

Computationally Efficient Multiple Hypothesis Association of Intelligence Reports

Johan Schubert

Department of Data and Information Fusion
Division of Command and Control Systems
Swedish Defence Research Agency
SE-164 90 Stockholm, Sweden
schubert@foi.se
<http://www.foi.se/fusion/>

John Cantwell¹

Division of Philosophy
Royal Institute of Technology
SE-100 44 Stockholm, Sweden
cantwell@kth.se
<http://www.infra.kth.se/~cantwell/>

Abstract - In this paper we develop a computationally efficient multiple hypothesis association algorithm for generation of alternative association hypotheses regarding cluster memberships of intelligence reports represented as belief functions. We have previously an $O(N^2K^2)$ clustering algorithm using a measure of pairwise conflicts, and a fast algorithm for classification of clusters using a more advanced measure. As these measures are similar but not identical and may have different minima we generate additional multiple association hypotheses around the solution found by the clustering algorithm. These hypotheses may then be evaluated by the classification algorithm in order to find the best overall classification of all clusters. In order to maintain the computational complexity we will investigate algorithms that run in no worse than $O(N^2K^2)$ time.

Keywords: Multiple hypotheses association, clustering, Dempster-Shafer theory, intelligence processing, belief function, force aggregation.

1 Introduction

In this paper we develop a computationally efficient multiple hypothesis association algorithm that may serve as an interface between a clustering algorithm and a classification algorithm in order to evaluate multiple different associations. When we cluster large amounts of intelligence data with any efficient clustering algorithm we usually find a good local minimum. This is the case in Dempster-Shafer clustering (DSC) [1–4], when intelligence reports represented as belief functions [5–7] are clustered based on their pairwise conflict [8] into subsets corresponding to different events that should be handled independently. In order to have a

computationally efficient algorithm we only use pairwise information.

If the following classification of the intelligence reports in each cluster is made with a more advanced measure, not only taking pairwise information into account, the minimum of this measure is not guaranteed to exactly coincide with the overall minimum of the pairwise conflicts used in the clustering process. This is the reason we are interested in finding a larger set of alternative multiple hypothesis of the association of all intelligence reports. With a larger set of alternative association hypotheses the optimization of the classification measure is improved. However, it is important that the computational complexity of the generation and evaluation of all association hypotheses are no worse than the computational complexity of the clustering algorithm used, so as not to reduce the efficiency of the overall clustering and classification method.

The computational complexity of DSC using Potts spin [9, 10] mean field theory [11] is $O(N^2K^2)$ [12–15], where N is the number of intelligence reports and K the number of clusters, both assumed fixed for the problem. Problems where N is not fixed was studied in [16], and problems where K is not fixed was studied in [17], and problems where neither N nor K was fixed was studied in [18].

We will in this paper investigate different schemes to generate additional association hypothesis that may be evaluated in $O(N^2K^2)$ time and present an algorithm that achieves the computational requirement while maintaining a good balance between the search *depth* (i.e., iterative changes) and the search *width* of maintained alternative association hypotheses.

The proposed multiple hypothesis association algorithm, while general in scope, was developed as an interface between two modules for clustering intelligence reports and classification of clustered

1. This work was done while the author was with the Swedish Defence Research Agency.

subsets in a force aggregation [19–21] method for an information fusion demonstrator for tactical intelligence processing [22, 23].

In Section 2 we discuss the idea of multiple hypothesis association and possible algorithms. An algorithm is then presented in Section 3, and its computational complexity is discussed. In Section 4 we give for the sake of completeness a short overview of the clustering and classification parts of force aggregation describing how the multiple hypothesis association algorithm can fit in. Finally (in Section 5), conclusions are drawn.

2 Multiple hypothesis association

2.1 Generation of multiple hypotheses

Let us start with the initial association hypothesis given as output from the clustering algorithm. At this level 0 we have one initial hypothesis with k clusters, Figure 1.

