Determining Confidence When Integrating Contributions from Multiple Agents

Raphen Becker[*]
Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01002
raphen@gmail.com

Daniel D. Corkill
Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01002
corkill@cs.umass.edu

ABSTRACT

Integrating contributions received from other agents is an essential activity in multi-agent systems (MASs). Not only must related contributions be integrated together, but the confidence in each integrated contribution must be determined. In this paper we look specifically at the issue of confidence determination and its effect on developing "principled," highly collaborating MASs. Confidence determination is often masked by ad hoc contribution-integration techniques, viewed as being addressed by agent trust and reputation models, or simply assumed away. We present a domain-independent analysis model that can be used to measure the sensitivity of a collaborative problem-solving system to potentially incorrect confidence-integration assumptions. In analyses performed using our model, we focus on the typical assumption of independence among contributions and the effect that unaccounted-for dependencies have on the expected error in the confidence that the answers produced by the MAS are correct. We then demonstrate how the analysis model can be used to determine confidence bounds on integrated contributions and to identify where efforts to improve contribution-dependency estimates lead to the greatest improvement in solution-confidence accuracy.

General Terms

Measurement, performance, experimentation, theory

Keywords

Contribution integration, confidence, analysis models

This work is supported by the "Fusion Based Knowledge for the Future Force" ATO program and the "Advanced REsearch Solutions - Fused Intelligence with Speed and Trust" program at the U.S. Army RDECOM CERDEC Intelligence and Information Warfare Directorate, Fort Monmouth, NJ, under contract W15P7T-05-C-P621. The views contained in this paper are the authors'.

[*]Now at Google, Inc., Mountain View, California

AAMAS'07 May 14–18 2007, Honolulu, Hawai'i, USA.

1. INTRODUCTION

Integrating contributions received from others is an essential activity in multi-agent systems (MASs) [2, 6], blackboard [3, 5] and other collaborating software systems [7, 4] (where problem solving is performed by multiple knowledge sources), and in real life [12]. Not only must related contributions received from other agents be integrated together, but the confidence in each integrated contribution must be determined. The issue of confidence determination and its effect on developing "principled," highly collaborating MASs is the focus of this paper.

Consider a simple "thought experiment" MAS application. There are three "observer" agents, each able to observe a coin toss and report whether it saw a "heads" or a "tails." A fourth "integrating" agent receives the observer reports and combines them to produce an overall answer. Each observer agent has undergone extensive certification of its coin-flip acuity and reports the flip correctly 80% of the time. We wish to improve the accuracy of the four-agent MAS over that of a single observer agent by having the integrating agent combine the reports received from the three observers.

To clarify the issue of confidence determination, we make the following assumptions in this paper:

- trust [8] is not an issue—agents always do their best to provide accurate contributions
- agent contributions are not always correct—despite its best efforts, an agent's contributions may be wrong
- received contributions can be integrated trivially—contributions to be integrated have identical semantics
- the probability of an agent's contributions being accurate is known (0.8 in our thought experiment)
- the "confidence" in a contribution reflects the probability that the contribution is correct

Suppose the integrating agent receives a *heads* report from each observer agent. What confidence should the integrating agent assign to the integrated *heads* result? Common approaches used when combining contributions are to: 1) assume independence among the contributions, 2) use ad hoc heuristics to approximate dependencies, 3) avoid the issue by using only a single ("best") contribution, or 4) not determine a confidence for the integrated contribution at all.

Our integrating agent could assume that the contributions are independent, apply Bayes rule, and assign the integrated result a confidence of .985 (reflecting the collective corroboration of the three contributions). But what if the contributions are not independent? In this thought experiment, let's implement each of our observation agents as a simple function that is given the actual (ground truth) value of the toss and reports that value except for 20% of the time when it reports the opposite side. The decision of when to report an incorrect observation is determined by using a pseudo-random number generator. Our observer accuracy is 80% (and we've saved a lot of coding). If each agent uses a different seed for its random-number generator, the times when the agents' reports are incorrect is independent and the integrating agent's confidence assignments (based on assumed independence) are realistic. However, if all observer agents use the same seed, their pseudo-random numbers will be identical and they will be mistaken at the same times. If our integrating agent accounts for this contribution dependency, the confidence in the integrated result should remain at 0.8 no matter how many observer reports are integrated. (The reports are fully redundant.)

Of course, real MAS agents are not about random-number generators and ground-truth cheats, and the interactions among agents are often significantly more involved than sending complete results to an integrating agent. A real agent-based coin-flip detector might consist of a number of camera agents (each with its own camera), low-level image processing agents (with different processing approaches and algorithms), feature-detector agents (*Eye*, *Nose*, *Hair*, *Building*, *Head*, etc.), side-assessor agents (again, potentially with different knowledge/strategies), and so on. Yet this complexity only serves to mask the fundamental issue of determining the confidence of integrated contributions. Even if we know the accuracy of individual contributions, we need to account for the confidence uncertainty inherent in integrating them together and how that confidence uncertainty propagates over long reasoning chains and agent interactions. A "principled" integration agent in our thought experiment should represent the confidence in the result as the interval [0.8..0.985], with any additional knowledge of the dependency characteristics of observer contributions applied to reducing this confidence interval.

