Yale University
Department of Computer Science

Uncertain Knowledge in Distributed Systems

Michael J. Fischer Lenore D. Zuck

YALEU/DCS/TR-604
January 1988

This work was supported in part by the National Science Foundation under grant DCR-8405478 and by the Office of Naval Research under Contract N00014-82-K-0154.

Uncertain Knowledge in Distributed Systems

Michael J. Fischer Lenore D. Zuck

Abstract

We present a formal system to reason about *implicit belief*. Implicit belief captures the (possibly probabilistic) information available to agents in probabilistic distributed systems. Our system also deals with non-determinism where *all* the non-deterministic choices are made at the beginning of the computation. We demonstrate the naturalness of our approach by offering new analyses and solutions to some classical distributed computing problems, namely the coordinated attack and authenticated Byzantine agreement.

1 Introduction

1.1 Uncertainty in Distributed Systems

Uncertainty is inherent in distributed systems and is what distinguishes their study from the study of "parallel computation". Uncertainty arises from many factors:

1. Lack of knowledge of system configuration.
2. Lack of knowledge of the protocol being run by other processors.
3. Lack of knowledge of inputs received at other sites.
4. Unreliability of hardware components of the processors or communication system.
5. Variability of processor step times.
6. Variability of message delivery times.
7. Unpredictability of random coin tosses.
8. Unpredictability of future external inputs.
9. Lack of compute power to extract knowledge from the available information.

The first three items concern uncertainties of an individual agent (process) in the system; these uncertainties are of facts that *are* known to an external agent with a global view of the entire system. Items 4–8 concern uncertainty about the system as a whole, i.e., what course the run of the system will take in the future. From the agent's local point of view, all of these items have the potential of introducing error into a computation, and all force

This work was supported in part by the National Science Foundation under grant DCR-8405478 and by the Office of Naval Research under Contract N00014-82-K-0154.

the agent to view its own knowledge with a degree of skepticism. In this work, we introduce a formal system that enables one to reason about the knowledge of an agent in a system that has elements of 1–8. Item 9 is of a slightly different nature as it concerns issues of computational complexity. It is a non-issue in most distributed systems (although it is a major issue in cryptographic systems [HMT87,FZ87,GMR85,TW87]). In the interest of simplicity we do not treat it in this paper, although we believe the formal system of reasoning about knowledge, probability, and time presented here can be extended to encompass the notions of relative knowledge and belief presented in [FZ87].

Generally speaking, uncertainty may be considered to be either *probabilistic* or *non-deterministic* in nature. If we have some a priori knowledge that uncertainty is determined by a random process independent of the operation of the system, then we can model it as a random variable, i.e., probabilistic. Else, we are forced to consider worst-case scenarios, namely, we view the cause of uncertainty as if it were controlled by an "adversary" who wants to cause the system to behave as "badly" as possible, i.e., non-deterministic.

1.2 Formal Treatment of Uncertainty

The goal of this paper is to define a formal system adequate to describe the kind of "knowledge" possessed by agents in distributed systems that involve elements of uncertainty. Our approach is similar to that of [FH87,HMT87], but it differs in two major respects:

1. Our system treats uncertainty due to lack of information and uncertainty due to the unpredictability of future random events in a uniform way. Thus, we can give an exact characterization of the "probabilistic knowledge" possessed by an agent at the end of a protocol as well as at the beginning.
2. We handle non-determinism explicitly in our model, rather than trying to allow for it implicitly by making certain sets unmeasurable. The resulting system appears to be more expressive as well as being simpler and more natural.

Consider a simple 2-party protocol between agents p and q in which p flips a private unbiased coin *and nothing further happens.* q cannot see the outcome of the coin toss. Thus, there are only three global states in the system: the initial state s_0 before the coin has been flipped, the state s_h in which the coin has landed "heads", and the state s_t in which the coin has landed "tails". Because the coin is unbiased, the probabilities of reaching s_h from s_0 and of reaching s_t from s_0 are both $1/2$.

In state s_0, q knows that the coin will land heads with probability $1/2$. Halpern, Moses, and Tuttle (cf. [HMT87]) would express this fact by the formula

$$\mathsf{K}_q^{1/2} \Diamond \mathsf{heads}$$

which says that q knows that with probability at least $1/2$, the statement "at the next state (after the coin has been flipped), the coin will be heads" holds, where the probability is taken over the possible future extensions of the run. In this example, there are two equally likely runs, one ending in s_h and the other in s_t. Since **heads** is true at the end of the first run and false at the end of the second, q reasons in s_0 that **heads** will hold at the next step $1/2$ of the time.

After the coin has been flipped, q still does not know the outcome (since p has not told him). From q's perspective, it is still just as likely that the coin landed "heads" as it is that it landed "tails". Intuitively, the statement

$$\mathsf{K}_q^{1/2}\mathsf{heads}$$

should now hold and reflect this uncertainty in q's knowledge. However, the [HMT87] logic does not permit this uncertainty to be expressed, for the only uncertainty it can accommodate is that resulting from future randomness. After the coin has been flipped, the outcome is determined and there is no more future uncertainty. The global state is now either s_h or s_t. In s_h, heads holds with probability 1, and in s_t it holds with probability 0, but in neither state does it hold with probability 1/2. Since q does not know which is the true state, the formula $\mathsf{K}_q^{\alpha}\mathsf{heads}$ only holds for α equal to the minimum of those two probabilities, which is 0.

