Scalable Algorithms for Aggregating Disparate Forecasts of Probability

J. B. Predd S. R. Kulkarni H. V. Poor
Department of Electrical Engineering
Princeton University
Princeton, NJ 08544
{jpredd,kulkarni,poor}@princeton.edu

D. N. Osherson
Department of Psychology
Princeton University
Princeton, NJ 08544
osherson@princeton.edu

Abstract - ***In this paper, computational aspects of the* panel aggregation problem *are addressed. Motivated primarily by applications of risk assessment, an algorithm is developed for aggregating large corpora of internally incoherent probability assessments. The algorithm is characterized by a provable performance guarantee, and is demonstrated to be orders of magnitude faster than existing tools when tested on several real-world data-sets. In addition, unexpected connections between research in risk assessment and wireless sensor networks are exposed, as several key ideas are illustrated to be useful in both fields.***

Keywords: aggregation, forecasting, fusion, risk assessment, sensor networks

1 Introduction

1.1 Aggregating Human Expertise

In this paper, we address the problem of aggregating disparate forecasts of probability, motivated primarily by applications of risk assessment and analysis [20]. In these settings, a dearth of hard data often limits one's ability to extrapolate the future from the past. As a result, panels of human experts are frequently consulted to make forecasts about future events and to characterize the uncertainty therein. For example, stock market analysts are employed to design risk-balanced investment portfolios, and geopolitical forecasters help construct robust policies [16] and risk-based resource allocation schemes [18, 26]. Typically, a multiplicity of experts are consulted in order to maximize the information available to the would-be decision-maker. However, to be useful for decision-making and analysis, the panel's generally disparate opinion must be fused into a single, coherent body of forecasts.

This *panel aggregation problem* has been usefully addressed in many fields, including philosophy, law, statistics, risk assessment and computer science. Recently, Osherson and Vardi [21] revisited the problem, to address the case where a panel of human judges provides forecasts of probability for both logically simple and complex events. A *coherent approximation principle* (CAP) was proposed as a generalization of linear averaging methods (see, e.g., [8], [9]). As discussed below, CAP is practically motivated, accommodating both incoherent (e.g., human) judges and partially specified forecasts. However, as noted in [21], implementing CAP is NP-Hard in the general case. Thus, for problems of interest, the CAP approach to aggregation is computationally infeasible in theory and practice.

Also in [21], Osherson and Vardi propose a method for addressing CAP's computational challenge. Termed SAPA (Simulated Annealing over Probability Arrays), their algorithm applies to a very broad class of logically complex forecasts. Though vastly better than off-the-shelf tools, SAPA nonetheless requires many hours to aggregate reasonably sized panels; CAP remains of limited use in practice.

Nevertheless, in several experiments documented in [21], it was noted that on real-world data sets, *aggregating expertise using CAP (via SAPA) improves the forecasting accuracy of a panel* according to several naturally quantified measures for stochastic accuracy (we elaborate on this finding below). This empirical result invites us to develop computationally efficient tools for implementing (or approximately implementing) CAP, so that these findings may be exploited in practice.

Thus, the primary motivation for this paper is CAP's computational challenge. Here, we derive a scalable algorithm for aggregating human-provided forecasts of probability using CAP. By exploiting the logical simplicity of the events in question, a convenient application of alternating projection algorithms provides a fast tool for risk assessment with a provable performance guarantee and documented empirical success.

1.2 Wireless Sensor Networks

A recurrent theme in the study of wireless sensor networks (WSNs) [1] is the need to exploit node-level intelligence when designing communication-efficient systems for distributed detection and estimation. With sensors that communicate inferences (rather than raw data), future WSNs will trade computational power for energy and bandwidth. This vision is a driver behind the demand for col-

*This research was supported in part by the Army Research Office under Grant DAAD19-00-1-0466, in part by the U. S. Army Pantheon Project, in part by Draper Laboratory under Grant IR&D 6002, and in part by the National Science Foundation under Grants CCR-0020524, CCR-0312413 and IIS-9978135.

laborative signal processing [15] and for fusion strategies for aggregating inferences made by smart sensors. As alluded to above, researchers in risk assessment have long been interested in extracting robust and calibrated forecasts from *human* experts through collaboration and aggregation, and have developed a host of tools for doing so. Given this thematic connection between WSNs and risk assessment, tools and insights may be shared between these seemingly disconnected fields. Thus, a secondary motivation for this paper is to connect studies in risk assessment with research in sensor networks (and *vice-versa*) and to expose a set of fundamental tools that may be useful for both.

1.3 Organization

The remainder of this paper is organized as follows. In Section II, we introduce notation and review alternating projection algorithms, a tool that we exploit in deriving our scalable aggregation algorithm. In Section III, we formalize the panel aggregation problem, review Osherson and Vardi's coherent approximation principle, and discuss its relation to other approaches to aggregation. In Section IV, we derive an iterative algorithm which approximately implements CAP and we prove a theorem which characterizes the algorithm's dynamics. In Section V, we validate our approach with experiments on several real-world data sets. Finally, in Section VI, we discuss extensions of the current work and connections to collaborative signal processing in WSNs.

2 Preliminaries

2.1 Notation

Let $X = (X_1, \ldots, X_n)$ be a vector of Boolean[1] variables. Each component of X models a *basic event*. For example, the event that "Google stock outperforms the NASDAQ in the third quarter" may be described by a Boolean variable X_1 whose value is 1 if the event is true and 0 otherwise. X therefore models a set of n basic events, which could describe the performance of a set of stocks, the status of various economic indicators, the outcome of geopolitical events, etc.