The remainder of the paper is structured as follows. First, we introduce a domain-independent analysis model that can be used to measure the sensitivity of complex, collaborative problem-solving systems to potentially incorrect confidence-integration assumptions. We then demonstrate how the analysis model can be used to determine confidence bounds on integrated contributions and to identify where efforts to improve contribution-dependency estimates lead to the greatest improvement in solution-confidence accuracy. We conclude with future directions to explore in contribution integration.

2. THE ANALYSIS MODEL

We have developed a Bayesian network model [9, 10] that facilitates analysis of confidence integration in an arbitrarily complex collaborative problem-solving system. While the analysis model is Bayesian, the system being modeled can have any representation and inference mechanisms. The model is designed for off-line analysis, so we are not concerned with distributed application of the model.

Accuracy and correlated errors

Let's begin with a closer look at the independence of our coin-toss contributions in terms of accuracy and correlated errors when two agents have different accuracy probabilities. Suppose agent A_i has an accuracy of $P(A_i)$ and agent A_j has an accuracy of $P(A_j)$, where $P(A_i) \geq P(A_j)$. Figure 1 shows an intuitive pictorial of this two-agent situation. Region x in the fully dependent graphic depicts the probability that both agents are correct ($P(A_j)$) and region z depicts the probability that both agents are incorrect ($1 - P(A_i)$). Region y is the probability that the contributions disagree ($1 - (P(A_j) + (1 - P(A_i)))$). When the agents are fully independent, the probability that they agree is $P(A_i)P(A_j)$, shown in region $\bar{x}$, and where they disagree ($(1 - P(A_i))(1 - P(A_j))$), shown in region $\bar{z}$. Region $\bar{y}$ is the probability that the independent agents disagree. Note that $x \geq \bar{x}$ and $z \geq \bar{z}$.

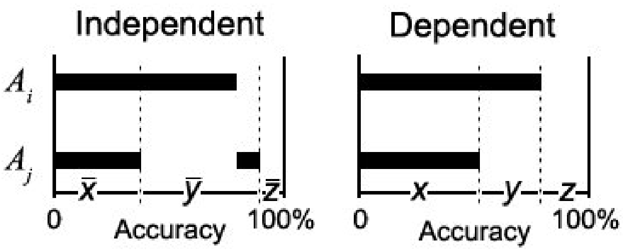


Figure 1: Independence of two contributions. The black bars indicate instances in which the contribution is correct (corresponding instances are aligned vertically.) The instances where A_i is correct are arranged together on the left side of each graph, with instances where A_j is correct also on the left as much as possible within the A_i ordering. The percentage of black versus white is the accuracy of the agent. When the agent contributions are dependent, errors are maximally correlated. Every time the less accurate agent (A_j) is correct, A_i is also correct (region x). Every time the more accurate agent (A_i) is wrong, A_j is also wrong (region z). The remaining region (y) is where their contributions disagree. When the contributions from each agent are independent, A_i and A_j are correlated, but conditionally independent given the coin flip. Region $\bar{x}$ represents where both agents are correct, region $\bar{z}$ where both are wrong, and region $\bar{y}$ is where their contributions disagree.

This pictorial helps in visualizing what occurs as the individual accuracies and the difference between them change. As agent accuracies increase, the contributions naturally become more correlated and the difference between the dependent and independent extremes becomes smaller. At 100% accuracy there is no difference. Therefore, if our agents are close to perfect, incorrect confidence-integration assumptions will have little effect in comparison to mediocre agents operating in the same system.

Given this view of contribution independence, let's look at applying it to integrated contributions.

Concurrent processing model

A set of decisions made by agents is concurrent if the decisions are made without knowledge of any of the other decisions in the set. Figure 2 (a) illustrates the concurrent decisions of three agents, A_1, A_2, and A_3. These agents are all making decisions related to node W, which is the state of the world that the agents are trying to understand. In our model, W can take on any number of values, though for simplicity we will illustrate it for two values, $true$ and $false$. A_1, A_2 and A_3 are all trying to match W's value. This abstraction could represent a coin flip ($W = \{heads, tails\}$) and three agents that are processing images of W and trying

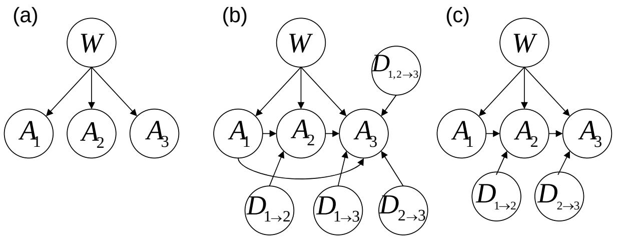


Figure 2: The concurrent processing model. (a) shows nodes representing independent contributions, (b) shows the entire set of dependencies, and (c) shows the subset of dependencies we represent in our model.

to figure out if W was $heads$ or $tails$.

In Figure 2 (a) the contributions made by the three agents are independent, while in Figure 2(b) we have represented all potential dependencies between them. The D nodes in Figure 2(b) explicitly represent the error dependencies among the A nodes. However, the full-dependency model quickly becomes unwieldy as the number of agents grows. To keep the number of dependencies manageable, we simplify the model to the linear structure shown in Figure 2(c), where only adjacent nodes are directly dependent on each other and conditional independence is assumed between all non-adjacent nodes. This simplification is reasonable because, at worst, this modeled error will understate the potential error of assuming full contribution independence. If the error is a concern in the simplified 2(c) model, it will be even greater in the full-dependency model.

