_2}, C \in \mathcal{F}_m}} m_2(B) \cdot x_C$$

This linear system is consistent since it has at least one solution. Consider the assignment that is obtained by setting $x_A = m_1(A)$ if $A \in \mathcal{F}_{m_1}$ and setting $x_A = 0$ otherwise. By Proposition 2 we know that $\mathcal{F}_{m_1} \subseteq \mathcal{F}_m$. This means that when writing out the value of $m(A)$ for each $A \in \mathcal{F}_m$ using Equation (4), coincides with substituting x_C by $m_1(C)$ for each $C \in \mathcal{F}_{m_1}$ in the system of linear equations. Therefore, we know that this solution satisfies all equations.

In addition, all equations of this linear system are independent. Let us assume $A_1, \ldots, A_t$ are different focal elements of m. Additionally, let us assume the equation defined for A_1, $\mathrm{Eq}(A_1)$, is a linear combination of the equations defined for the remaining focal elements $A_2, \ldots, A_t$, i.e., $\mathrm{Eq}(A_2), \ldots, \mathrm{Eq}(A_t)$ respectively. Therefore, every variable in equations $\mathrm{Eq}(A_2), \ldots, \mathrm{Eq}(A_t)$ is also a variable in $\mathrm{Eq}(A_1)$. By definition of $\mathrm{Eq}(A_1)$, if a variable x_C forms part of this equation then $A_1 \subseteq C$; so $A_1 \subseteq A_j$ for every $A_j \in \{A_2, \ldots, A_t\}$. Following a similar reasoning, we know the variable x_{A_1} is at least in one equation $\mathrm{Eq}(A_j)$ with $A_j \in \{A_2, \ldots, A_t\}$, so $A_j \subseteq A_1$. Consequently, there exists A_j such that $A_1 = A_j$, which leads through a contradiction since all focal elements $A_1, \ldots, A_t$ are supposed to be different.

This system of linear equations can be computed in polynomial time, because finding the pairs in (A, B) of focal elements of m and m_2 (respectively) whose intersection is a certain focal element of m requires computing $\ell \cdot s$ intersections, where ℓ and s are the number of focal elements of m and m_2, respectively. Moreover, computing a solution to a system of linear equations can be done in polynomial time, using linear programming algorithms—see, e.g., [12]. As the system has as many variables as equations, the solution is unique. □

Example 1. Let $\mathcal{X} = \{a, b, c\}$ be a state space and m_1 and m_2 be two mass functions given by $m_1(\{a, b\}) = 0.3$, $m_1(\{b, c\}) = 0.3$, $m_1(\mathcal{X}) = 0.4$, and $m_2(\{a, c\}) = 0.5$, $m_2(\{a\}) = 0.2$, $m_2(\mathcal{X}) = 0.3$.

Applying the unnormalized Dempster's rule of combination, their combination $m = m_1 \oplus m_2$ is defined by $m(\{a, b\}) = 0.09$, $m(\{b, c\}) = 0.09$, $m(\{a, c\}) = 0.2$, $m(\{a\}) = 0.29$, $m(\{c\}) = 0.15$, $m(\emptyset) = 0.06$, $m(\mathcal{X}) = 0.12$.

Now, let us assume that m and m_2 are known and we want to compute m_1. Since m has 7 focal elements, we need to solve the following 7×7 linear system:

$$\begin{aligned}
m(\{a,b\}) &= m_2(\mathcal{X}) \cdot x_{\{a,b\}} \\
m(\{b,c\}) &= m_2(\mathcal{X}) \cdot x_{\{b,c\}} \\
m(\{a,c\}) &= m_2(\{a,c\}) \cdot x_{\{a,c\}} + m_2(\{a,c\}) \cdot x_{\mathcal{X}} + m_2(\mathcal{X}) \cdot x_{\{a,c\}} \\
m(\{a\}) &= m_2(\{a,c\}) \cdot x_{\{a,b\}} + m_2(\{a,c\}) \cdot x_{\{a\}} + m_2(\{a\}) \cdot x_{\{a,b\}} \\
&\quad + m_2(\{a\}) \cdot x_{\{a,c\}} + m_2(\{a\}) \cdot x_{\{a\}} + m_2(\{a\}) \cdot x_{\mathcal{X}} \\
&\quad + m_2(\mathcal{X}) \cdot x_{\{a\}} \\
m(\{c\}) &= m_2(\{a,c\}) \cdot x_{\{b,c\}} + m_2(\{a,c\}) \cdot x_{\{c\}} + m_2(\mathcal{X}) \cdot x_{\{c\}} \\
m(\emptyset) &= m_2(\{a,c\}) \cdot x_{\emptyset} + m_2(\{a\}) \cdot x_{\{b,c\}} + m_2(\{a\}) \cdot x_{\emptyset} \\
&\quad + m_2(\{a\}) \cdot x_{\{c\}} + m_2(\mathcal{X}) \cdot x_{\emptyset} \\
m(\mathcal{X}) &= m_2(\mathcal{X}) \cdot x_{\mathcal{X}}
\end{aligned}$$

Note that $x_{\{b\}}$ do not appear, because there are not focal events B of m_2 and C of m whose intersection is $\{b\}$.