In our system, we *can* formalize the fact that at the end of this protocol q considers the two states s_h and s_t to be equally likely and therefore has confidence 1/2 that the coin has landed heads. Confidence, the way we use it, is well defined; its intuitive meaning is that, if q bets even money on heads and the game is repeated many times, then its expected loss is zero. To avoid confusion with true knowledge, we call our notion of knowledge with a possibility of error *implicit belief*, and we denote it with the symbol B instead of K.

In the above example, the formula

$$\mathsf{B}_q^{1/2}\mathsf{heads}$$

holds at both s_h and s_t. It should be read, "q believes with confidence 1/2 that the coin has landed heads". It might seem that in s_t the formula should not hold. However, q has no clue whether the real state is s_t or s_h as it cannot distinguish one from the other. The only additional information q obtained about the outcome of the coin flip is that it had been determined. Therefore, q reasons that 1/2 of the times in which it finds itself in this situation (of the coin having been flipped but not knowing the outcome), the true state is s_h and the other half of the times it is s_t. Since heads is true in s_h, it is quite reasonable for q to believe with confidence 1/2 that the coin is "heads".

More generally, i has only partial information about the true global state s of the system, so i must consider any state s' possible for which its local view is the same as for s. Let $[s]_i$ be the set of all such states. Even though i cannot distingiush those states, it does have some a priori knowledge about the *likelihood* of being in each of those states (assuming for the time being that we are considering a purely probabilistic system, i.e., with no non-determinism). Namely, since the probability distribution on the runs of the system is common knowledge to all agents, i can determine for each state $s' \in [s]_i$ the probability of the system being in s', given that the system is in some state of $[s]_i$ and can therefore determine the probability α_φ of being in a state in which φ holds. If $\alpha \leq \alpha_\varphi$, we say that agent i believes φ with confidence at least α in state s, which we write as

$$s \models \mathsf{B}_i^{\alpha}\varphi.$$

When we add temporal operators, we obtain formulas such as $\diamondsuit\mathsf{heads}$ mentioned above which are neither true nor false at a given state but rather have a certain probability of

being true there. Definining belief with confidence α of such formulas requires a slight generalization of the above definition. Details are presented in Section 3 below.

Non-determinism presents a special problem in reasoning about probabilistic protocols, for how can one talk about the probability of a statement being true when that probability is affected by non-deterministics choices? The answer is that one can't, but once *all* the non-deterministic choices are fixed, the resulting system is a pure probabilistic one and the probability of a formula being true is well defined. We consider only initial non-determinism, that is, the non-deterministic choices must be made before the protocol is run and before the outcomes of any of the coin tosses are known. This allows us to model uncertainty in network parameters, network configuration, initial inputs, and protocols run by the other agents, but it does not allow us to handle external inputs that arrive during the execution of the protocol and which may depend (in unknown ways) on the execution history up to that point. We leave the extension of our formalism to full non-determinism for future work.

Fagin and Halpern [FH87] define a formal logic for reasoning about knowledge and probability which is based on Kripke structures that have been extended to include a "subjective" probability space for each agent i at each global state s. Subjective probability generalizes the indistinguishability relationship of classical knowledge logic and tells for each set of states S the agent's belief that the true state belongs to S when in fact the true state is s. Because of non-determinism, it does not make sense to assign probabilities to all possible sets of states. Fagin and Halpern note that it is okay to leave such problematic sets unmeasurable since a probability space does not require that all sets be measurable. For example, in game G_1 of the next section, the probability of ending up in the set $\{s_1, s_3\}$ is either 0.5 or 0.8, depending on the initial non-deterministic choice, so there is no single "right" measure to assign to it.

The [FH87] model allows additional generality that might be used in trying to capture the non-determinism more exactly. For example, it permits an agent to have different subjective probability spaces in in different global states. While this additional generality may make the logic more expressive, we also find it very unnatural that an agent's subjective probabilities should depend on information not available to it.

The main contributions of this paper are the following:

1. We present a formal system to reason about "knowledge" in probabilistic systems, where knowledge is subject to a probability of error. Our system treats uncertainty due to lack of information and uncertainty due to unpredictability of probabilistic events uniformly. This allows one to make probabilistic statements about a random event *after* its occurrence and before information about its outcome has been obtained.

2. The degree of confidence expressed by our notion of implicit belief corresponds exactly to the worst-case conditional probability of a fact holding, given only the information in the local view of the agent. Thus, we have captured all of the probabilistic knowledge available to the agent in the worst case, that is, all that the agent can count on in the face of adversity.