Use of the D nodes is not necessary, and they could be marginalized out of the model by placing all the dependency information between A_i and A_{i-1} into the link between them. We include an additional node, D_i, for each A_i node (except the first) in our model, because we feel this separation makes the model more intuitive. Each D_i node represents whether the errors in A_i and A_{i-1} are dependent or independent. When $D_i = false$ then A_i is conditionally independent of A_{i-1} given W, $P(A_i|W) = P(A_i|W, A_{i-1})$. When $D_i = true$, however, the errors are maximally correlated with one another (as in the Dependent graphic in Figure 1).

In effect, D_i breaks $P(A_i|A_{i-1}, W)$ down into a mixture of distributions: one distribution for when A_i and A_{i-1} are dependent and one when they are independent, $P(A_i|A_{i-1}, W) = P(D_i = t)P(A_i|A_{i-1}, W, D_i = t) + P(D_i = f)P(A_i|A_{i-1}, W, D_i = f)$. The likelihood that they are dependent is simply the prior probability on D_i. This is similar to the idea of separability [11]. $P(D_i)$ captures the likelihood—not that A_i and A_{i-1} will have the same answer—but that they have the same answer because they are dependent. We can change this prior to represent the range of dependency from conditionally independent ($P(D_i = t) = 0$) to fully dependent ($P(D_i = t) = 1$).

Table 1 shows what this means in terms of the conditional probably table (CPT) for node A_2. All the parameters used in our model are defined in Table 2. With this simple dependency, the directionality of the arrow between A_{i-1} and A_i does not matter. However, given three or more nodes their ordering will make a difference because we are only representing a subset of the possible dependencies (between adjacent nodes) and changing the ordering changes which subset of dependencies we are modeling.

A_1	W	D_2	$true$
t	t	t	$min(\frac{B_2}{B_1}, 1.0)$
f	f	t	$1 - min(\frac{B_2}{B_1}, 1.0)$
t	f	t	$min(\frac{1-B_2}{1-B_1}, 1.0)$
f	t	t	$1 - min(\frac{1-B_2}{1-B_1}, 1.0)$
$-$	t	f	B_{01}
$-$	f	f	$1 - B_{01}$

Table 1: The CPT for A_2 showing how the model parameters are used when there are no parents.

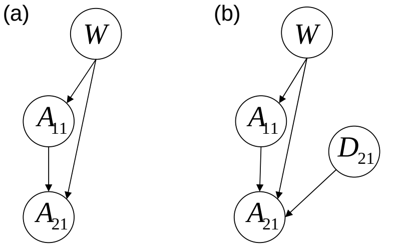


Figure 3: The simplest sequential processing model with one parent node. The result produced by A_{11} is used by A_{21} in its processing. (a) shows the model without an explicit dependence. (b) makes the dependency explicit with the node D_{21}.

Sequential processing model

Two decisions are sequential if the output of one decision is influenced by the output of another decision. While in the concurrent model the dependency between the contributions of two agents could range from conditionally independent to completely dependent, in the sequential model the agents are inherently dependent from the start. For example, in the coin-flip MAS, the *Hair* agent might identify a region in the image that has hair-like texture, while the *Nose* agent hypothesizes a nose. Both of these contributions are sent to the *Head* agent which evaluates those contributions and hypothesizes a region it believes to be someones head on the coin. Hair and nose are correlated with the obverse side of the coin, and identification of a head has an even stronger correlation with the obverse side. The additional processing that the *Head* agent performed using the contributions it received improves the MAS's confidence in the answer over just hair and nose. This kind of sequential processing is represented in Figure 3 for one parent and Figure 4 for two.

Figure 4(a) shows a simple example of sequential processing where contributions of agents A_{11} and A_{12} form the input to agent A_{22}. The integrated contribution A_{22} depends on the output of its parents as well as whether its parents' contributions are correct, hence the link from W to A_{22}. When the parents are both correct there is some chance that the child will introduce an error (parameter I_{ij}^2 in Table 2). When the parents are both wrong there is some chance that the child will be able to compensate for the parent errors and determine the correct answer anyway (parameter C_{ij}^2 in Table 2). When the parents disagree then the child is left to come up with its own answer (parameter B_{ij} in Table 2). Clearly this is a simplification of the possible relationships between parents and children, but it is sufficient for our analysis model.

A down side to sequential processing comes from detri-

Parameter	Label	Description
W	prior event	Prior probability of event
D	dependency	Prior probability that the nodes are dependent
D^p	parent	Probability that child is dependent on its parent instead of its neighbor to the left, when it is dependent (as given by D)
B_{ij}	base correctness	Probability that child A_{ij} is correct given no parents or parents who disagree
I_{ij}^1	introduce errors	Probability that child is incorrect given its one parent is correct
I_{ij}^2	introduce errors	Probability that child is incorrect given its two parents are correct
C_{ij}^1	correct errors	Probability that child is correct given its one parent is incorrect
C_{ij}^2	correct errors	Probability that child is correct given its two parents are incorrect

Table 2: The parameters used in the CPTs in the general analysis model.