Level 0:

Hypothesis 1:

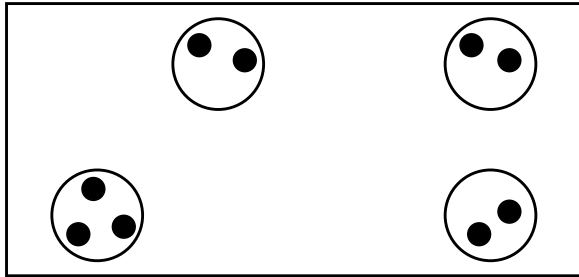


Figure 1: Example with one initial hypothesis, four clusters and nine reports (level 0).

At the next level we generate all possible hypotheses with one association change. Since we have nine reports and each report can be moved to any of the other three clusters we have 27 new hypotheses, or $n(k-1)$ new hypotheses in the general case, where n is the number of reports and k the number of clusters in the initial hypothesis. In Figure 2 we observe the first two of these 27 new hypotheses.

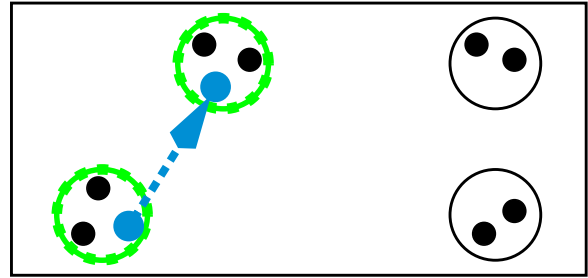
Note that the first hypothesis generates two new clusters that must be evaluated, while the second hypothesis (and all the other hypotheses generated by moving the same report) generates only one additional cluster. In total we have in this level 36 new clusters, or nk new clusters in the general case.

Let us now study an interesting phenomenon on levels two and deeper. As an example we continue with hypothesis one from level one, in Figure 3.

The report that was moved at the previous level must now be excluded. Moving this report to any other cluster would give a hypothesis at level 3 already generated at level 2 (or level 1, if it is moved back to where it came from). For instance, if we moved it to the blue cluster (top right in Figure 3) we would obtain hypothesis number 2 at level 1, already generated.

Level 1:

Hypothesis 1:



Hypothesis 2:

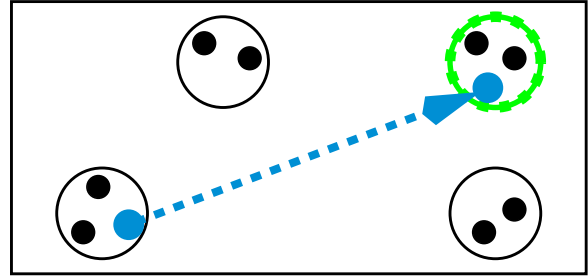


Figure 2: At level one we have $n(k-1)$ new hypotheses and nk new clusters (green).

Level 2:

Hypothesis 1:

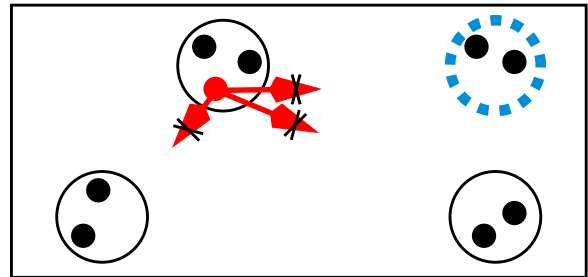


Figure 3: At level 2, the report just moved (red; in the top left corner) is excluded from further moving as that would only give a hypothesis already generated.

With one less report, we generate 24 new hypotheses, i.e., 8×3 , on level 3 starting from hypothesis 1 on level 2, and not the 27 (9×3) we were able to generate on level 2.

In the general case, for any level p , we generate $(n-p+1)(k-1)$ new hypotheses starting from each hypothesis on the previous level.