If one substitutes the example numbers in the above equation, the obtained linear system has a unique solution: $x_{\{a,b\}} = 0.3$, $x_{\{b,c\}} = 0.3$, $x_{\{a,c\}} = 0$, $x_{\{a\}} = 0$, $x_{\{c\}} = 0$, $x_{\emptyset} = 0$, and $x_{\mathcal{X}} = 0.4$, which corresponds to m_1.

We can find a similar solution for the normalized Dempster's rule, that is, given $m = m_1 \oplus_K m_2$ and m_2, we can find m_1 in polynomial time. To this end, we will explore the corresponding variation of Forgetting.

Problem. Normalized-Forgetting
Input: *two mass functions m and m_2 (where m_2 is non-dogmatic), both given as a list of their focal elements together with the corresponding mass values.*
Output: *the mass function $m_1 = m \ominus_K m_2$, as a list of focal elements together with the corresponding mass values.*

Similarly to the case for Forgetting, Normalized-Forgetting can be solved by means of computing commonality numbers—see Definition 1—but this requires exponential time. We show that a similar approach as used to establish Proposition 3 also works for Normalized-Forgetting.

Proposition 4. Normalized-Forgetting *can be computed in polynomial time.*

Proof. Let $\mathcal{F}_m$, $\mathcal{F}_{m_1}$ and $\mathcal{F}_{m_2}$ be the sets of focal elements of m, m_1 and m_2 respectively, and ℓ the number of focal elements of m. Let us define a linear system with $\ell + 1$ equations and $\ell + 1$ variables. For each $A \in \mathcal{F}_m$, we introduce a variable x_A and add the following equation:

$$m(A) \cdot x_K = \sum_{\substack{B \cap C = A, \\ B \in \mathcal{F}_{m_2}, C \in \mathcal{F}_m}} m_2(B) \cdot x_C$$

We also introduce a variable x_K and the following additional equation:

$$x_K = \sum_{\substack{A \cap B \neq \emptyset, \\ A \in \mathcal{F}_m, B \in \mathcal{F}_{m_2}}} m_2(B) \cdot x_A$$

This linear system is consistent since it has at least one solution. Namely, consider the following assignment. We let $x_K = K$, where K is the value of Equation (3) for m_1 and m_2. We set $x_A = m_1(A)$ if $A \in \mathcal{F}_{m_1}$ and $x_A = 0$ otherwise. By Proposition 2 we know that $\mathcal{F}_{m_1} \subseteq \mathcal{F}_m$. This means that when writing out the value of $m(A)$ for each $A \in \mathcal{F}_m$ using Equation (2), coincides with substituting x_C with $m_1(C)$ for each $C \in \mathcal{F}_{m_1}$ in the system of linear equations. Therefore, we know that this solution satisfies all equations.

An analogous argument to the one used in the proof of Proposition 3 shows that the system has a unique solution (that corresponds to $m_1 = m \ominus_K m_2$). Moreover, the system of linear equations can be constructed and solved in polynomial time. □

Example 2. Consider Example 1 with the same $\mathcal{X}$, m_1 and m_2 with normalization. Applying the normalized Dempster's rule of combination, their combination $m = m_1 \oplus_K m_2$ is defined by $m(\{a,b\}) = m(\{b,c\}) = {}^{9}\!/_{94}$, $m(\{a,c\}) = {}^{10}\!/_{47}$, $m(\{a\}) = {}^{29}\!/_{94}$, $m(\{c\}) = {}^{15}\!/_{94}$, $m(\mathcal{X}) = {}^{6}\!/_{47}$.