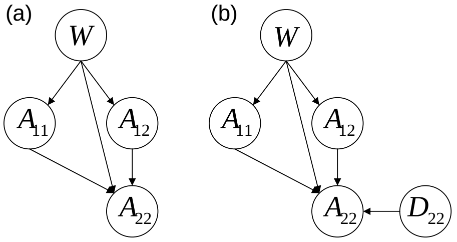


Figure 4: The sequential processing model with two parents. A_{22} takes as input the output of both A_{11} and A_{12}. (a) shows the model without the explicit dependency, (b) shows the model with the explicit dependency D_{22}.

(a) CPT for A_{21}

A_{11}	W	D_{21}	*true*
t	t	f	$1 - I_{21}^1$
f	f	f	I_{21}^1
t	f	f	$1 - C_{21}^1$
f	t	f	C_{21}^1
t	$-$	t	1.0
f	$-$	t	0.0

(b) CPT for A_{22}

A_{11}	A_{12}	W	D_{22}	*true*
t	t	t	f	$1 - I_{22}^2$
f	f	f	f	I_{22}^2
t	t	f	f	$1 - C_{22}^2$
f	f	t	f	C_{22}^2
t	f	t	f	B_{22}
f	t	t	f	B_{22}
t	f	f	f	$1 - B_{22}$
f	t	f	f	$1 - B_{22}$
$-$	t	$-$	t	1.0
$-$	f	$-$	t	0.0

Table 3: The CPTs for nodes A_{21} (Figure 3b) and A_{22} (Figure 4b) in the sequential model.

mental *information cascades* [1]. If one agent makes a mistake it can mislead the next agent in the sequence, and this wrong result can cascade, leading the entire sequence to generate incorrect results. While this phenomenon could be captured in the existing links between the parent nodes and the child, we explicitly represent it with a dependency node, D_{ij}. This is similar to the approach that we used with concurrent processing to explicitly represent contribution dependencies (see Figures 3(b) and 4(b)). When D_{ij} is true, A_{ij} takes the same value as $A_{i-1,j}$ instead of depending on both of its parent nodes. Table 3(a) gives the CPT for the node A_{21} in the one parent case, and Table 3(b) for the two parent case. This representation can be extended easily to more than two parents. Again, Table 2 provides a description of the parameters used.

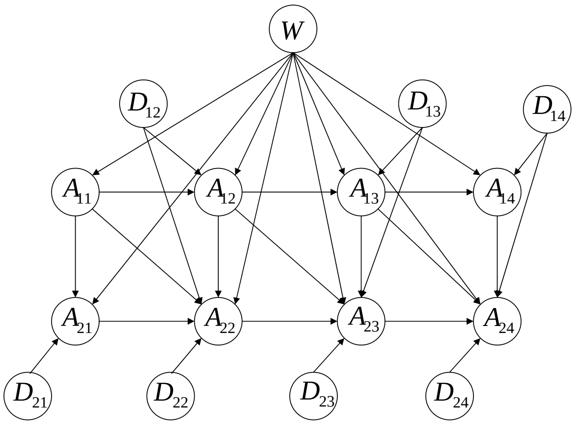


Figure 5: A combined model.

Combining both models

The concurrent and sequential processing models can be combined into a general model of an arbitrarily complex system, such as the grid shaped model shown in Figure 5. The rows in the grid represent concurrent processing, while the columns model the sequential processing.

The concurrent and sequential processing models interact in two ways. First, we merge the dependency node D_{ij} for A_{ij} from the two models into one node with three values:

- *parent*—A_{ij} produces the same result as its parent directly above due to information cascading.
- *neighbor*—A_{ij} produces the same result as its neighbor to the left due to dependency.
- *false*—A_{ij} is not forced to answer a particular way due to a dependency.

Changing the priors on the D_{ij} nodes allows the levels of dependence between the nodes to be adjusted easily in order to see how confidence and expected error changes for different degrees of dependence.

The second change introduced by combining the two processing models deals with the accuracy of a node when its parents are dependent. The basic idea is that when A_{ij}'s parents are independent, the rate that it introduces errors or corrects errors will be different then when its parents are dependent. For example, suppose a *Head* agent trying to locate a head on an image of a coin is provided the image

$A_{i,j\text{-}1}$	$A_{i\text{-}1,j\text{-}1}$	$A_{i\text{-}1,j}$	W	$D_{i\text{-}1,j}$	D_{ij}	***true***
–	t	t	t	p/f	f	$1 - I^2_{ij}$
–	f	f	f	p/f	f	I^2_{ij}
–	t	t	f	p/f	f	$1 - C^2_{ij}$
–	f	f	t	p/f	f	C^2_{ij}
–	t	f	t	p/f	f	B_{ij}
–	f	t	t	p/f	f	B_{ij}
–	t	f	f	p/f	f	$1 - B_{ij}$
–	f	t	f	p/f	f	$1 - B_{ij}$
–	t	t	t	n	f	$1 - I^1_{ij}$
–	f	f	f	n	f	I^1_{ij}
–	t	t	f	n	f	$1 - C^1_{ij}$
–	f	f	t	n	f	C^1_{ij}
–	t	f	t	n	f	B_{ij}
–	f	t	t	n	f	B_{ij}
–	t	f	f	n	f	$1 - B_{ij}$
–	f	t	f	n	f	$1 - B_{ij}$
t	–	–	t	–	n	$min(\frac{B_{ij}}{B_{i,j\text{-}1}}, 1.0)$
f	–	–	f	–	n	$1 - min(\frac{B_{ij}}{B_{i,j\text{-}1}}, 1.0)$
t	–	–	f	–	n	$min(\frac{1-B_{ij}}{1-B_{i,j\text{-}1}}, 1.0)$
f	–	–	t	–	n	$1 - min(\frac{1-B_{ij}}{1-B_{i,j\text{-}1}}, 1.0)$
–	–	t	–	–	p	1.0
–	–	f	–	–	p	0.0