Complex events are modeled by joining the components of X with logical connectives like $\{\neg, \wedge, \vee, \ldots\}$. For example, the complex event that "Google stock outperforms the NASDAQ *AND* the U.S. GDP increases in the third quarter" may be modeled by the conjunction $X_1 \wedge X_2$, with X_2 appropriately chosen. In a slight abuse of notation, we henceforth refer to components of X and logical combinations thereof by basic events and complex events, respectively.

A *forecast* $(E, \hat{p})$ is an event E (basic or complex) paired with a real-number $\hat{p} \in [0, 1]$. $\hat{p}$ is interpreted as an assessment of the probability that the event E is true. In the sequel, we deal with collections of forecasts $\{(E_i, \hat{p}_i)\}_{i=1}^m$, an important concept of which is *probabilistic coherence*.

Definition 1 *A set of forecasts* $\{(E_i, \hat{p}_i)\}_{i=1}^m$ *is* probabilistically coherent *if and only if they are implied by a joint distribution of probability over* X*.*

The following easy-to-prove lemma is important for the subsequent development.

Lemma 1 *Let* $C = C(\{E_i\}_{i=1}^m) \subseteq [0, 1]^m$ *be the set such that* $\{(E_i, \hat{p}_i)\}_{i=1}^m$ *is probabilistically coherent if and only if* $\hat{\mathbf{p}} = (\hat{p}_i)_{i=1}^m \in C$*. Then, for any set of events* $\{E_i\}_{i=1}^m$*,* C *is closed and convex.*

2.2 Alternating Projection Algorithms

Let $C_1, \ldots, C_l$ be closed, convex subsets of $\mathbb{R}^m$, whose intersection $C = \cap_{i=1}^l C_i$ is non-empty. For any $\hat{\mathbf{x}} \in \mathbb{R}^m$, let $P_C(\hat{\mathbf{x}})$ denote the least-squares projection of $\hat{\mathbf{x}}$ onto C, i.e.,

$$P_C(\hat{x}) := \arg\min_{\mathbf{x} \in C} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2.$$

Alternating projection algorithms [6] provide a way to compute $P_C(\cdot)$ given $\{P_{C_i}(\cdot)\}_{i=1}^l$. For example, the von Neumann-Halperin algorithm is one natural approach; it is depicted in Table 1.

Initialize:	$\mathbf{x}_0 := \hat{\mathbf{x}}$
Iterate:	$\mathbf{x}_{n+1} := P_{C_{(n \mod l)+1}}(\mathbf{x}_n)$

Table 1: The von Neumann-Halperin Algorithm

In words, the algorithm successively and iteratively projects onto each of the subsets. In the case where C_i is a linear subspace for all $i \in \{1, \ldots, l\}$, this algorithm was first studied by von Neumann [25] and subsequently by Halperin [12]. Much of the behavior of this algorithm can be understood through Theorem 1, the proof of which can be found in [3].

Theorem 1 *Let* $\{C_i\}_{i=1}^l$ *be a collection of closed, convex subsets of* $\mathbb{R}^m$ *whose intersection* $C = \cap_{i=1}^l C_i$ *is nonempty. Let* $\mathbf{x}_n$ *be defined as in the von Neumann-Halperin algorithm. Then, for every* $\mathbf{x} \in C$ *and every* $n \geq 1$*,*

$$\|\mathbf{x}_n - \mathbf{x}\|_2 \leq \|\mathbf{x}_{n-1} - \mathbf{x}\|_2.$$

Moreoever, $\lim_{n\to\infty} \mathbf{x}_n \in \cap_{i=1}^l C_i$*. If* C_i *is affine for all* $i \in \{1, ..., l\}$*, then* $\lim_{n\to\infty} \|\mathbf{x}_n - P_C(\hat{\mathbf{x}})\|_2 = 0$*.*

Often examined in the context of the *convex feasibility problem*, the von Neumann-Halperin algorithm has been generalized in various ways to address more general convex sets and non-orthogonal projections; accordingly, the algorithm often takes on other names (e.g., Bregman's algorithm, Dykstra's algorithm).

[1]The assumption that the variables are boolean is made merely to simplify exposition; all the subsequent discussion and results hold for more general multi-valued discrete variables.

3 The Panel Aggregation Problem

3.1 A Model

Suppose that each of m judges assesses the likelihood of a set of events; let $\mathcal{E}_i = \{(E_{ij}, \hat{p}_{ij})\}_{j=1}^{m_i}$ denote the set of forecasts provided by judge i. We assume that the events that make up $\mathcal{E}_i$ are defined over the same X for all $i = 1, \ldots, m$; however, we make no additional assumptions regarding the logical relationship between events in $\mathcal{E}_i$ and $\mathcal{E}_j$. In other words, we assume that panel members provide forecasts for the same "problem domain", but may assess the likelihood for altogether different, though perhaps logically related, events. With this model, the panel aggregation problem can be stated as follows:

> *Given a collection of forecasts $\{\mathcal{E}_i\}_{i=1}^{m}$ made by the panel, derive a coherent set of forecasts that jointly reflects the panel's expertise.*