If this process is represented by a tree where each node is an hypothesis and the first hypothesis at level 0 is the root, we will have a branching factor of $n(k-1)$ towards level 1, and a branching factor of $(n-1)(k-1)$ towards level 2, and in general a branching factor of $(n-p+1)(k-1)$ towards level p .

However, the branching factor and the number of generated hypotheses and clusters on each level overstates the problem size as the same hypothesis may be reached through different branches. As an example, if report 1 is moved from A to B on one level and report 2 from C to D on the next level, this gives the same hypothesis as if report 2 was moved

from C to D on the first level and report 1 from A to B on the next level.

Taking this into account we may calculate the total number of new hypotheses and clusters for all branches at each level. For instance, for level 2 we start out with the $n(k-1)$ hypotheses at level 1 and with a branching factor of $(n-1)(k-1)$, eliminating for any duplicate generation of hypothesis in different branches, we receive a total of

$$\frac{n(n-1)}{2}(k-1)^2 \quad (1)$$

new hypotheses at level 2. The total number of hypotheses for level 0 to 2 becomes

$$\frac{n(n-1)}{2}(k-1)^2 + n(k-1) + 1, \quad (2)$$

etc. If this is continued until a depth of $p < n$ we will have a number of hypotheses equal to

$$O\left(\binom{n}{p}k^p\right). \quad (3)$$

Similarly, the number of new clusters may be calculated level by level. At level 1 we have nk new clusters distributed over the hypotheses. At level 2 we receive $n(n-1)(k-1)^2$ new clusters and if we continue to a depth of p we have

$$O\left(\frac{n!}{p!}k^p\right) \quad (4)$$

number of clusters, Table 1.

Table 1: Total number of hypotheses and evaluated clusters for each maximum depth.

Depth	# Clusters	# Hypotheses
0	k	1
1	$nk + k$	$n(k-1) + 1$
2	$n(n-1)(k-1)^2 + nk + k$	$\frac{n(n-1)}{2}(k-1)^2 + n(k-1) + 1$
...
p	$O\left(\frac{n!}{p!}k^p\right)$	$O\left(\binom{n}{p}k^p\right)$

When p is a constant and $p \ll n$ a full search to level p has a computational complexity of $O(n!k^p)$ which is much worse than that of DSC. If we intend to develop an $O(n^2k^2)$ algorithm (the computational complexity of DSC) for generation and evaluation of multiple hypothesis association, this is only acceptable if

$p \leq 2$. However, such a wide but shallow search space for finding a good solutions is hardly what we are looking for. By reducing the width of the search we may increase the depth to something more applicable.

2.2 Depth vs. width in hypotheses

Both the width and the depth of the search should be some function of N , the number of intelligence reports, and K , the number of clusters.

Let us investigate seven different algorithms with different *widths* and *depths*, in Table 2. All hypotheses at a particular level are evaluated. For instance, $L1_wNK$ indicate the number of hypotheses evaluated at level 1, etc. The selection is then made of the top ranking hypotheses based on the evaluation. $L0_w$, $L1_w$ and $L2_w$ indicate the number of hypotheses selected for that level (i.e., the *width*). This process is repeated level-by-level until the specified *depth*.

Table 2: Algorithms A1-A7, their widths, depths, and their computational complexity (last row). Example with $N = 1000$ and $K = 10$.