Table 4: The CPT for A_{ij}. This illustrates the interaction between the sequential dependency and the concurrent dependency. The ***D*** nodes have three values: n for ***neighbor***, p for ***parent***, and f for ***false***.

of the coin and contributions from two *Hair* agents containing the regions of the image that they each believes to be hair. It would be reasonable to expect that if the regions in the contributions are in agreement, and they were generated independently, then the *Head* agent would be less likely to make a mistake than if the *Hair* agents agreed simply because some factor in the environment forced them to agree (such as excessive wear on the coin).

We handle this by adding a link from the dependency node of the relevant parent to the child. When that dependency is *parent* or *false* we have the same probabilities as in Table 3(b). However, when the dependency is *neighbor*, we use different parameters I^1_{ij} and C^1_{ij} instead of I^2_{ij} and C^2_{ij}. See Table 4 for the entire CPT.

3. ANALYSES USING THE MODEL

The combined model can be used to to measure the sensitivity of complex MASs with both concurrent and sequential processing to incorrect confidence-integration assumptions. In this section we demonstrate using the model by analyzing a family of hypothetical systems in which the model nodes are arranged in a rectangular grid shape. Many specific systems would often result in other shape models, based on their contribution interactions. For example, an hourglass shape could reveal the effects of confidence-integration in a system with information bottlenecks. However, by varying the width, depth, and other parameters of our hypothetical grid-system family, we can easily explore some general characteristics of confidence integration.

We make several assumptions about the contribution-exchange structure of our hypothetical systems. For example, there are two parent nodes for the child nodes in each step of sequential processing while a specific system might have some nodes with many parents and some with as few as one. Another assumption is that our agent contributions are equally likely ($P(A = t|W = t) = P(A = f|W = f)$).

After the MAS determines its answer, we are interested in computing the probability that the answer is correct. In this analysis, we assume that we can identify the system's "answer" by observing the values of each A_{ij} in the model. Since the answer the system gives should be consistent with the value having the highest probability, we do not need to explicitly represent it in our model. We assume the system answer is the value of W that maximizes $P(W|A)$ where A is the set of all A_{ij}. Our confidence is then $max_W P(W|A)$. If we were modeling a specific system instance, we might be more interested in values at one or more specific nodes.

This is how we compute the confidence of a particular problem instance in our grid systems, but we want to evaluate the sensitivity of assuming the nodes are independent, $P(D) = 0$, when in fact they are not, $P(D) > 0$. First we can measure our **expected belief** that our answer is correct by taking a weighted average over all possible instances of the observed data:

$$\sum_A P(A) \cdot max_W P(W|A)$$

If agents assume that $P(D) = 0$, then our expected belief is:

$$\sum_A P(A) \cdot max_W P(W|A, P(D) = 0)$$

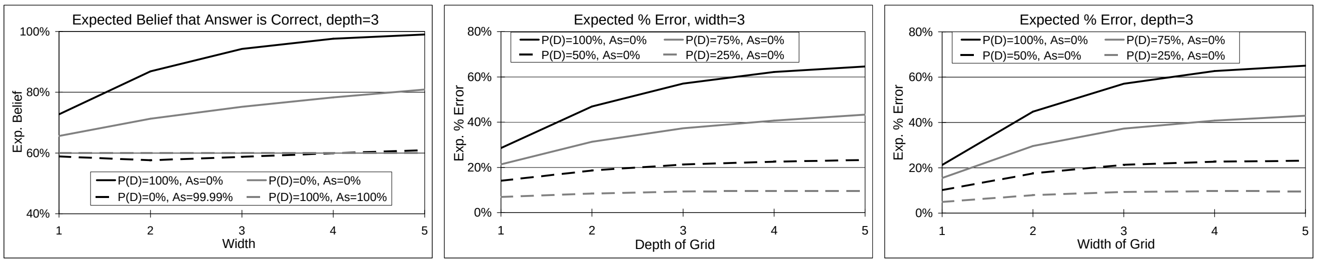
Figure 6(a) illustrates the expected belief for four different cases, two where the independence assumption is correct and two where it is incorrect. If we examine the two lines where $P(D) = 100\%$, the amount of error involved in assuming that $P(D) = 0\%$ is roughly the difference between them. We compute the actual error as a percentage change in the belief:

$$Error(A) = \left| \frac{P(W|A) - P(W|A, P(D) = 0)}{P(W|A)} \right|$$

As before, we can compute the **expected error** by doing a weighted average, $\sum_A P(A) \cdot Error(A)$. Figures 6(b) and 6(c) show the expected percent error for two different shaped grid models. What we see is that even when the dependency between the nodes is fairly small, 25%, we see an expected error in our belief of 10%. As the dependency grows, the percent error can become huge.