3. Our system sheds light on some classical problems of distributed computing. Namely, by requiring only high confidence rather than certainty in the outcome of a protocol, we can obtain easy solutions to a large variety of problems, some of which are are otherwise provenly insoluble. We demonstrate our approach on the the Coordinated

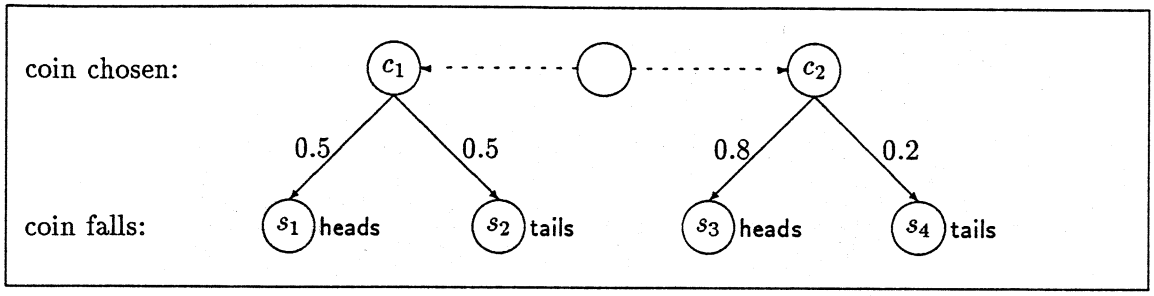

Figure 1: The system for game G_1.

attack problem. We also show that implementations of the simple authenticated Byzantine agreement protocol of [DS83] using digital signatures do not attain either common knowledge or certain agreement. All they attain is agreement with a high degree of confidence.

2 The Computational Model

Throughout the paper we frequently refer to two slightly more involved "coin-flipping" examples G_1 and G_2, which are illustrated in Figures 1 and 2. A minor variation of G_1 is extensively discussed in [FH87].

G_1: This game is played by two agents, p and q. p holds two seemingly identical coins, c_1 which is fair and c_2 which has a 0.8 bias towards "heads". p chooses non-deterministically one of the coins and flips it. The coin falls either heads or tails.

G_2: This game is G_1 with an additional step: After p chooses a coin and flips it, it flips some coin c_3 that has a 0.8 bias to "heads". If c_3 falls heads, p tells q the result of the first coin flip. If c_3 falls tails, p lies to q about the result of the first coin flip.

We model a *terminating synchronous probabilistic distributed system* by a set of finite trees, each of which corresponds to some nondeterministic choice that could be made in the system, i.e., we assume that all the nondetreministic choices are made at the beginning. For example, in G_1 there are two possible trees, T_1 which is rooted at c_1, and T_2 which is rooted at c_2. Each node in each of the trees is labelled by some distinct *global state*, e.g., c_1, s_1, s_4, etc. An internal tree node s which has k outgoing edges leading to $s_1, \ldots, s_k$, labelled by $\beta_1, \ldots, \beta_k$ respectively, corresponds to a probabilistic action that can lead from s to s_i with probabilisty β_i, for every $i = 1, \ldots, k$. This of course implies that $\sum_{i=1}^{k} \beta_i = 1$. We use $pr(s, s')$ to denote the probability of reaching s' from s in one step, i.e., $pr(s, s')$ is the label of the edge leading from s to s' if s' is an immediate successor of s in the tree, and $pr(s, s') = 0$ otherwise.

For every node s, we denote by $tree(s)$ the tree that s is in. We associate with s a probability $pr(s)$, which is the probability of reaching s from the root of $tree(s)$, i.e., $pr(s)$ is the product of the labels of edges on the path leading from the root of $tree(s)$ to s. For example, in G_1, $pr(c_1) = pr(c_2) = 1$, $pr(s_1) = pr(s_2) = .5$, $pr(s_3) = .8$ and $pr(s_4) = .2$.

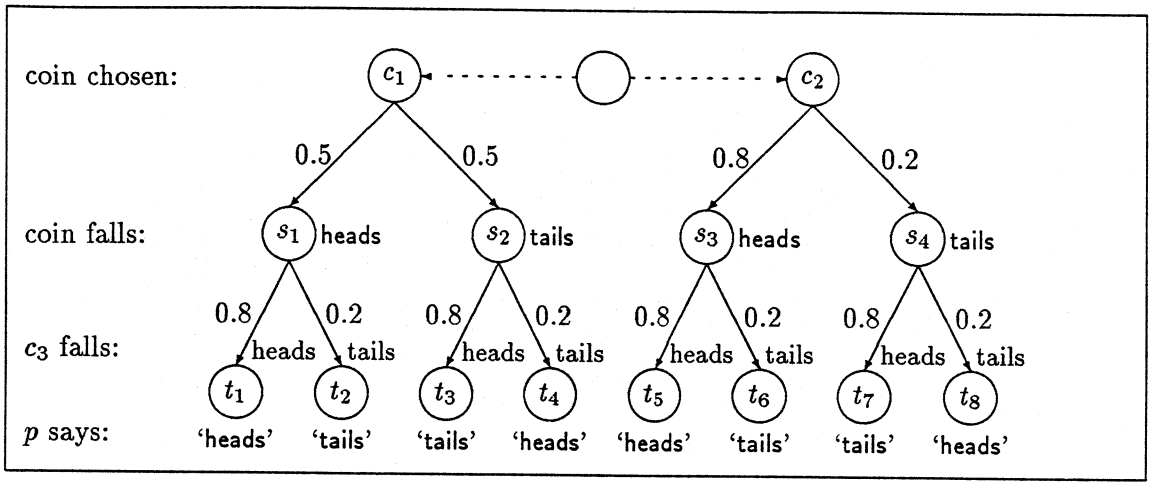

Figure 2: The system for game G_2.