3.2 The Coherent Approximation Principle

Osherson and Vardi [21] propose a *coherent approximation principle* (CAP) for addressing the panel aggregation problem. In particular, they suggest aggregating the panel's expertise by solving the following optimization problem:

$$\begin{array}{ll} \min & \sum_{i=1}^{m} \sum_{j=1}^{m_j} |p_{ij} - \hat{p}_{ij}|^2 \\ \text{s.t.} & \cup_{i=1}^{m} \{(E_{ij}, p_{ij})\}_{j=1}^{m_i} \text{ is coherent.} \end{array} \quad (1)$$

Here, the optimization variables are $\{p_{ij}\}$; the events in $\{E_{ij}\}$ and the probability assessments $\{\hat{p}_{ij}\}$ are the program data. Consistent with the definition of the panel aggregation problem in Section 3.1, the output of CAP is a coherent set of forecasts for the events in $\{E_{ij}\}$, and not (necessarily) a joint probability distribution over X.

By solving (1), one finds the coherent forecasts that are minimally different (with respect to squared-deviation) from those provided by the panel, intuitively preserving the "information" provided by the judges while gaining probabilistic coherence. From a statistical perspective, computing (1) can be interpreted as finding the maximum-likelihood coherent forecasts $\{p_{ij}\}$ given additive white noise corrupted observations $\{\hat{p}_{ij}\}$. Finally, CAP offers a geometric interpretation: by Lemma 1, there exists a closed convex set $C = C(\{E_{ij}\})$ that defines the numbers which comprise coherent forecasts for the events in question; $\hat{\mathbf{p}}$, a vector concatenation of $\{\hat{p}_{ij}\}$, lies outside this set. CAP suggests fusing the panel's expertise by computing the orthogonal projection of $\hat{\mathbf{p}}$ onto C. Henceforth, the forecasts determined by solving (1) will be referred to as the *CAP-Aggregate* for the panel.

As discussed in [21], solving (1) (and therefore, implementing CAP) is NP-Hard in the general case. In particular, note that checking whether a set of forecasts $\{(E_{ij}, \hat{p}_{ij})\}_{j=1}^{m_i}$ is probabilistically coherent can be reduced to solving (1); and checking for probabilistic coherence is strictly more general than checking whether the formulae that describe the events $\{E_{ij}\}$ are mutually satisfiable.

3.3 Related Work

The literature on the panel aggregation problem is expansive, as it has been touched upon in philosophy, law, statistics, risk analysis, and computer science. We refer the interested reader to the brief survey in [21] for an entry point to this voluminous body of literature. Here, for completeness, we survey the literature immediately relevant to CAP, and augment Osherson and Vardi's survey with additional connections to related work in computer science.

Linear averaging [8],[9] is arguably the most popular aggregation principle, given its simplicity and documented empirical success. To illustrate this natural approach, consider the panel exhibited in Table 2. Here, three judges provide forecasts for three events, a conjunction and its conjuncts. The "Aggregate" forecast is the simple un-weighted average of the three judges' forecasts. Though appealing, linear averaging is not without pitfalls, as can be illustrated with a few examples.

For instance, an underlying assumption in linear averaging is that each judge is probabilistically coherent. Averaging is appropriate under this assumption since (by Lemma 1) the linear averaged aggregate is probabilistically coherent whenever the individual judges are coherent. However, in applications of interest, the judges are humans, who are notoriously incoherent. For example, the *conjunction fallacy*, a robust finding from psychology [14], [24], demonstrates that human judges (even experts!) often assign higher probability to a conjunction that its conjuncts. Table 2 illustrates such a case. In particular, note that "Chris" is incoherent since the probability assigned to the event $p \wedge q$ is greater than the probability assigned to q, i.e., $0.6 > 0.0$; the linear averaged aggregate is similarly incoherent. Thus, though linear averaging naturally addresses *inter-judge disagreement*, it will not in general provide a coherent aggregate when individual judges are themselves incoherent.

	Alice	Bob	Chris	Aggregate
p	0.75	0.60	0.95	0.67
q	0.20	0.10	**0.00**	**0.10**
$p \wedge q$	0.10	0.10	**0.60**	**0.20**

Table 2: Linear Averaging: Incoherent Judges

In practice, human judges may be unable or unwilling to offer forecasts for every event in question. Communication constraints may preclude judges from collaborating, or individual judges may be find themselves unqualified to forecast the likelihood of particular events. For example, a market analyst may be unqualified to forecast events pertaining to specific stocks in the technology sector, but may be willing to discuss the correlation between the NASDAQ and currency exchanges. Such a case is illustrated in Table 3, where each judge provides an incomplete but coherent set of forecasts. The incoherence of the pairwise average aggregate demonstrates that linear averaging is also inappropriate in the case where judges provide only partial forecasts.

Thus, a natural question arises: how should one aggregate the opinion expressed by incoherent judges on overlap-

	Alice	Bob	Chris	Aggregate
p	0.75	0.60	**NA**	0.67
q	0.20	**NA**	0.00	**0.10**
$p \wedge q$	**NA**	0.40	0.00	**0.20**

Table 3: Linear Averaging: Partial Forecasts

ping but generally different sets of logically complex events? CAP addresses this question by generalizing linear averaging. In particular, note that the CAP aggregate *equals* the un-weighted averaged aggregate whenever probabilistically coherent judges provide forecasts for the same set of events.