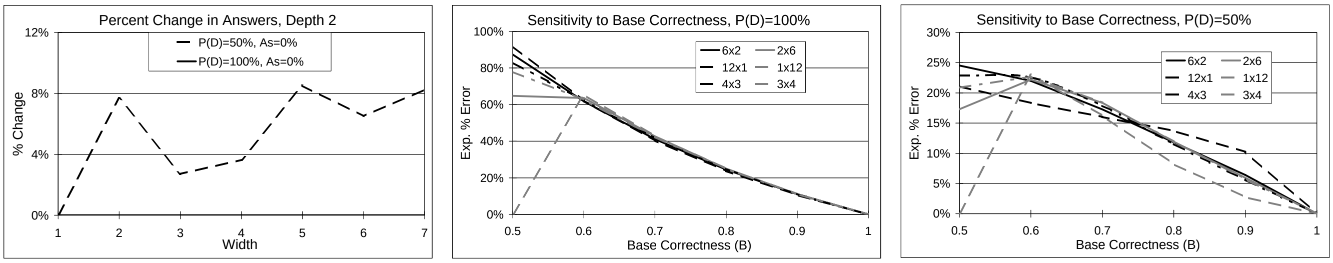
For those applications in which the confidence in the answer is less important than the actual answer, we show in Figure 7(a) the percentage of the answers that are different when assuming independence versus dependence. This does not necessarily mean that the answer is incorrect, just that if the level of dependence had been correctly modeled then the best answer would have been different. In 2xn systems this is trending towards 8% of the time.

We next explore the sensitivity of our family of systems to significant changes in accuracy of contributions. As one would expect, as a system becomes more accurate, the room for independence-assumption error decreases—even when there is high dependency. The flip side is also true. If the



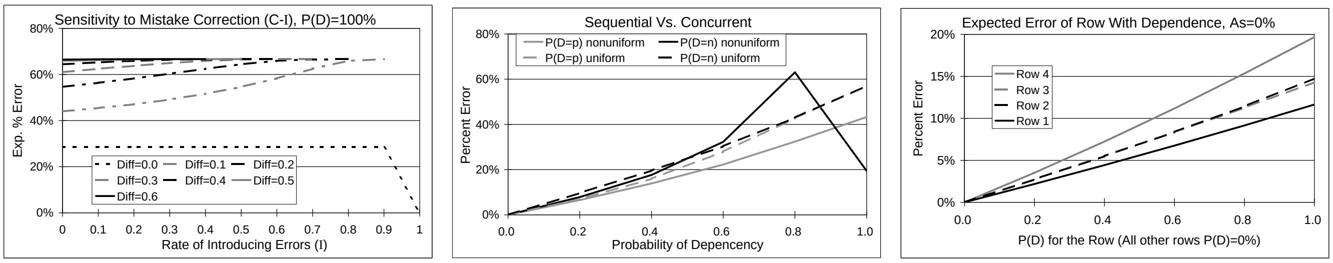
(a) (b) (c)

Figure 6: Expected belief and percent error for grid models of different width and depth. $P(D)$ is the actual dependence between the nodes, while As is the assumption made about the level of independence. $P(D) = 100\%$ means that the nodes are completely dependent, and an $As = 0\%$ means that agents are assuming that $P(D) = 0\%$. (a) The expected belief that the answer the system gives is correct, which is also the expected confidence. (b) The expected percent error as depth increases. (c) The expected percent error as width increases. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.



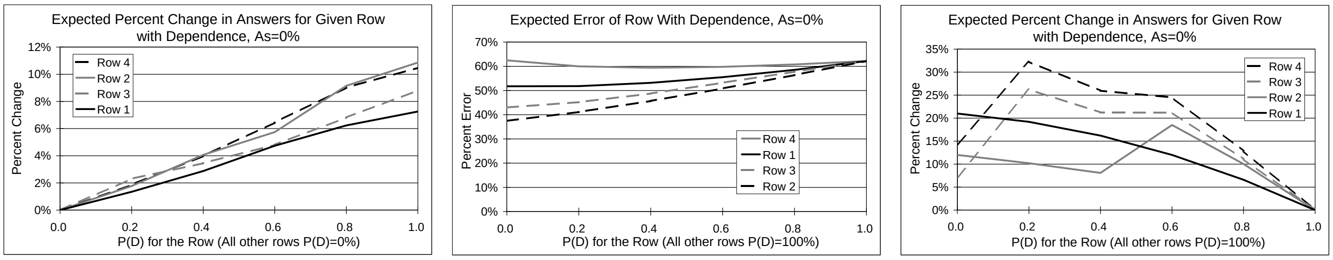
(a) (b) (c)

Figure 7: Effect of independence assumption on system answer and expected error sensitivity to agent accuracy. (a) The percentage of the time that the answer given when assuming independence differs from the answer given if the level of dependence is correctly modeled. When the dependence is 100%, the answer never changes because the nodes can only either be all $true$ or all $false$. (b) The sensitivity of the expected error to the base accuracy (B) of the nodes, when $P(D) = 100\%$. Each line represents a graph with the same number of nodes, but a different shape. The shape only has an effect when the accuracy of the nodes is near random ($B = 50\%$). (c) The sensitivity of the expected error to the base accuracy (B) of the nodes, when $P(D) = 50\%$. In this case the shape of the grid has a small effect on the expected error, but the general trend of each line is the same. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.