	A1	A2	A3	A4	A5	A6	A7
<i>width</i>	1	1	K	N/K	N	N/K	\sqrt{NK}
<i>depth</i>	N	K	K	K	K	N/K	\sqrt{NK}
$L0_w$	1	1	1	1	1	1	1
$L1_wNK$	10,000	10,000	10,000	10,000	10,000	10,000	10,000
$L1_w$	1	1	10	100	1,000	100	100
$L2_wNK$	10,000	10,000	100,000	1,000,000	10,000,000	1,000,000	1,000,000
$L2_w$	1	1	10	100	1,000	100	100
$L3_wNK$	10,000	10,000	100,000	1,000,000	10,000,000	1,000,000	1,000,000
...							
$O(\cdot)$	$O(N^2K)$	$O(NK^2)$	$O(NK^3)$	$O(N^2K)$	$O(N^2K^2)$	$O(N^3K^{-1})$	$O(N^2K^2)$

We may observe in Table 2 that algorithms A1-A2 are very fast, but with a width equal to one. A6 is much too slow, while A3-A5 and A7 are computationally reasonable. Here A3 has some width but is shallow. A4-A5 have good width, but are both shallow. Only A7 is fast with a good width and depth. To summarize:

- A1: Fast, no expansion.
- A2: Faster, no expansion, but shallow.
- A3: Fast, some expansion, but shallow.
- A4: Fast, with expansion, but shallow.
- A5: Good speed, expansion, but shallow.
- A6: Slow, and deep.
- A7: Good speed, expansion, and depth.

The computational complexity in the last row of Table 2 is calculated as $depth \times width \times NK$. For most applications algorithm A7 seems appropriate.

2.3 Evaluation of multiple hypotheses

At each level all hypotheses are evaluated using some problem specific measure of fitness. We assume that the evaluation of all hypotheses (except for the first evaluation at level $L0$) can be performed incrementally in constant time.

For instance, the evaluation of the fitness $\Gamma(u^*)$ of a new hypothesis u^* need only concern the two clusters involved in the incremental change if we use the previous fitness, $\Gamma(u)$. This makes the computational complexity independent of K .

Similarly, the evaluation of the fitness μ of the two clusters is preceded by an incrementally performed combination and decombination between the belief function moved and the belief functions corresponding to the intelligence of the two clusters, followed by the actual calculation of fitness, all usually assumed independent of N . Thus, allowing the evaluation to run in constant time, i.e., independent of N and K .

3 A multiple hypothesis association algorithm

Let us present an algorithm (A7) for generation and evaluation of multiple hypothesis association of belief functions, Figure 4. We assume that all belief functions have been clustered and we receive the initial association hypothesis $\{\chi_a\}$ as the first hypothesis u and input to the A7-algorithm.

In order to achieve a good balance between the *depth* and the *width* of the search with an $O(N^2K^2)$ algorithm we choose $depth = width = \lfloor \sqrt{NK} \rfloor$. All clusters are classified and the fitness for each cluster μ and for the entire hypothesis Γ is calculated.

For each successive level $[u^*, \Gamma(u^*)]$ is stored in a hash list and u^* is looked up before $\Gamma(u^*)$ is calculated in order to avoid duplicate evaluations.

The algorithm is presented in Figure 4. Here, n is a belief function, k is a cluster (of belief functions) and u is a hypothesis (of clusters). Let $k(n)$ be the cluster that n belongs to, $u(k)$ the k^{th} cluster of u . We have $u(k(n))$, the cluster of u that n belongs to.

3.1 Computational complexity

We maintain the $\lfloor \sqrt{NK} \rfloor$ best hypotheses (using the *width*) after evaluation at each level. Each of these hypotheses generates NK new hypotheses on the next level for a total of $NK \lfloor \sqrt{NK} \rfloor$ evaluations at each level.

Performing the analysis to a search depth of $\lfloor \sqrt{NK} \rfloor$ gives a computational complexity for the A7-algorithm of $O(N^2K^2)$.