Lindley et. al. [17] consider a Bayesian approach to reconciling probability forecasts, whereby "noisy" observations $\{\hat{p}_{ij}\}$ are assumed to arise from a coherent set $\{p_{ij}\}$. CAP can be viewed as a special-case of their model, since as discussed above, the solution to (1) admits a Bayesian interpretation as the maximum-likelihood coherent forecasts given additive white noise corrupted observations $\{\hat{p}_{ij}\}$. However, note that [17] sought to eliminate incoherence from a single judge, whereas CAP was introduced to address the panel aggregation problem. Moreover, Osherson and Vardi were motivated by non-statistical interpretations of CAP and as here, addressed the computational issue of implementing CAP.

Finally, a panel-aggregation problem is addressed in the "online" learning model, which is frequently studied in the machine learning community; see, for example, references [7], [11] and [19]. In that setting, a panel of experts provides predictions for the true outcome of set of events. A central agent bases its own forecast on a weighted average of the experts' predictions. Upon learning the truth, the agent suffers a loss, often specified by a quadratic penalty function. In repeated trials, the agent updates the weights of its weighted average, by taking into account the performance of each expert. Under minimal assumptions on the evolution of these trials, bounds are derived that compare the trial-averaged performance of the central agent with that of the best (weighted combination of) expert(s).

In contrast to the current framework, the online model typically assumes that each expert provides a forecast for the same event or partition of events. Thus, weighted averaging is an appropriate aggregation strategy in the online model, for the same reasons discussed above. For another difference with online learning, observe that the present model concerns a single "trial", not many.

4 A Scalable Approach to Aggregation

In principle, implementing CAP by solving (1) can be accomplished using quadratic programming. In the general case, this approach requires a representation of joint distributions on X, for which $O(2^n)$ free variables are necessary. For panels that assess relatively small numbers of events, the quadratic programming approach is nonetheless feasible. In cases of interest, hundreds of judges forecast thousands of events, yet off-the-shelf tools for solving quadratic programs do not scale.

Nevertheless, the logical complexity of the events assessed by human judges is usually bounded. For example, experts are often constrained to forecast events with no more than *three* literals (e.g., three-term conjunctions). The idea at the heart of our approach is to exploit such logical simplicity by decomposing (1) into a collection of small sub-problems, each of which can be solved quickly using off-the-shelf tools.

We now present our main result, a general algorithm for aggregating large corpora of probability forecasts. To aid exposition, let us do away with the multi-judge distinction by assuming that there is a single body of forecasts $\mathcal{E} = \{(E_i, \hat{p}_i)\}_{i=1}^m$. We do so without loss of generality, since we may construct $\mathcal{E}$ by pooling all the judges' forecasts into a single set. Also, let us assume that every event in $\{E_i\}_{i=1}^m$ is unique. Below, we demonstrate how this assumption may be relaxed.

4.1 A General Algorithm

To state our general algorithm, it is helpful to introduce a notion of *local coherence*. Let $\{(E_i, \hat{p}_i)\}_{i=1}^m$ be a collection of forecasts and let $\sigma \subseteq \{1, \ldots, m\}$. The requirement that $\{(E_i, \hat{p}_i)\}_{i=1}^m$ be probabilistically coherent can be relaxed by requiring only the subset $\{(E_i, \hat{p}_i)\}_{i\in\sigma}$ be coherent. For notational convenience, we henceforth say that $\{(E_i, \hat{p}_i)\}_{i=1}^m$ is *locally coherent with respect to* σ whenever $\{(E_i, \hat{p}_i)\}_{i\in\sigma}$ is coherent.

With this formalism, note that "global" coherence is recovered by taking $\sigma = \{1, \ldots, m\}$. Moreover, note that any probabilistically coherent set $\{(E_i, \hat{p}_i)\}_{i=1}^m$ must be locally coherent with respect to σ for all $\sigma \subseteq \{1, \ldots, m\}$.

With that, let us relax (1) by choosing a collection of subsets $\{\sigma_j\}_{j=1}^l$ and defining the following optimization problem.

$$\begin{array}{ll} \min & \sum_{i=1}^m |p_i - \hat{p}_i|^2 \\ \text{s.t.} & \{(E_i, p_i)\}_{i\in\sigma_j} \text{is coherent} \quad \forall j = 1, \ldots, l \end{array} \tag{2}$$

To emphasize, (2) is a relaxation of (1), since in general local coherence does not imply global coherence. However, this relaxation permits a geometric interpretation, as a projection onto the intersection of l convex sets. Thus, alternating projection algorithms are applicable to solving (2). In particular, an algorithm for solving (2) is detailed in Table 4; note that it is exactly the von Neumann-Halperin algorithm interpreted in the language of the panel aggregation problem.

In this algorithm, computation occurs in the inner loop, when projecting $\mathbf{q}$ onto a set of local coherence constraints. This computation requires only $|\sigma_j|$ forecasts, since $p_{tj,i} = q_i$ for all $i \notin \sigma_j$.