Let S be an independent subset of states (i.e., S contains no two states that are on the same path). We define the probability of a node s relative to the set S, denoted by $pr(s \mid S)$, as the conditional probability of the real state being s, given that the real state is in $S \cap tree(s)$. Formally,

$$pr(s \mid S) = pr(s) \,/\Big(\sum_{t \in S \cap tree(s)} pr(t) \Big).$$

Note that by this definition, $pr(s \mid S) = pr(s \mid S \cap tree(s))$, thus, the probability is taken only over those states of S that are in the same tree as s.

For example, consider Figure 2. Denote the left tree by T_1 and the right tree by T_2. Then $pr(t_1) = 0.4$, $pr(t_4) = 0.1$, $pr(t_6) = 0.16$. Let $S = \{t_1, t_4, t_6\}$. Then

$$\begin{aligned} pr(t_1 \mid S) &= pr(t_1 \mid S \cap T_1) \\ &= pr(t_1 \mid \{t_1, t_4\}) = 0.4/(0.4 + 0.1) = 0.8. \end{aligned}$$

Similarly, $pr(t_4 \mid S) = 0.2$ and $pr(t_6 \mid S) = 1$.

Let $\mathcal{S}$ denote the set of all possible global states. We assume a set Φ of *basic facts*, and an evaluation function α that maps every state $s \in \mathcal{S}$ and every fact $\varphi \in \Phi$ to a real number $\alpha_s(\varphi) \in [0, 1]$. The number $\alpha_s(\varphi)$ denotes the *degree of confidence* which we associate with the truth of φ in s. Returning to G_1, let $\Phi = \{C1, C2, \mathsf{heads}, \mathsf{tails}\}$ where $C1$ (resp. $C2$) stands for "p chose c_1 (resp. c_2)", and heads (resp. tails) stands for "the coin fell 'heads' (resp. 'tails')". Then, we define:

- $\alpha_s(C1) = 1$ and $\alpha_s(C2) = 0$ for every $s \in T_1$.
- $\alpha_s(C1) = 0$ and $\alpha_s(C2) = 1$ for every $s \in T_2$.
- $\alpha_{c_1}(\mathsf{heads}) = \alpha_{c_1}(\mathsf{tails}) = \alpha_{c_2}(\mathsf{heads}) = \alpha_{c_2}(\mathsf{tails}) = 0$.
- $\alpha_{s_1}(\mathsf{heads}) = \alpha_{s_3}(\mathsf{heads}) = \alpha_{s_2}(\mathsf{tails}) = \alpha_{s_4}(\mathsf{tails}) = 1$.

- $\alpha_{s_1}(\mathsf{tails}) = \alpha_{s_3}(\mathsf{tails}) = \alpha_{s_2}(\mathsf{heads}) = \alpha_{s_4}(\mathsf{heads}) = 0$.

We extend Φ by closing it under boolean operations ($\neg$, $\vee$, and $\wedge$) and the temporal operators $\Diamond\!\!\!\!+$ (next time) and $\Diamond\!\!\!\!\geq$ (eventually). We extend the degree of confidence function α using the following rules:[1]

$$\begin{aligned}
\alpha_s(\neg(\varphi)) &= 1 - \alpha_s(\varphi) \\
\alpha_s(\varphi \vee \psi) &= \max\{\alpha_s(\varphi), \alpha_s(\psi)\} \\
\alpha_s(\varphi \wedge \psi) &= \max\{1 - (1 - \alpha_s(\varphi)) - (1 - \alpha_s(\psi)), 0\} \\
&= \max\{\alpha_s(\varphi) + \alpha_s(\psi) - 1, 0\} \\
\alpha_s(\Diamond\!\!\!\!+\varphi) &= \sum_{s' \in \mathcal{S}} pr(s, s') \cdot \alpha_{s'}(\varphi) \\
\alpha_s(\Diamond\!\!\!\!\geq\varphi) &= \max\{\alpha_s(\varphi), \sum_{s' \in \mathcal{S}} pr(s, s') \cdot \alpha_{s'}(\Diamond\!\!\!\!\geq\varphi)\}
\end{aligned}$$

For example, in G_1,

$$\alpha_{c_1}(\Diamond\!\!\!\!+\mathsf{heads}) = \alpha_{c_1}(\Diamond\!\!\!\!+\mathsf{tails}) = 0.5, \quad \alpha_{c_2}(\Diamond\!\!\!\!+\mathsf{heads}) = 0.8, \quad \alpha_{c_2}(\Diamond\!\!\!\!+\mathsf{tails}) = 0.2,$$

and

$$\begin{aligned}
\alpha_{c_1}(\Diamond\!\!\!\!+(\mathsf{heads} \vee \mathsf{tails})) &= .5 \cdot \alpha_{s_1}(\mathsf{heads} \vee \mathsf{tails}) + 0.5 \cdot \alpha_{s_2}(\mathsf{heads} \vee \mathsf{tails}) \\
&= 0.5 \cdot \max\{1, 0\} + 0.5 \cdot \max\{0, 1\} = 1.
\end{aligned}$$