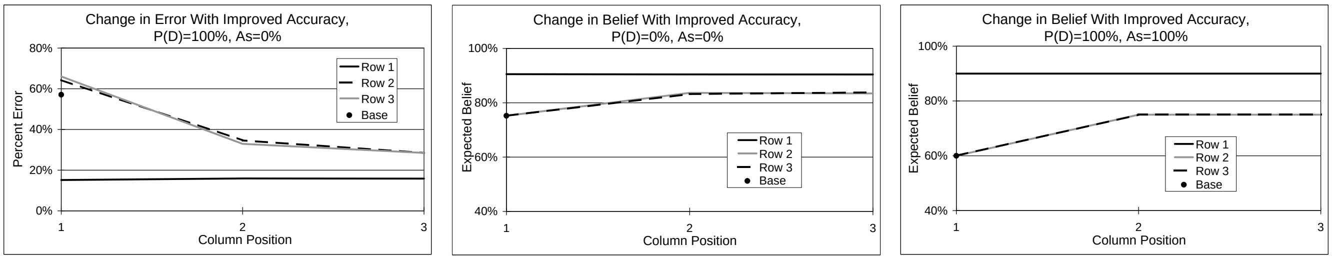
(a) (b) (c)

Figure 8: The effect of error-correction rates, concurrent and sequential dependencies, and row independence. (a) The sensitivity of the expected error to the net rate at which mistakes are corrected, $C - I$. The larger the difference between the rate at which errors are corrected and the rate at which new errors are introduced the less effect the rate of error introduction has on the expected error. In general the effect ranges from high expected error, to higher expected error. (b) This graph compares the difference between the sequential and concurrent dependencies in a 3x3 system. The lines labeled $P(D = p)$ have the probability that the nodes are dependent on its parents (sequential dependence) range from 0 to 1, while the probability that the nodes are dependent on its neighbors (concurrent dependence) is always 0. The lines labeled $P(D = n)$ is the opposite. The lines that are ***uniform*** mean that the base accuracy of the nodes B is the same for all of the nodes, $B = 0.6$, while for the ***nonuniform*** lines the base accuracy of the nodes is 0.6 for the first column, 0.65 for the second, and 0.7 for the third. (c) The expected error when all of the nodes are independent except for one row. The system is most sensitive to having dependency in the last row since there is not any processing after that to correct the mistakes made. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.



(a) (b) (c)

Figure 9: The effect of single-row dependencies. (a) The expected percentage of answers that change when there is dependence only in the given row. (b) The opposite of Figure 8(c) in that each row is completely dependent except for the given row. (c) The expected percentage of answers that change when there is complete dependence in all rows but the given row. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.



(a) (b) (c)

Figure 10: Effect of single-node accuracy. These figures were obtained by taking a 3x3 grid system and changing the base accuracy B from 0.6 to 0.9 for one node only. The node is specified by the row given by the line and the column given by the x-axis. The dot labeled ***base*** is the value obtained when no node's accuracy is increased. (a) is the expected percent error. (b) The change in expected belief for increasing the accuracy of one node when the nodes are independent and agents (correctly) assume that they are independent. (c) This is the change in expected belief when the nodes are completely dependent and agents (correctly) assume that they are dependent. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.

system's decisions are mostly random, then even when there is strong dependence it cannot get much worse by assuming independence. Figures 7(b) and 7(c) illustrate this by varying the base accuracy parameter from random, $B = 0.5$, to perfect, $B = 1.0$.

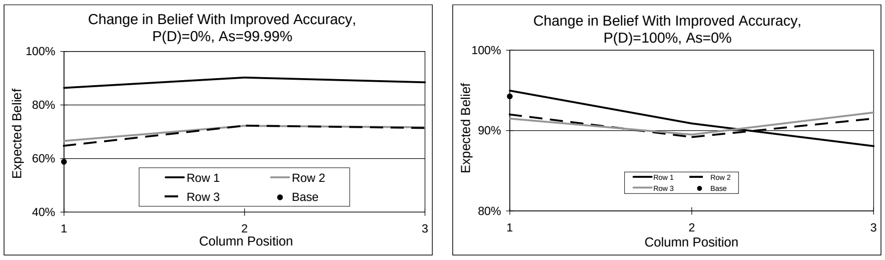
Figure 8(a) shows that the percent error is not very sensitive to the rate that mistakes are corrected versus introduced in sequential processing. Figure 8(b) shows the difference effect on percent error between sequential and concurrent dependencies, and Figure 8(c) shows the effect on percent error when all nodes are independent except for one selected row in the grid. As one might expect, later rows in sequential processing had a greater effect on percent error.

Figure 9(a) explores the effect on changed answers when all nodes are independent except for one selected row in the grid. Figure 9(c) shows the opposite case, where only the given row is independent. The percent error in this case is shown in Figure 9(b) (the opposite case of the percent error shown in Figure 8(c)).