The crucial step in this algorithm is Step 1, designing $\{\sigma_j\}_{j=1}^l$. Intuitively, the fewer events that each subset contains, the faster each inner computation can run. However, as subsets get larger, a richer set of coherence constraints are represented and thus, the solution to (2) more closely approximates the CAP-aggregate. When designing $\{\sigma_j\}_{j=1}^l$,

Input:	$\{(E_i, \hat{p}_i)\}_{i=1}^m$
Initalize:	Auxiliary forecasts $\{(E_i, q_i)\}_{i=1}^m$, with $q_i := \hat{p}_i$.
Step 1:	Design $\{\sigma_j\}_{j=1}^l$ with $\sigma_j \subseteq \{1, \ldots, m\}$.
Step 2:	for $t = 1, \ldots, T$ for $j = 1, \ldots, l$ $\mathbf{p}_{tj} := \arg\min \sum_{i=1}^m \lvert p_i - q_i \rvert^2$ s.t. $\{(E_i, p_i)\}_{i \in \sigma_j}$ is coherent. Update $\{(E_i, q_i)\}_{i \in \sigma_j} \leftarrow \{(E_i, p_{tj,i})\}_{i \in \sigma_j}$
Output:	$\{(E_i, q_i)\}_{i=1}^m$.

Table 4: A von Neumann-Halperin Approach to Aggregation

one must therefore strike a balance between *approximation* and *speed*.

A natural way to make this trade-off is by exploiting the logical simplicity of the events in question. To illustrate, consider the case where the events in $\{E_i\}_{i=1}^m$ are constrained to be basic events, negations of basic events, and two-term conjunctions (or disjunctions) of the basic events or their negations. A sample set of events that meet these criteria are drawn in Figures 1, 2, and 3 (ignoring the dashed lines for a moment).

The linear averaging approach to aggregation can be viewed as a special case of this general method, where one subset is chosen per event; these subsets are depicted by the dashed lines in Figure 1. This highly local approach can be implemented very quickly, however the solution to (2) may poorly approximate the CAP-aggregate since few of the coherence constraints are represented. CAP, on the other hand, groups all the events into a single subset, requiring global coherence; this case is depicted in Figure 2. The CAP approach represents all the coherence constraints, but as discussed above, is computationally infeasible in practice.

A cleverer design may select subsets according to the logical relationship between the events in question. In Figure 3, for example, it is proposed to group basic events with their negations, and conjunction (disjunctions) with their corresponding conjuncts (disjuncts). By choosing *all* subsets of this form, we enforce a very strong set of local coherence constraints; crucially, however, each subset contains at most three events. Intuitively, solving (2) using these subsets will quickly approximate the CAP-aggregate given the balance we've struck between approximation and speed. This intuition is borne out in the experiments.

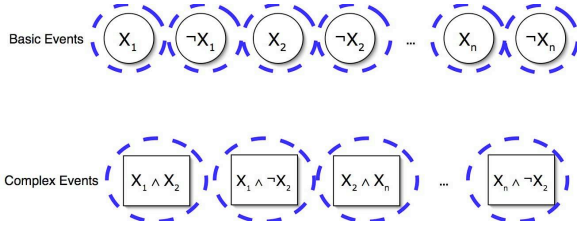

Figure 1: Linear Averaging

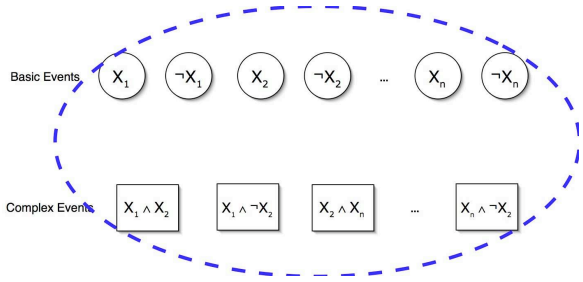

Figure 2: CAP

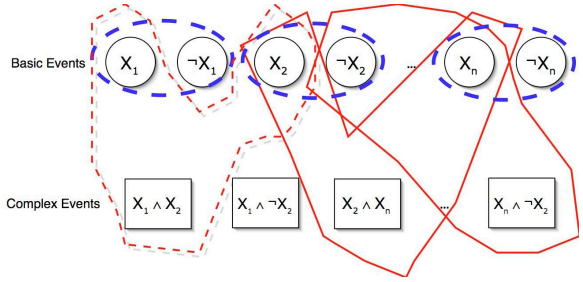

Figure 3: A Scalable Approach

4.2 Comments

First, let us emphasize that $\{\sigma_j\}_{j=1}^l$ is a design parameter. Depending on this design, the output forecasts may or may not be coherent; recall, the algorithm solves (2), a *relaxation* of CAP. Intuitively, however, for any $\{\sigma_j\}_{j=1}^l$ the output will be closer to coherence, since it will satisfy a set of local coherence constraints. This intuition is formalized by Theorem 2 below.

Second, in Step 2, the local coherence constraints are addressed in sequence. Note that this ordering is non-essential and parallelism may be introduced. In particular, two projections can occur simultaneously as long as there is no overlap between the events in question (i.e., two projections cannot change the same variables simultaneously).

Next, the assumption that each event in $\{E_i\}_{i=1}^m$ is unique can be relaxed. For any set $\{(E_i, \hat{p}_i)\}_{i=1}^m$ with N_j forecasts for each unique event $F_j \in \{E_i\}_{i=1}^m$, one can construct a new set $\{(F_j, \hat{q}_j)\}$ with $\hat{q}_j = \frac{1}{N_j} \sum_{i:E_i=F_j} \hat{p}_i$. Then, solving

$$\begin{array}{ll} \min & \sum_j N_j |p_j - \hat{q}_j|^2 \\ \text{s.t.} & \{(F_j, p_j)\} \text{ is coherent.} \end{array}$$

is equivalent to solving (1) with $\{(E_i, \hat{p}_i)\}_{i=1}^m$. The same trick can be applied to the relaxation (2), and the algorithm in Table 4 can be adjusted similarly.