4 Multi hypothesis force aggregation

We have defined evidential force aggregation as the combination of two processes: DSC using neural

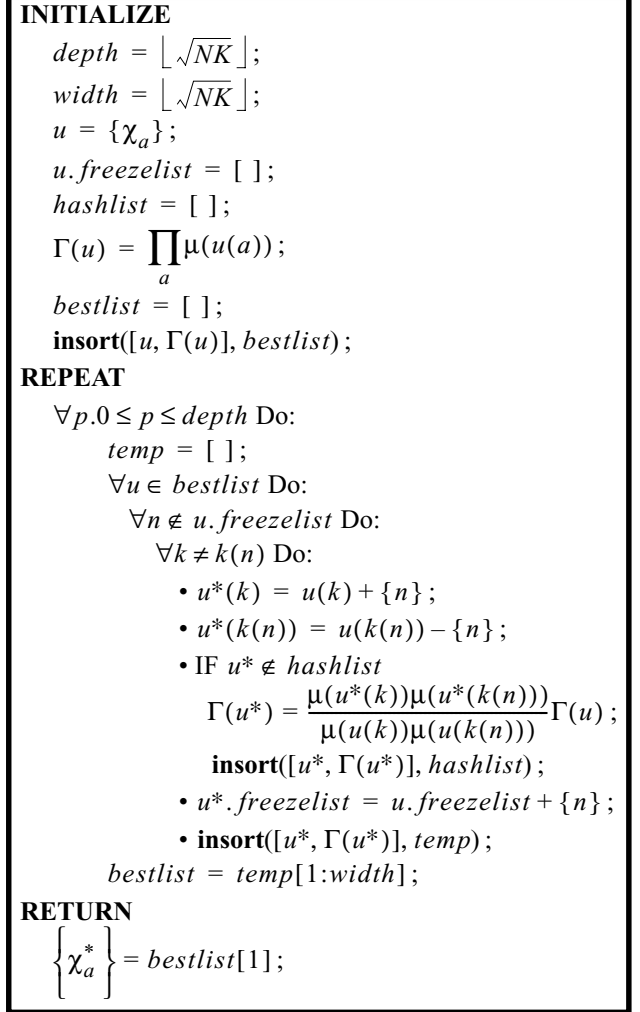


Figure 4: A multiple hypothesis association algorithm (Algorithm-A7).

clustering [14] followed by evidential classification and matching (ECM) [20] of clustered subsets.

We now add the intermediate component of multiple hypothesis association (MHA). Combining the developed MHA in this paper with two previous results, DSC and ECM, we achieve a multiple hypothesis force aggregation, Figure 5.



Figure 5: Multiple hypothesis force aggregation.

We include the algorithms of DSC, Figure 6, and ECM, Figure 7, here for the sake of completeness of this paper. For a full description see [14, 20].

From DSC we receive a complete partitioning χ_a of all intelligence into subsets. This will serve as the initial association hypothesis in MHA. MHA will generate a large but manageable number of alternative associations around the initial solution with a computational complexity not worse than DSC. These alternative associations u are evaluated using ECM.

We have

$$\Gamma(u) = \prod_a \mu(u(a)), \quad (5)$$

INITIALIZE
 K (# clusters); N (# intelligence reports);
 $J_{ij}^- = -\log(1 - c_{ij})\delta_{|A_i \cap A_j|} \quad \forall i, j$;
 $J_{ij}^+ = -\log(1 - p_{ij})(1 - \delta_{|A_i \cap A_j|}) \quad \forall i, j$;
 $s = 0$; $t = 0$; $\varepsilon = 0.001$; $\tau = 0.9$; $\gamma = 0.5$;
 $T^0 = T_c$ (a critical temperature)
 $= \frac{1}{K} \cdot \max(-\lambda_{min}, \lambda_{max})$, where λ_{min} and λ_{max} are
the extreme eigenvalues of M , where
 $M_{ij} = J_{ij}^- - J_{ij}^+ + \alpha - \gamma\delta_{ij}$;
 $V_{ia}^0 = \frac{1}{K} + \varepsilon \cdot \text{rand}[0,1] \quad \forall i, a$;