Now, to compute m_1 from m and m_2 we need to solve the following 7×7 linear system:

$$\begin{aligned}
{}^{9}\!/_{94} \cdot x_K &= 0.3x_{\{a,b\}} \\
{}^{9}\!/_{94} \cdot x_K &= 0.3x_{\{b,c\}} \\
{}^{10}\!/_{47} \cdot x_K &= 0.8x_{\{a,c\}} + 0.5x_{\mathcal{X}} \\
{}^{29}\!/_{94} \cdot x_K &= 0.7x_{\{a,b\}} + x_{\{a\}} + 0.2x_{\{a,c\}} + 0.2x_{\mathcal{X}} \\
{}^{15}\!/_{94} \cdot x_K &= 0.5x_{\{b,c\}} + 0.8x_{\{c\}} \\
{}^{6}\!/_{47} \cdot x_K &= 0.3x_{\mathcal{X}} \\
x_K &= x_{\{a,b\}} + 0.8x_{\{b,c\}} + x_{\{a,c\}} + x_{\{a\}} + 0.8x_{\{c\}} + x_{\mathcal{X}}
\end{aligned}$$

3.2 Disjunctive combination

Solving linear systems to decombine evidence can be also applied to the disjunctive rule of combination. However, we show that these linear systems may be dependent–i.e., there will be an infinity of solutions–and this method does not solve the problem in polynomial time in general, as it does for the conjunctive rules. This is bad news computationally-wise, yet forgetting for the disjunctive operator is less critical application-wise, as classical inversion operations concerns the handling of inconsistency or the use of Dempster's rule in valuation based systems.

Problem. DISJUNCTIVE-FORGETTING
Input: *two mass functions m and m_2, both given as a list of their focal elements together with the corresponding mass values.*
Output: *a mass function m_1 such that $m_1 \circledcirc m_2 = m$, as a list of focal elements together with the corresponding mass values.*

When considering the disjunctive rule of combination, assuming that the retracted mass function is non-dogmatic is not enough to ensure that $\mathcal{F}_{m_1} \subseteq \mathcal{F}_m$. For example, given a state space $\mathcal{X} = \{a, b, c, d\}$, $m_1(\{a, b\}) = 0.6$, $m_1(\mathcal{X}) = 0.4$, $m_2(\{c\}) = 0.8$, and $m_2(\mathcal{X}) = 0.2$, the focal elements of $m = m_1 \circledcirc m_2$ are $\{a, b, c\}$ and $\mathcal{X}$, so $\mathcal{F}_{m_1} \not\subseteq \mathcal{F}_m$.

However, given an input for DISJUNCTIVE-FORGETTING, we can define a linear system to find m_1 by following a similar strategy as in the proof of Proposition 3. Let us consider the following linear system:

$$m(C) = \sum_{\substack{A \cup B = C, \\ A \in \mathcal{F}_{m_2}, B \in 2^{\mathcal{X}}}} m_2(A) \cdot x_B, \text{ for every } C \in \mathcal{F}_m \tag{12}$$

This system is consistent since m_1 corresponds to one of its solutions. However, there may be more variables than equations, leading to an infinite solution set. This is the case in the previous example. Again, suppose that the state space is $\mathcal{X} = \{a, b, c, d\}$, that $m_1(\{a, b\}) = 0.6$, $m_1(\mathcal{X}) = 0.4$, $m_2(\{c\}) = 0.8$, and $m_2(\mathcal{X}) = 0.2$, and all other mass values are 0. Then $m(\{a, b, c\}) = 0.48$ and $m(\mathcal{X}) = 0.52$. Now, let us construct the corresponding linear system:

$$\begin{aligned} 0.48 &= 0.8x_{\{a,b\}} + 0.8x_{\{a,b,c\}}, \\ 0.52 &= 0.8x_{\{a,b,d\}} + 0.8x_{\mathcal{X}} + \textstyle\sum_{B \in 2^{\mathcal{X}}} 0.2x_B. \end{aligned}$$

This example also gives us the key to understand that this linear system cannot be built in polynomial time in general.

Proposition 5. *Given an input for* DISJUNCTIVE-FORGETTING*, the number of variables of the linear system defined in Equation* (12) *grows exponentially in the size of the state space.*

Proof. Take an arbitrary $n \in \mathbb{N}$, and let $\mathcal{X} = \{1, \ldots, n\}$. Then choose m_2 such that $\mathcal{X}$ is a focal element of m_2, i.e., $m_2(\mathcal{X}) > 0$. Then $m(\mathcal{X}) > 0$ as well. So we have an equation in the linear system defined in Equation (12) that corresponds to $\mathcal{X} \in \mathcal{F}_m$. This equation includes an additive factor $m_2(B) \cdot x_C$ for all $B \in \mathcal{F}_{m_2}$ and all $C \in 2^{\mathcal{X}}$ such that $B \cup C = \mathcal{X}$. In particular, for $B = \mathcal{X}$, this includes variables x_C for each $C \subseteq \mathcal{X}$, which are 2^n many. □

We can add constraints to the linear system to get some suitable mass function m_1, allowing us to solve DISJUNCTIVE-FORGETTING in exponential time.