Let S be some subset of $\mathcal{S}$. We extend α to capture the degree of confidence of formulae $\varphi \in \Phi$ in the set S, denoted by $\alpha_S(\varphi)$. Intuitively, $\alpha_S(\varphi)$ is our degree of confidence that φ is true given that we know the true state is in S. We formally define it by:

$$\alpha_S(\varphi) = \min_{T \in \mathcal{T}} \sum_{s \in S \cap T} \alpha_s(\varphi) \cdot pr(s \mid S).$$

The summation expresses the degree of confidence that φ holds in S, given that the true state is in tree T. Because the tree is chosen non-deterministically, we minimize over the possible trees T.

For example, consider G_2 where $\alpha_t(\mathsf{heads}) = 1$ and $\alpha_t(\mathsf{tails}) = 0$ for every $t \in \{t_1, t_2, t_5, t_6\}$, and $\alpha_t(\mathsf{heads}) = 0$ and $\alpha_t(\mathsf{tails}) = 1$ for every $t \in \{t_3, t_4, t_7, t_8\}$. Let $S = \{t_1, t_4, t_6\}$, then:

$$\begin{aligned}
\alpha_S(\mathsf{heads}) &= \min_{T \in \{T_1, T_2\}} \sum_{s \in S \cap T} \alpha_s(\mathsf{heads}) \cdot pr(s \mid S) \\
&= \min\{\sum_{s \in \{t_1, t_4\}} \alpha_s(\mathsf{heads}) \cdot pr(s \mid \{t_1, t_4\}), \sum_{s \in \{t_6\}} \alpha_s(\mathsf{heads}) \cdot pr(s \mid \{t_6\})\} \\
&= \min\{\alpha_{t_1}(\mathsf{heads}) \cdot 0.8 + \alpha_{t_4}(\mathsf{heads}) \cdot 0.2, \alpha_{t_6}(\mathsf{heads}) \cdot 1\} \\
&= \min\{1 \cdot 0.8 + 0 \cdot 0.2, 1 \cdot 1\} = \min\{0.8, 1\} = 0.8.
\end{aligned}$$

Intuitively, this means that if all one knows is that the system is in one of S's states, then we can bet, with 80% probability of success, that heads holds.

[1] When we say that 'we extend and Φ by closing it under some operators', we really mean that construct a new Φ' which we close under the new as well as the old operators, and then term it Φ. We also implicitly assume that all the previous semantic definitions hold for the new Φ.

3 Belief

Consider the system of G_1. If the coins look identical, then agent p, who chooses the coin and flips it, cannot distinguish between the states in each of the pairs $\{c_1, c_2\}$, $\{s_1, s_3\}$, and $\{s_2, s_4\}$. It can however distinguish between elements of different pairs. On the other hand, q can only distinguish between the sets $\{c_1, c_2\}$ and $\{s_1, \ldots, s_4\}$ but cannot distinguish between pairs of elements in the same set.

We assume that for each agent $i \in \mathcal{A}$ the states of the system are partitioned by some equivalence relation $\sim_i$, where $s \sim_i s'$ if agent i cannot distinguish between s and s'. For every state s and agent i, we denote by $[s]_i$ the set of states that are indistinguishable from s by i, so $[s]_i = \{t \mid s \sim_i t\}$.

For example, in G_2, $\sim_p$ is the equivalence relation induced by the partition $\{\{c_1, c_2\}, \{s_1, s_3\}, \{s_2, s_4\}, \{t_1, t_5\}, \{t_2, t_6\}, \{t_3, t_7\}, \{t_4, t_8\}\}$, and $\sim_q$ is the equivalence relation induced by $\{\{c_1, c_2\}, \{s_1, \ldots, s_4\}, \{t_1, t_4, t_5, t_8\}, \{t_2, t_3, t_6, t_7\}\}$.

Consider now agent q when the system G_2 is in t_2, i.e., when c_1 was chosen, flipped and fell **heads**, and p told q '**tails**'. q cannot tell whether the system is in t_2, t_3, t_6 , or t_7. It however knows that if c_1 was chosen (i.e., the 'real' state is in T_1), then in 0.8 of the cases **tails** is true, and if c_2 was chosen, then in 0.5 of the cases **tails** is true. Similarly, if c_1 was chosen then in 0.2 of the cases **heads** is true, and if c_2 was chosen then in 0.5 of the cases **heads** is true. Therefore, q believes that no matter which coin is chosen, **tails** is true in at least 0.5 of the cases, and **heads** is true in at least 0.2 of the cases.