REPEAT
REPEAT-2
 $\forall i$ Do: $\sum_{j=1}^N (J_{ij}^- - J_{ij}^+ + \alpha)V_{ja}^s \begin{cases} s+1, j < i \\ s, j \geq i \end{cases} - \gamma V_{ia}^s \forall a$;
• $H_{ia}^s = \sum_{j=1}^K e^{-H_{ia}^s/T^t}$;
• $F_i^s = \sum_{a=1}^K e^{-H_{ia}^s/T^t} + \varepsilon \cdot \text{rand}[0,1] \quad \forall a$;
• $s = s + 1$;

UNTIL-2
 $\frac{1}{N} \sum_{i,a} |V_{ia}^s - V_{ia}^{s-1}| \leq 0.01$;
 $T^{t+1} = \tau \cdot T^t$;
 $t = t + 1$;

UNTIL
 $\frac{1}{N} \sum_{i,a} (V_{ia}^s)^2 \geq 0.99$;

RETURN
 $\left\{ \chi_a \mid \forall S_i \in \chi_a, \forall b \neq a, V_{ia}^s > V_{ib}^s \right\}$;

Figure 6: Clustering algorithm.

Figure 4, where

$$\mu(u(a)) = \pi_{\oplus J_a}(T_i) \mid \forall j \neq i, \pi_{\oplus J_a}(T_i) > \pi_{\oplus J_a}(T_j) \quad (6)$$

and $\{T_i\}$ is a set of templates and $\pi_{\oplus J}(T_i)$ is the fitness between template T_i and the combined belief functions of χ_a , Figure 7.

This should improve the quality of the classification algorithm and the overall force aggregation performance.

5 Conclusions

We have shown that it is possible to start out from an already performed partitioning of intelligence into subsets, and from there generate and classify additional multiple hypothesis associations around the initial partitioning. This can be achieved with a computational complexity no worse than that of DSC.

INITIALIZE
• J_a (the intelligence in χ_a^*);

COMBINE
• $\oplus J_a$;

CALCULATE
• $m_{\oplus J_a}(\langle x_1, x_2, \dots, x_{|I_a|} \rangle)$;
• $\pi_{\oplus J_a}(T_i) \quad \forall i$;

RETURN
• $\pi_{\oplus J_a}(T_i) \mid \forall j \neq i, \pi_{\oplus J_a}(T_i) > \pi_{\oplus J_a}(T_j)$;

Figure 7: An evidential force aggregation algorithm.

References

- [1] J. Schubert, On nonspecific evidence, *International Journal of Intelligent System*, 8(6):711–725, 1993.
- [2] J. Schubert, Specifying nonspecific evidence, *International Journal of Intelligent Systems*, 11(8):525–563, 1996.
- [3] J. Schubert, Managing inconsistent intelligence, in *Proceedings of the Third International Conference on Information Fusion (FUSION 2000)*, Paris, France, 10–13 July 2000, International Society of Information Fusion, Sunnyvale, 2000, pages TuB4/10–16.
- [4] T. Denoeux and M. Masson, Clustering of proximity data using belief functions, in Bernadette Bouchon-Meunier, Laurent Foulloy, and Ronald R. Yager, editors, *Intelligent Systems for Information Processing: From Representation to Applications*, pages 291–302, Elsevier Science, Amsterdam, 2003.
- [5] A.P. Dempster, A generalization of Bayesian inference, *Journal of the Royal Statistical Society (Series B)*, 30(2):205–247, 1968.
- [6] G. Shafer, *A mathematical theory of evidence*, Princeton University Press, Princeton, NJ, 1976.
- [7] P. Smets and R. Kennes, The transferable belief model, *Artificial Intelligence*, 66(2):191–234, 1994.
- [8] P. Smets (Mar., 2005), Analyzing the Combination of Conflicting Belief Functions, Submitted. [Online]. Available: http://iridia.ulb.ac.be/~psmets/Combi_Confl.pdf
- [9] R.B. Potts, Some generalized order-disorder transformations, *Proceedings of the Cambridge Philosophical Society*, 48:106–109, 1952.
- [10] F.Y. Wu, The Potts model, *Reviews of Modern Physics*, 54(1):235–268, 1982.
- [11] C. Peterson and B. Söderberg, A new method for mapping optimization problems onto neural networks, *International Journal of Neural Systems*, 1(1):3–22, 1989.