Proposition 6. DISJUNCTIVE-FORGETTING *is solvable in exponential time.*

Proof. Given m_2 and m we construct the linear system that is defined in Equation (12). In addition to this, we add the following (in)equalities:

$$\begin{gathered} 1 = \textstyle\sum_{B \in 2^{\mathcal{X}}} x_B, \\ 0 \le x_B \le 1 \text{ for each } \; B \in 2^{\mathcal{X}}. \end{gathered}$$

Assuming that there exists some m_1 such that $m_1 \circledcirc m_2 = m$, the resulting system has at least one solution, i.e., the one corresponding to m_1. Moreover, using polynomial-time algorithms for linear programming, we can find a solution in time polynomial in the number of variables. As in the worst case there are an exponential number of them, we can find a suitable m_1 in exponential time. $\square$

Additionally, we can leverage (the polynomial-time computability of) optimization versions of linear programming to select among the set of all possible mass functions m_1 such that $m_1 \circledcirc m_2 = m$. For example, we could add the optimization criterion that maximizes the following value $opt = \sum_{B \subseteq \mathcal{X}} |B| \cdot x_B$. As indicated by results from the literature [5], any m_1 that gives an optimal value for *opt* will also be a maximal value in the partially ordered sets that are induced by various extensions of set inclusion to belief functions. This is therefore an example of a well-motivated additional selection criterion to add. In fact, the method shown in the proof of Proposition 6 also directly works for any additional selection criterion on m_1 that can be expressed as a linear function over the variables x_B.

Considering these results, we conclude that all the evidence rules of combination that have been discussed can be reversed by solving linear systems to some extent. However, using the intersection to define the focal elements of the combination provides these rules with better properties—proper forgetting (meaning that the result is uniquely defined) and polynomial-time computability—than using the union. This holds regardless of the presence of normalization factors. We should however note that, in the disjunctive case, considering subnormalized masses ($m_i(\emptyset) \neq 0$)–as in the decombination operator defined in [2] for this rule–would ensure that $\mathcal{F}_{m_1} \subseteq \mathcal{F}_m$, and similar results to those for the conjunctive cases could be developed. Yet it seems a much stronger assumption than $m_i(\mathcal{X}) \neq 0$, as it would mean that our pieces of information are inconsistent from the start.

4 Small illustrative use case

Consider that we have a user whose preferences we want to identify. We further assume that these preferences depend on two (commensurate) criteria $c_1, c_2 \in \mathbb{R}$ and that any object $\vec{c} = (c_1, c_2) \in \mathbb{R}^2$ can be evaluated by a two parameter function $\omega_1 c_1 + \omega_2 c_2$ with $\omega_i \geq 0$ for $i = 1, 2$ and $\omega_1 + \omega_2 = 1$. In other words, we assume the user preferences can be modelled by a weighted average.

A common issue in such problems is to identify or elicit the *true* parameters $\vec{\omega}^* = (\omega_1^*, \omega_2^*) \in [0, 1]^2$, typically through some interactions with the user that is asked to compare or evaluate different alternatives. However, one can expect the user to not to be fully certain of her assertions, and to potentially commit mistakes leading to globally inconsistent assessments. In such a case, the use of uncertainty frameworks such as possibility theory or belief functions has recently been advocated as a good way to model this uncertainty and to treat the potentially resulting inconsistency. We consider here that each user interaction results in a simple support belief function where a single focal element A_i is given a certainty value α_i, and the rest goes to the frame $\mathbb{R}^2$. As $\omega_1 + \omega_2 = 1$, any focal element can be summarised by the potential values of ω_1, and often in the form of closed intervals. We will adopt this simplifying assumption here.

We can already note that a straightforward application of Definition 1 is just impossible, due to the fact that the frame of discernment is continuous. A discretisation step would then be necessary at some point, but how to operationally define it would be difficult. In contrast, our approach can perfectly deal with continuous frames of discernment, as long as the number of focal elements remains finite (a much weaker and often satisfied assumption in practice).