Let $\mathsf{B}_q^\beta \varphi$ denote that "q *believes*, with degree of confidence at least β, that φ holds". Then, in t_2,

$$\mathsf{B}_q^{0.5}\mathsf{tails} \quad \text{and} \quad \mathsf{B}_q^{0.2}\mathsf{heads}.$$

Formally, we say that for every agent $i \in \mathcal{A}$, probability $\beta \in [0,1]$, and formula $\varphi \in \Phi$, $\mathsf{B}_i^\beta \varphi$ holds in a state $s \in \mathcal{S}$ iff φ is true in $[s]_i$ with degree of confidence at least β, i.e.,

$$s \models \mathsf{B}_i^\beta \varphi \ \text{ iff } \ \alpha_{[s]_i}(\varphi) \geq \beta$$

We next extend the set Φ to by adding the belief operators B_i^β for every $i \in \mathcal{A}$ and $\beta \in [0,1]$ to the set of operators in Section 2. We extend α to by adding the following case to the definition of Section 2:

$$\alpha_s(\mathsf{B}_i^\beta \varphi) = \begin{cases} 1 & \text{if } s \models \mathsf{B}_i^\beta \varphi \\ 0 & \text{otherwise} \end{cases}$$

If for some $s \in \mathcal{S}$, $i \in \mathcal{A}$, and $\varphi \in \Phi$, $s \models \mathsf{B}_i^1 \varphi$, then we say that in s agent i *knows* φ, and abbreviate $s \models \mathsf{B}_i^1 \varphi$ to $s \models \mathsf{K}_i \varphi$. Note that $s \models \mathsf{K}_i \varphi$ if $\alpha_{[s]_i}(\varphi) = 1$, i.e., if φ is true with certainty in all the states that are indistinguishable to i from s, so that our notion of knowledge coincides with the "classical" definitions (see, e.g., [HM84,FI86]).

Let us return to G_1. At the beginning, the system is in either c_1 or c_2. Both p and q believe that the coin will fall **heads** with probability at least 0.5, and **tails** with probabililty at least 0.2. Indeed, we check that for every $i \in \{p, q\}$ and $c \in \{c_1, c_2\}$,

$$\alpha_{[c]_i}(\diamondsuit\!\!\!\!+\mathsf{heads}) \ = \ \min_{T \in \{T_1, T_2\}} \sum_{s \in [c]_i \cap T} \alpha_s(\diamondsuit\!\!\!\!+\mathsf{heads}) \cdot pr(s \mid [c]_i)$$

$$
\begin{aligned}
&= \min\{\alpha_{c_1}(\diamondsuit\!\!\!\!+\!\!\!\!\diamondsuit\mathsf{heads}) \cdot pr(c_1 \mid \{c_1\}), \alpha_{c_2}(\diamondsuit\!\!\!\!+\!\!\!\!\diamondsuit\mathsf{heads}) \cdot pr(c_2 \mid \{c_2\})\} \\
&= \min\{0.5 \cdot 1, 0.8 \cdot 1\} = 0.5
\end{aligned}
$$

and similarly that $\alpha_{[c]_i}(\diamondsuit\!\!\!\!+\!\!\!\!\diamondsuit\mathsf{tails}) = 0.2$. Hence, $c \models \mathsf{B}_i^{0.5}(\diamondsuit\!\!\!\!+\!\!\!\!\diamondsuit\mathsf{heads})$ and $c \models \mathsf{B}_i^{0.2}(\diamondsuit\!\!\!\!+\!\!\!\!\diamondsuit\mathsf{tails})$.

However, after the coin has been flipped (i.e., in $s_1, \ldots, s_4$), p knows the result of the coin flip while q has gained no additional information about the result of the coin flip. Indeed, we can see that

$$\alpha_{[s_1]_p}(\mathsf{heads}) = \alpha_{\{s_1,s_3\}}(\mathsf{heads}) = 1 \text{ and } \alpha_{[s_1]_p}(\mathsf{tails}) = \alpha_{\{s_1,s_3\}}(\mathsf{tails}) = 0,$$

whereas

$$\alpha_{[s_1]_q}(\mathsf{heads}) = \alpha_{\{s_1,\ldots,s_4\}}(\mathsf{heads}) = 0.5 \text{ and } \alpha_{[s_1]_q}(\mathsf{tails}) = \alpha_{\{s_1,\ldots,s_4\}}(\mathsf{tails}) = 0.2.$$

So $s_1 \models \mathsf{K}_p\mathsf{heads}$, whereas $s_1 \models \mathsf{B}_q^{\beta}\mathsf{heads}$ is only true for $\beta \leq 0.5$. This corresponds to our intuition that p knows the outcome but q has learned nothing of it.

4 Coordinated Attack

Consider the Coordinated Attack problem as stated in [HM84]:

> Two divisions of army are camped on two hilltops overlooking a common valley. In the valley awaits the enemy. It is clear that if both divisions attack the enemy simultaneously they will win the battle, whereas if only one division attacks it will be defeated. The divisions do not initially have plans for launching an attack on the enemy, and the commanding general of the first division wishes to coordinate a simultaneous attack (at some time the next day). ... The generals can only communicate by means of a messenger. Normally, it takes the messenger one hour to get from one encampment to the other. However, it is possible that he will get lost in the dark, or, worse yet, captured by the enemy. ... How long will it take to coordinate an attack?