- [12] M. Bengtsson and J. Schubert, Dempster-Shafer clustering using Potts spin mean field theory, *Soft Computing*, 5(3):215–228, 2001.
- [13] J. Schubert, Clustering belief functions based on attracting and conflicting metalevel evidence, in Bernadette Bouchon-Meunier, Laurent Foulloy, and Ronald R. Yager, editors, *Intelligent Systems for Information Processing: From Representation to Applications*, pages 349–360, Elsevier Science, Amsterdam, 2003.
- [14] J. Schubert, Clustering belief functions based on attracting and conflicting metalevel evidence using Potts spin mean field theory, *Information Fusion*, 5(4):309–318, 2004.
- [15] J. Schubert, Managing decomposed belief functions, in *Proceedings of the Eleventh International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2006)*, Paris, France, 2–7 July 2006, to appear.
- [16] J. Schubert, Reliable force aggregation using a refined evidence specification from Dempster-Shafer clustering, in *Proceedings of the Fourth International Conference on Information Fusion (FUSION 2001)*, Montréal, Canada, 7–10 August 2001, pages TuB3/15–22, International Society of Information Fusion, Sunnyvale, CA, 2001.
- [17] J. Schubert, Simultaneous Dempster-Shafer clustering and gradual determination of number of clusters using a neural network structure, in *Proceedings of the 1999 Information, Decision and Control Conference (IDC'99)*, pages 401–406, Adelaide, Australia, 8–10 February 1999, IEEE, Piscataway, NJ, 1999.
- [18] J. Schubert and H. Sidenbladh, Sequential clustering with particle filters - Estimating the number of clusters from data, in *Proceedings of the Eighth International Conference on Information Fusion (FUSION 2005)*, Philadelphia, USA, 25–29 July 2005, paper A4-3, pages 1–8, IEEE, Piscataway, NJ, 2005.
- [19] J. Cantwell, J. Schubert and J. Walter, Conflict-based force aggregation, in *Proceedings of the Sixth International Command and Control Research and Technology Symposium*, Annapolis, USA, 19–21 June 2001, track 7, paper 031, pages 1–15, US Department of Defence CCRP, Washington, DC, 2001.
- [20] J. Schubert, Evidential force aggregation, in *Proceedings of the Sixth International Conference on Information Fusion (FUSION 2003)*, Cairns, Australia, 8–11 July 2003, pages 1223–1229, International Society of Information Fusion, Sunnyvale, CA, 2003.
- [21] P. Svenson, Capabilities-based force aggregation using random sets, in *Proceedings of the Eighth International Conference on Information Fusion (FUSION 2005)*, Philadelphia, USA, 25–29 July 2005, paper C7-4, pages 1–7, IEEE, Piscataway, NJ, 2005.
- [22] J. Schubert, C. Mårtenson, H. Sidenbladh, P. Svenson and J. Walter, Methods and system design of the IFD03 information fusion demonstrator, in *Proceedings of the Ninth International Command and Control Research and Technology Symposium*, track 7.2, paper 061, pages 1–29, Copenhagen, Denmark, 14–16 September 2004, US Department of Defence CCRP, Washington, DC, 2004.
- [23] S. Ahlberg, P. Hörling, H. Kjellström, K. Johansson, K. Jöred, C. Mårtenson, G. Neider, J. Schubert, P. Svenson, P. Svensson and J. Walter (Jan., 2006), An information fusion demonstrator for tactical intelligence processing in network-based defense, *Information Fusion*, to appear. [Online]. Available: <http://dx.doi.org/10.1016/j.inffus.2005.11.002>