The next figures explore the effect of an accuracy change to one node (agent) in the system. Figure10(a) shows the resulting change to percent error. The error when all of the nodes have an accuracy of 0.6 is 57%. Increasing the accuracy of nodes 2,1 and 3,1 in column 1 of our model actually increases the percent error value slightly. This is an edge effect of our modeling simplifications stemming from the fact that, with one parent, those two nodes in the model do not include the base correctness B in their CPTs. A more accurate measure for percent-error effects stemming from column one can be obtained by using a second model in which columns are exchanged. Figure 10(b) shows the change in expected belief for increasing the accuracy of one node when the nodes are independent and agents (correctly) assume that they are independent, while Figure 10(c) shows the change in expected belief when the nodes are completely dependent and agents (correctly) assume that they are dependent. Figures 11(a) and 11(b) shows the effect of one node's accuracy on the change in expected belief under incorrect dependency assumptions. Figures 12(a) and 12(b) show the change in system answer.

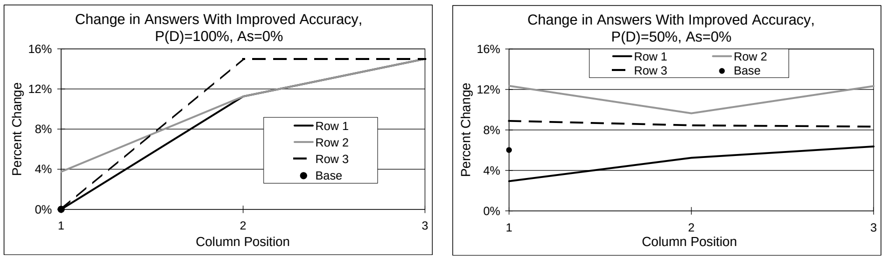
4. CONCLUSION

In this paper we demonstrated that a domain-independent model can be used to analyze the effect and propagation of potentially incorrect confidence-integration assumptions in a collaborating MAS application. We developed representations for both concurrent and sequential contribution processing and explored the implications of unaccounted-for dependencies in a hypothetical family of applications (modeled as rectangular grids) operating under a range of accuracy and dependency conditions. We showed that, although incorrectly assuming contribution independence is tolerable when agents are close to perfect, incorrect independence assumptions can be significant in systems involving mediocre agents. This can result in inaccurate result confidence values and changed system answers.



(a) (b)

Figure 11: Single-node accuracy and change in expected belief due to incorrect dependency assumptions. (a) is the change when the nodes are independent and we incorrectly assume that they are dependent. (b) is the change when the nodes are dependent and we incorrectly assume that they are independent. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.



(a) (b)

Figure 12: Single-node accuracy and change of answer due to incorrect dependency assumptions. In (a) the nodes are completely dependent and agents assume they are independent. In (b) the nodes are 50% dependent and agents assume that they are independent. The parameters used for these charts are $W = 0.5$, $D^p = 0.5$, $B = 0.6$, $I^1 = 0.2$, $I^2 = 0.1$, $C^1 = 0.4$, $C^2 = 0.3$.

Our modeling approach can be used to help in the design and improvement of MASs. If contributions are redundant, work by agents to develop them can be eliminated (unless the redundancy is desired due to possible agent failures). If additional data or more capabable processing can improve the accuracy of agent decisions, our model can identify the agents where decision-accuracy improvement efforts will produce the most overall system benefit.

An important future direction is to design a domain-independent mechanism that can be used to efficiently represent the dependence among contributions so that an independence assumption is unnecessary. The challenge is two-fold. First, the number of potential dependencies to capture grows exponentially with the number of entities in the system. Second, multi-agent and blackboard systems are typically open, and entities can come and go. While it is reasonable to measure or estimate the accuracy of decisions made by individual entities, assuming we know all of the dependencies among them is unrealistic. Entities may be added later while the system is already running, and some of them may not even exist when the application is developed. One possible approach is to characterize the error probabilities under specific conditions for each entity and to use these characterizations to estimate tighter bounds on the situational dependence/independence between entity contributions.

5. REFERENCES

[1] S. Bikhchandani, D. Hirshleifer, and I. Welch. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of Political Economy*, 100(5):992–1026, Oct. 1992.

[2] J. M. Bradshaw. *Software Agents*. AAAI Press & MIT Press, 1997.

[3] D. D. Corkill. Blackboard systems. *AI Expert*, 6(9):40–47, Sept. 1991.

[4] D. D. Corkill. Collaborating software: Blackboard and multi-agent systems & the future. In *Proceedings of the International Lisp Conference*, New York, New York, Oct. 2003.

[5] R. S. Engelmore and A. Morgan, editors. *Blackboard Systems*. Addison-Wesley, 1988.

[6] J. Ferber. *Multi-Agent Systems: Toward a collective intelligence*. Addison-Wesley, 1998.

[7] B. J. Grosz. Collaborative systems. *AI Magazine*, 17(2):67–85, Summer 1996.

[8] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Journal of Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, Mar. 2006.

[9] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[10] J. Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2000.

[11] A. Pfeffer. Sufficiency, separability and temporal probabilistic models. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence*, pages 421–428, Seattle, Washington, Aug. 2001.

[12] J. Surowiecki. *The Wisdom of Crowds*. Doubleday, 2004.