A correct solution (protocol) should guarantee:

Safety: If either party attacks, then they both attack at the same time.

It is shown in [HM84], that no correct solution to the problem will ever result in a coordinated attack. The results of [HM84] apply even if we assume some fixed probability β of the messenger successfully delivering a message within one hour.[2]

Suppose however that we are given such a probability β, and we look for solutions that satisfy some weaker safety requirement. For example, consider the γ-*weak coordinated attack problem* in which we require:

γ-Weak Safety: The probability that both parties attack at the same time, given that one party attacks, is at least γ.

[2]This observation is due to John Geanakoplos.

If $\gamma \leq \beta$ then the problem has a trivial solution: The first general (say p) sends a message to the other general (say q) with the attack time t, and then attacks at this time. If q receives the message, he also attacks at the designated time. Thus, p always attacks at time t, and since q receives p's message with probability β, q attacks at time t with probability β. Since $\beta \geq \gamma$, γ-weak safety is satisfied.

There are also solutions when $\gamma > \beta$. For example, if k is such that $(1-\beta)^k \leq (1-\gamma)$, then p can send k messengers to q carrying identical messages, and q attacks if it receives one or more messages. This occurs with probability at least $1-(1-\beta)^k \geq \gamma$.

Thus, we obtain:

Theorem 1 *The γ-weak coordinated attack problem has a correct solution for any $\beta > 0$, where β is the probability of the messenger successfully delivering the message, providing at least $\lceil \log(1-\gamma)/\log(1-\beta) \rceil$ messengers are available.*

The crucial point in the [HM84] proof that the problem cannot be solved is that the parties need to obtain *common knowledge* about the attack time. (See the discussion there about common knowledge.) The system we set forth is much weaker, as it allows p to attack at time t when it only *believes* the other party will attack but is not certain.

Theorem 2 *In any protocol solving the γ-weak coordinated attack problem and any global state s, if p attacks in s then*

$$s \models \mathsf{B}_p^\gamma q \text{ attacks},$$

and if q attacks in s then

$$s \models \mathsf{B}_q^\gamma p \text{ attacks}.$$

We can also prove a kind of converse to Theorem 2:

Theorem 3 *Consider any protocol C such that for some $\gamma \in [0,1]$ and for every global state s, if p attacks in s then*

$$s \models \mathsf{B}_p^\gamma q \text{ attacks},$$

and if q attacks in s then

$$s \models \mathsf{B}_q^\gamma p \text{ attacks}.$$

Then C solves the γ'-weak coordinated attack problem for $\gamma' = \gamma/(2-\gamma)$.

For comparison, the [HM84] proof relates to systems that guarantee that p attacks in state s only when

$$s \models \mathsf{K}_p q \text{ attacks}.$$

5 Authenticated Byzantine Agreement

Authenticated Byzantine Agreement (ABA) is Byzantine agreement under the assumption of authentication. See [DS83] for a thorough discussion on the subject. For example, it is said there that:

"..., we assume a protocol that will prevent any processor from introducing a new value or message into the information exchange and claiming to have received it from another ..."

Indeed, all the Byzantine agreement protocols proposed there make heavy use of some ideal authentication scheme that guarantees the above.

For example, consider the simple protocol P for achieving ABA in [DS83]:

1. Initially, every processor p has a set of values $C_p = \emptyset$, and some default value v_d.
2. At step 1, the sender sends a signed message with its value to all the processors.
3. At every step $k = 2, \ldots, t+1$, every processor p that received a properly signed message m in the previous step containing a value $v \notin C_p$, adds v to C_p, places his signature on m to obtain a new message m', and sends m' to all processors. A message received in step r is *properly signed* if it is signed by r distinct processors, the first of which is the sender and the last of which is the process from which the message was received.
4. At the end of step $t+1$, every processor p for which $C_p = \{v\}$ for some v chooses that v, and every other processor p chooses the default value v_d.

Dolev and Strong [DS83] show that the above protocol indeed guarantees Byzantine agreement if the number of faulty processors is at most t. However, suppose that the signature scheme is not totally secure, so that under certain circumstances a faulty processor can forge the signature of a reliable one. Consider the case, for example, that the sender is nonfaulty and sent $v \neq v_d$ in step 1. At step $t+1$, a faulty processor might send to some of the correct processors a message that contains another value v' together with the forged signature of the sender and (valid) signatures of the t faulty processors. Those correct processors receiving this bogus message will choose v_d, whereas the remaining correct processors will choose $v \neq v_d$, thereby violating Byzantine agreement.

If the sender is faulty, then it is sufficient for the faulty processors to forge a signature of any one correct processor. The argument proceeds along the same lines.

The scenarios described above might not be very likely; however, they have a positive probability of occurring when authentication is implemented through cryptographic techniques. For example, a signature might be forged simply through random coin flips; thus, forgery is always possible with probability at least 2^{-N}, where N bounds the message length. Thus, a real-life implementation of this protocol does *not* achieve Byzantine agreement since agreement sometimes fails to be reached.