Consider the following four mass functions, possibly given by a user:

$$m_1(A_1 = [0.4, 0.7]) = 0.8, m_1([0, 1]) = 0.2;\ m_2(A_2 = [0.2, 06]) = 0.6, m_2([0, 1]) = 0.4$$

$$m_3(A_3 = [0.65, 1]) = 0.4, m_3([0, 1]) = 0.6;\ m_4(A_4 = [0, 0.3]) = 0.7, m_4([0, 1]) = 0.3$$

The final mass resulting from Dempster combination is

$$m(A_1) = 0.0576, m(A_2) = 0.0216, m(A_3) = 0.0056, m(A_4) = 0.0336,$$

$$m(A_5 = [0.4, 0.6]) = 0.0864, m(A_6 = [0.65, 0.7]) = 0.0384,$$
$$m(A_7 = [0.2, 0.3]) = 0.0504, m([0, 1]) = 0.0144$$

and $m(\emptyset) = 0.692$, indicating a high degree of conflict in the result of the merging. A way to restore some consistency is then to *forget* one of the sources of information, in which case it is natural to look for the source allowing for the highest removal, i.e., the source i^* such that

$$i^* = \arg \min_{i \in \{1,\dots,4\}} m_{-i}(\emptyset)$$

In order to do that, we just need to find the values of $x_\emptyset$ for four 7×7 linear systems, a straightforward operation. We get $i^* = 4$, and a diminshed conflict of $m_{-4}(\emptyset) = 0.24$.

5 Conclusion

In this paper, we studied the complexity of decombining belief functions, both for the conjunctive (unnormalized and normalized versions) and disjunctive rules. We concluded that while the problem is well-posed and can be efficiently solved for the conjunctive one, it is more problematic for the disjunctive one.

Next steps include applying our results to various applications (sensor measurements, preference elicitation [9]) and studying the same problem for other combination rules that are associative and commutative, such as the cautious and bold rules [2].

Acknowledgments

Daira Pinto Prieto has benefited from a visit grant of the SAFE AI chair (funded by the Fondation UTC pour l'Innovation).

References

1. A. P. Dempster. Upper and lower probabilities induced by a multivalued mapping. In *Classic works of the Dempster-Shafer theory of belief functions*, pages 57–72. Springer, 2008.
2. T. Denœux. Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence. *Artificial Intelligence*, 172(2-3):234–264, 2008.
3. T. Denœux. Logistic regression, neural networks and Dempster–Shafer theory: A new perspective. *Knowledge-Based Systems*, 176:54–67, 2019.
4. S. Destercke and T. Burger. Toward an axiomatic definition of conflict between belief functions. *IEEE transactions on cybernetics*, 43(2):585–596, 2013.
5. S. Destercke and D. Dubois. Idempotent conjunctive combination of belief functions: Extending the minimum rule of possibility theory. *Information Sciences*, 181(18):3925–3945, 2011.
6. D. Dubois, F. Faux, and H. Prade. Prejudiced information fusion using belief functions. In *Belief Functions: Theory and Applications: 5th International Conference, BELIEF 2018, Compiègne, France, September 17-21, 2018, Proceedings 5*, pages 77–85. Springer, 2018.
7. P. H. Giang. Decision with Dempster–Shafer belief functions: Decision under ignorance and sequential consistency. *International Journal of Approximate Reasoning*, 53(1):38–53, 2012.
8. M. L. Ginsberg. Non-monotonic reasoning using Dempster's rule. In *AAAI*, volume 84, pages 112–119. Citeseer, 1984.
9. P.-L. Guillot and S. Destercke. Preference elicitation with uncertainty: Extending regret based methods with belief functions. In *13th International Conference on Scalable Uncertainty Management (SUM 2019)*, volume 11940, pages 289–309, Compiègne, France, Dec. 2019.
10. I. Kramosil. Measure-theoretic approach to the inversion problem for belief functions. *Fuzzy Sets and Systems*, 102(3):363–369, 1999.
11. S. Moral and N. Wilson. Markov chain Monte–Carlo algorithms for the calculation of Dempster–Shafer belief. In *Proceedings of the Twelfth AAAI National Conference on Artificial Intelligence*, pages 269–274, 1994.
12. A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.
13. G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, NJ, 1976.
14. P. P. Shenoy. Conditional independence in valuation-based systems. *International Journal of Approximate Reasoning*, 10(3):203–234, 1994.
15. P. P. Shenoy. Binary join trees for computing marginals in the Shenoy–Shafer architecture. *International Journal of approximate reasoning*, 17(2-3):239–263, 1997.
16. P. Smets. Belief functions: The disjunctive rule of combination and the generalized Bayesian theorem. *International Journal of Approximate Reasoning*, 9(1):1–35, 1993.
17. P. Smets. The canonical decomposition of a weighted belief. In *International Joint Conference on Artificial Intelligence*, 1995.
18. P. Smets and R. Kennes. The transferable belief model. *Artificial Intelligence*, 66(2):191–234, 1994.
19. F. Xiaojing, H. Deqiang, Y. Yi, and J. Dezert. De-combination of belief function based on optimization. *Chinese Journal of Aeronautics*, 35(5):179–193, 2022.