Finally, the algorithm depicted in Table 4 permits a performance guarantee. In particular, assume that after learning the "truth" of the events in question, the accuracy of the forecasts in $\{(E_i, \hat{p}_i)\}_{i=1}^m$ is assessed using the *Brier score* [5], a quadratic penalty:

$$QP(\{(E_i, \hat{p}_i)\}) = \sum_{i:E_i=\text{TRUE}} (1-\hat{p})^2 + \sum_{i:E_i=\text{FALSE}} (0-\hat{p})^2 \quad (3)$$

The algorithm in Table 4 offers a stepwise improvement in accuracy as measured by the Brier score, independent of the truth or falsity of the events in question. Theorem 2 formalizes this important fact.

Theorem 2 *Let $\{(E_i, q_{T,i})\}_{i=1}^m$ denote the set of forecasts output by the algorithm after running T iterations with input forecasts $\{(E_i, \hat{p}_i)\}_{i=1}^m$. Then,*

$$QP(\{(E_i, q_{T,i})\}) \leq QP(\{(E_i, q_{T-1,i})\})$$

for every realizable truth assignment to the events in $\{E_i\}_{i=1}^m$. Moreover, as $T \to \infty$, the output forecasts converge (i.e., $\mathbf{q}_T$ converges in norm), and are locally coherent with respect to σ_j for all $j = 1, \ldots, l$.

The proof of Theorem 2 follows from Theorem 1 and de Finetti's Theorem [10], [23].

If $\{(E_i, \hat{p}_i)\}_{i=1}^m$ contains a single judge's forecasts (i.e., the algorithm is applied to eliminate intra-judge incoherence), then Theorem 2 predicts a step-wise improvement in the accuracy of that judge. If instead $\{(E_i, \hat{p}_i)\}_{i=1}^m$ contains a panel's forecasts, then Theorem 2 predicts that at each step, a randomly selected judge will improve on average.

Note that T, the number of iterations through the forecasts, is a second design parameter for this algorithm that in principle must be tuned. However, Theorem 2 demonstrates a sense in which performance is monotonic in T. Moreover, for any T, the output forecasts will be more accurate than the input forecasts (with respect to the Brier score), independent of the truth or falsity of the events in question.

5 Experiments

In this section, we empirically validate the aggregation algorithm presented in Section 4. In particular, our experiments focus on two issues: (i) the effect that aggregation has on the panel's forecasting accuracy and (ii) how the algorithm scales to large data sets, i.e., how "fast" the algorithm is in practice.

5.1 The Data

Five previously collected data sets will be used in these experiments. The STCK database was first published in [21] and contains forecasts made by MBA students at Rice University on events pertaining to 10 stocks in the third quarter of 2000; the FIN database is documented in [2] and summarizes forecasts made by students at Rice on events related to various economic indicators in the fourth quarter of 2001; the NBA1 and NBA2 data sets appeared in [2] and detail forecasts made by self-proclaimed basketball enthusiasts regarding the outcome of two Houston Rockets National Basketball Association games; the HSTN data set [13] contains forecasts made by Houston homeowners on events pertaining to the local real-estate market and pollution.

In each of the five data sets, subjects were asked to assess the likelihood of 34 randomly selected basic (10) and complex (24) events. The complex events were constrained to have one the following forms: $p \wedge q$, $p \wedge \neg q$, $p \vee q$, or $p \vee \neg q$. The number of subjects (i.e., the size of the panel) per data set is summarized in Table 5, as is the total number of basic events (i.e., the length of X) from which the forecasted events were constructed. Due to the random allocation of events per subject, multiple experts often provided forecasts pertaining to the same event. In Table 5, "Events/Agg" describes the number of unique events per panel.

	STCK	FIN	NBA1	NBA2	HSTN
Subjects	47	31	29	36	17
Basic Events	**30**	10	10	10	10
Events/Agg.	**1598**	1054	986	1224	578

Table 5: Data Summary

5.2 The Method

In each of the following experiments, we employ the aggregation algorithm detailed in Section IV. Since in each data set, complex events are constrained to one of the forms $p \wedge q$, $p \wedge \neg q$, $p \vee q$, or $p \vee \neg q$, subsets are chosen precisely as illustrated by Figure 3.

For every forecast reported in each database, the truth-value of the corresponding event is known. This allows us to asses the accuracy of various forecasts *a posteriori*. Here, accuracy is measured using the Brier score (3) and slope, which is defined as follows: if T denotes the number of true events in $\{E_i\}_{i=1}^m$, slope measures the stochastic accuracy of the forecasts $\{(E_i, \hat{p}_i)\}_{i=1}^m$ using $\frac{1}{T}\sum_{i:E_i=\text{TRUE}} \hat{p}_i - \frac{1}{m-T}\sum_{i:E_i=\text{FALSE}} \hat{p}_i$; higher slope indicates more accurate forecasts. We assess the accuracy of forecasts in four cases of interest.

- *Raw:* the accuracy of the judge's raw forecasts. The average accuracy of each judge's unprocessed forecasts is reported.
- *Individual:* the accuracy after eliminating intra-judge incoherence (i.e., after running the algorithm on each individual judge). The average accuracy of the judge's forecasts after processing is reported.
- *Aggregate:* the accuracy after aggregation using our method. The accuracy of each judge is assessed after replacing her original forecasts with the aggregate forecasts (for the same events); the average score is reported.
- *Linear Avg.:* the accuracy of the linear averaged aggregate. The accuracy of each judge is assessed after replacing her original forecasts with the linear averaged aggregate (again, for the same events); the average accuracy is reported.