What then is achieved? We define

γ-Weak Agreement: With probability at least γ, all correct processors choose the same value.

γ-Weak Liveness: If the sender is non-faulty, then with probability at least γ, all correct processors choose the sender's value.

A protocol achieves γ-weak Byzantine agreement if it satisfies γ-weak agreement and γ-weak liveness for any choice of faulty processors. We then have the following theorem.

Theorem 4 *If the probability of the faulty processors successfully forging the signature of a reliable processor is at most* $1-\gamma$*, then protocol* P *achieves* γ*-weak Byzantine agreement.*

Dwork and Moses [DM86] introduce the notion of Simultaneous Byzantine Agreement (SBA), which is Byzantine agreement in which all processors choose values at the same step. They show that any protocol that achieves SBA must satisfy

$$p \text{ chooses } v \implies \mathsf{K}_p \text{ (every correct } q \text{ chooses } v)$$

for every correct processor p. Protocol P, when run in an idealized environment with perfect authentication, achieves SBA, for all processors decide at the same step (cf. [DM86]).

A γ-weak Byzantine agreement protocol does not satisfy SBA since it does not even achieve agreement. We define a corresponding weak notion of SBA, termed γ-SBA, by replacing the condition of γ-weak agreement with:

γ-Weak Simultaneous Agreement: With probability at least γ, all correct processors choose the same value at the same step.

In the full paper, we prove the following:

Theorem 5 *Every* γ*-SBA protocol satisfies*

$$p \textit{ decides } v \implies \mathsf{B}_p^{\gamma}(\textit{every correct } q \textit{ decides } v)$$

for every correct processor p*.*

6 Further Work

Real-world systems must make decisions based on uncertain information. We have shown how to model uncertainty in the framework of knowledge logics, and we have shown how allowing for uncertainty in two classical problems of distributed computing radically alters their properties. We believe this work is important not only for the formal machinery it provides but also to help clarify people's thinking in making subtle distinctions between probabilistic and non-deterministic choices, and we expect to be influential in future distributed computing research.

We leave as an open problem the extension of our framework to eliminate the assumption that all non-deterministic choices are made before the protocol has begun. To do so will require one to handle alternations of non-deterministic and probabilistic choices, which will give rise to expressions of alternating minimization and summation operators.

In order to reason about cryptographic protocols, it is necessary to introduce *feasibility* into notions of knowledge and belief. Intuitively, computational complexity considerations are like dark glasses over the eyes of an agent. Even though an agent's local view of two states is not the same, he may not be able to make any useful distinctions between them and thus should believe a fact with the same confidence as if the states were totally indistinguishable to him. We address this issue in [FZ87] in which we express what an accepting verifier believes *at the end* of an interactive proof of "knowledge" using concepts of relative knowledge and belief developed there. We are currently working to extend those concepts to the temporal logic framework presented here.

Acknowledgements

We would like to thank Young-il Choo for helpful and stimulating discussions.

References

[DM86] C. Dwork and Y. Moses, Knowledge and common knowledge in a Byzantine environment I: crash failures (extended abstract), *Theoretical Aspects of Reasoning about Knowledge: Proceedings of the 1986 Conference* (J. Y. Halpern, ed.), Morgan Kaufmann, 1986, pp. 149–170.

[DS83] D. Dolev and H. R. Strong, Authenticated algorithms for Byzantine agreement, *SIAM Journal on Computing* **12**:4, 1983, pp. 656–666.

[FH87] R. Fagin and J. Y. Halpern, Reasoning about knowledge and probability: preliminary report, 1987. To be presented at the conference on Theoretical Aspects of Reasoning about Knowledge, March 1988.

[FI86] M. J. Fischer and N. Immerman, Foundations of knowledge for distributed systems, *Theoretical Aspects of Reasoning about Knowledge: Proceedings of the 1986 Conference* (J. Y. Halpern, ed.), Morgan Kaufmann, 1986, pp. 171–186.

[FZ87] M. J. Fischer and L. D. Zuck, *Relative knowledge and beilief*, Technical Report YALE/DCS/TR 589, Yale University, 1987.

[GMR85] S. Goldwasser, S. Micali, and C. Rackoff, The knowledge complexity of interactive proof-systems, *Proc. 17th ACM Symp. on Theory of Computing*, 1985, pp. 291–304.

[HM84] J. Y. Halpern and Y. Moses, Knowledge and common knowledge in a distributed environment, *Proc. 3rd ACM Symp. on Principles of Distributed Computing*, 1984, pp. 50–61. A revised version appears as *IBM Research Report RJ 4421*, Aug., 1987.

[HMT87] J. Y. Halpern, Y. Moses, and M. Tuttle, A knowledge-theoretic analysis of zero knowledge, 1987. To be presented at the 20th ACM Symp. on Theory of Computing, May 1988.

[TW87] M. Tompa and H. Woll, Random self-reducibility and zero knowledge interactive proofs of possession of information, *Proc. 28th IEEE Symp. on Foundations of Computer Science*, 1987, pp. 472–482.