5.3 Experiment 1: Scalability

Figures 4 and 5 detail the average Brier score achieved by the panel vs. the number of iterations (T) made by our algo-

rithm, in the *Individual* and *Aggregation* cases respectively. Note that the monotonicity of these plots is predicted by Theorem 2. In both cases and in every data set, the algorithm converges within 10 iterations through the forecasts.

From a computational perspective, the most interesting data set is the STCK database, since it contains the largest number of unique events per aggregate and the most basic events. On a 1GHz PowerPC G4, aggregating the database of 1598 forecasts took approximately 10s. In contrast, the rival method SAPA [21] was reported to take multiple hours. Incidentally, the time required to eliminate incoherence from individual judges was less than 0.6s.

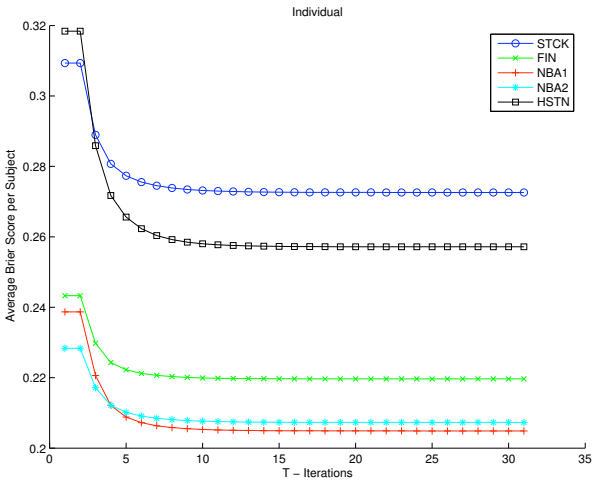

Figure 4: Eliminating Intra-subject Incoherence: Average Brier Score (*Individual*) vs. T.

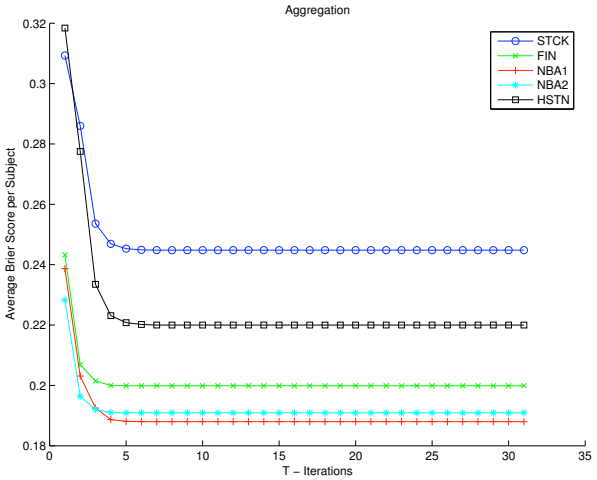

Figure 5: Aggregation: Average Brier Score (*Aggregation*) vs. T.

5.4 Experiment 2: Forecasting Accuracy

Osherson and Vardi [21] report three important empirical findings. First, they observe that eliminating intra-judge incoherence improves the forecasting accuracy of individual judges (i.e., *Individual* is better than *Raw*). Second, they observe that panel aggregation improves the forecasting accuracy of panel members (i.e., *Aggregate* improves over *Raw*). Finally, [21] reports that aggregation improves the accuracy of panel members as compared to incoherence-corrected forecasts (i.e., *Aggregate* improves over *Indivdual*). Discussed in part in reference [21], these findings are anticipated by de Finetti's theorem [10] when accuracy is assessed using the Brier score. However, Osherson and Vardi's findings hold up under alternative accuracy measurements including slope.

In the previous section, we documented a several orders of magnitude speed-up. Here, we question whether this has been achieved at the expense of accuracy. In particular, we question whether Osherson and Vardi's empirical observations hold up when using our method.

Tables 6 and 7 summarize the result for Brier score and slope, respectively. These results are in agreement with the findings of Osherson and Vardi except that the aggregate slopes are not consistently higher than for the individual application of our algorithm. As reported in reference [21] for the STCK dataset, the SAPA method yielded average per subject accuracy as 0.276 (*Indivdual*), while the "optimal" CAP calculation computed using quadratic programming yieled 0.272 (*Individual*). We thus conclude that the proposed method provides a significant computational speed-up while achieving competitive forecasting gains.

	STCK	FIN	NBA1	NBA2	HSTN
Raw	0.309	0.243	0.239	0.228	0.318
Individual	0.273	0.220	0.205	0.207	0.257
Aggregate	0.245	0.200	0.188	0.191	0.220
Linear Avg.	0.286	0.207	0.203	0.196	0.234

Table 6: Forecasting Accuracy: Brier Score

	STCK	FIN	NBA1	NBA2	HSTN
Raw	0.064	0.153	0.140	0.141	0.129
Individual	0.109	0.172	0.186	0.169	0.210
Aggregate	0.114	0.153	0.173	0.150	0.202

Table 7: Forecasting Accuracy: Slope

6 Discussion

6.1 Extensions

An underlying assumption of the current study is that the Brier-score (e.g., squared-error) is the appropriate measure for assessing forecasting accuracy and probabilistic (in)coherence. However, de Finetti's theorem, the von-Neumann-Halperin algorithm (and generalizations such as Dykstra's algorithm) have all been extended to a wide class of distance measures known as *Bregman divergences* [4] (which include the Brier-score and relative-entropy as special cases); for details, see [6] and [23]. As a result, our methods and analysis can be generalized to accommodate a large class of alternative accuracy measurements.

6.2 Graphical Models

The message-passing algorithm derived in Section IV is reminiscent of belief propagation, the sum-product algorithm, and junction-trees more generally[2]. It is thus natural to ask (i) whether CAP could be solved using an appropriate factor graph representation and the junction tree algorithm and (ii) whether the algorithm derived in Section IV can be viewed as an instantiation of one such approach. Addressing (ii) may require one to interpret alternating projection algorithms in the context of the junction-tree algorithm applied to a factor graph representation of our local coherence constraints. At the time of this writing, however, the authors have not fully resolved these issues.

6.3 Wireless Sensor Networks

Since researchers in wireless sensor networks are interested in similar aggregation problems, it is natural to ask whether these tools are applicable in WSN setting where the "experts" are electro-mechanical sensors. If in a given WSN application, sensors provide forecasts of probability for both logically simple and complex events, then these tools are immediately applicable. However, the general idea of relaxing a projection by exploiting an underlying notion of locality is more widely applicable. For example, in [22], a distributed algorithm is constructed for collaboratively training least-square kernel regression estimators. Similarly to above, the algorithm was derived using alternating projection algorithms applied to network topology dependent relaxation of the classical least-squares estimator.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, 2002.

[2] R. Batsell, L. Brenner, D. Osherson, S. Tsavachidis, and M. Y. Vardi. Eliminating incoherence from subjective estimates of chance. In *Proceedings of the 8th International Conference on the Principles of Knowledge Representation and Reasoning (KR 2002),* Toulouse, France, pages 353 – 364, 2002.

[3] H. H. Bauschke and J. M. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3):367–426, September 1996.

[4] L. M. Bregman. The relaxation method of finding a common point of convex sets andits application to the solution of problems in convex programming. *U. S. S. R. Computational Mathematics and Mathematical Physics*, 78(384):200–217, 1967.

[5] G. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78:1 – 3, 1950.

[6] Y. Censor and S. A. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications.* Oxford, New York, 1997.

[7] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth. How to use expert advice. *Journal of the ACM*, 44(3):427–485, 1997.

[2]The authors thank David Blei for pointing out this latter connection.

[8] R. Clemen and R. Winkler. Combining probability distributions from experts in risk analysis. *Risk Analysis*, 19:187 – 203, 1999.

[9] R. T. Clemen. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5:559–583, 1989.

[10] B. de Finetti. *Theory of Probability, Vol. 1.* John Wiley and Sons, New York, NY, 1974.

[11] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Computer and System Sciences*, 55:119–139, 1997.

[12] I. Halperin. The product of projection operators. *Acta. Sci. Math (Szeged)*, 23:96–99, 1962.

[13] M. E. Hendrix, P. R. Hartley, and D. Osherson. Real estate values and air pollution: Measured levels and subjective expectations. Discussion paper, Rice University, 2005.

[14] D. Kahneman and A. Tversky, editors. *Choices, Values, and Frames.* Cambridge University Press, New York, 2000.

[15] S. Kumar, F. Zhao, and D. Shephard. Collaborative signal and information processing in microsensor networks. *IEEE Signal Processing Magazine*, 19(2):13–14, 2002.

[16] R. Lempert, S. W. Popper, and S. C. Bankes. *Shaping the Next One Hundred Years: New Methods for Quantitative, Long-Term Policy Analysis.* Number MG-1626. RAND Corporation, Santa Monica, 2003.

[17] D. V. Lindley, A. Tversky, and R. V. Brown. On the reconciliation of probability assessments. *Journal of the Royal Statistical Society A*, 142 (Part 2):146–180, 1979.

[18] E. Lipton. New rules set for giving out antiterror aid. *The New York Times*, Jan 3 2006.

[19] N. Littlestone and M. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[20] M. G. Morgan and M. Henrion. *Uncertainty: a guide to dealing with uncertainty in quantitative risk and policy analysis.* Cambridge University Press, Cambridge, UK, 1990.

[21] D. Osherson and M. Y. Vardi. Aggregating disparate esimtates of chance. *Games and Economic Behavior,* in press, 2005.

[22] J. B. Predd, S. R. Kulkarni, and H. V. Poor. Distributed kernel regression: An algorithm for training collaboratively. In To appear in, *Proceedings of the 2006 IEEE Information Theory Workshop*, Punta del Este, Uruguay, March 2006.

[23] J. B. Predd, D. Osherson, S. R. Kulkarni, and H. V. Poor. A generalization of de Finetti's theorem. preprint, Princeton University, 2005.

[24] K. Tentori, N. Bonini, and D. Osherson. The conjunction fallacy: a misunderstanding about conjunction? *Cognitive Science*, 28(3):467–477, 2004.

[25] J. von Neumann. *Function Operators II.* Princeton University, Princeton, NJ, 1950.

[26] H. H. Willis, A. R. Morral, T. Kelly, and J. J. Medby. *Estimating Terrorism Risk.* Number MG-388. RAND Corporation, Santa Monica, 2006.