Knowledge Fusion in the Large – taking a cue from the brain

Lokendra Shastri and Carter Wendelken
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704

Abstract *Even the most commonplace cognitive behaviors such as vision and language understanding involve large-scale fusion of disparate pieces of evidence. Therefore, our capacity to rapidly and effortlessly produce coherent interpretations of visual and verbal inputs points to the remarkable ability of the human mind/brain to fuse evidence. We discuss a neurally motivated computational model that attempts to replicate some of this remarkable ability, and illustrate the functioning of the model with the help of a few examples.*

Keywords: Neural Networks, Inference, Evidence combination, Coherence

1 Introduction

Even commonplace cognitive behaviors such as vision and language understanding involve large-scale fusion of evidence from disparate sources. Consider the task of understanding language wherein squiggles on a surface or fluctuations in air-pressure are mapped by the reader or hearer into coherent mental descriptions of events and situations. The process underlying this mapping appears to be remarkably complex. It involves, among other things, recognizing words, disambiguating word senses, incorporating grammatical constraints, and carrying out inferences based on fuzzy and partial knowledge to establish causal and referential coherence.[1]

Any system that attempts to explain our ability to establish causal and referential coherence during language understanding must possess a number of properties. First, such a system must be representationally adequate. It must be capable of encoding specific facts as well as general regularities (`aka` rules) that capture the causal structure of the environment. In particular, the system should be capable of encoding context-dependent and evidential cause-effect relationships. Second, the system should be inferentially adequate, that is, it should be capable of drawing a wide range of explanatory and predictive inferences. In doing so, the system should be able to *combine evidence* provided by disparate sources and arriving at *coherent* and mutually reinforcing interpretations. Third, the system should be capable of establishing referential coherence. In particular, it should be able to posit the existence of entities that may be only implicit in the "input" ("John bought a book" implies the existence of an entity that sold the book to John) and unify (or merge) entities by recognizing that two entities referred to in a discourse may be one and the same. Fourth, the system should be capable of learning and fine-tuning its causal model based on experience, instruction, and exploration. Finally, the system should be scalable and computationally effective.

In this paper we describe a neurally motivated system that exhibits — at least to a certain extent — the properties enumerated above. This system is an extension of SHRUTI [11]. It can express causal knowledge involving n-place relations, limited quantification, and type restrictions. It can encode specific events as well as context-sensitive priors over events. It expresses dynamic bindings via the synchronous firing of appropriate node clusters and performs inferences via the propagation of rhythmic activity over node clusters. This propagation amounts to a parallel breadth first activation of the underlying causal graph, and hence, the reasoning in SHRUTI is extremely fast. The use of weighted links and activation combination functions at nodes allow SHRUTI to encode soft rules and perform evidential inference. SHRUTI supports supervised learning which allows it to fine-tune its causal model in a data-driven manner. Moreover, SHRUTI supports short-term associative learning which allows it to dynamically favor stable coalitions of activity. The latter plays a critical role in establishing coherence. In this paper we focus on the ability of SHRUTI to (i) rapidly establish causal and referential coherence and (ii) combine evidence in a flexible and context-dependent manner using a family

---

[1] Causal coherence refers to the establishment of causal relationships among various events mentioned in a discourse. Referential coherence involves keeping track of entities referenced in a discourse and determining which entities are the same. It is well known that inferences required to establish causal and referential coherence occur rapidly and automatically during text understanding (see e.g., [6, 7, 5]). The evidence for the automatic occurrence of predictive inferences is mixed, but their occurrence cannot be ruled out [9].

of evidence combination functions. Other details may be found in [11, 10].

2 Representational Overview

To motivate and concretize the description of SHRUTI's behavior consider the following narrative: "(S1) John fell in the hallway. (S2) Tom had cleaned it. (S3) He got hurt." Upon being presented with the above narrative (we will see how below) SHRUTI rapidly infers the following: (i) Tom had cleaned the hallway, (ii) The hallway floor was wet, (iii) John was walking in the hallway, (iv) John slipped and fell because the floor was wet, and (v) John got hurt because he fell. Notice that SHRUTI draws several inferences required to establish referential and causal coherence. It explains John's fall by making the plausible inference that John was walking in the hallway and he slipped because the floor was wet. It also infers that John got hurt because of the fall. Moreover, it determines that "it" in (S2) refers to the hallway, and that "He" in (S3) refers to John, and not to Tom. SHRUTI draws these inferences based on commonsense knowledge such as that shown in Figure 1, as well as several additional commonsense rules and facts about cleaning, wet floors, and being hurt.

2.1 Interplay of structure and dynamics

A description of SHRUTI requires the specification of its *structure* as well as a description of its *dynamic* behavior. All long-term (persistent) knowledge is encoded in SHRUTI via structured networks of nodes and links. The dynamic aspects of SHRUTI involve the encoding and propagation of dynamic bindings via synchronous activity, the activation of long-term facts in response to dynamic bindings, evidence combination, the dynamic instantiation and unification of entities, the short-term increase (potentiation) of weights due to convergent activity, and the emergence of coherence in the form of reverberant activity along closed loops.

2.1.1 Encoding Relations Using Focal Clusters

Each relation is represented by a focal cluster depicted by a dotted ellipse in Figure 1. Consider the focal cluster for *slip*. This cluster includes an enabler node labeled *?:slip*, two collector nodes labeled *+:slip* and *-:slip*, and two role nodes labeled *slip-pat* and *slip-loc* for its two roles *patient* and *location*. In general, the cluster for an *n*-place relation will contain *n* role nodes, with the synchronized activity of each indicating a particular role binding (as described below).

Activity in *?:slip* indicates the strength with which information about the particular instance of the *slip* relation is sought. The activation levels of the collectors *+:slip* and *-:slip* encode a graded belief ranging continuously from *no* on the one extreme (only *-:slip* is active), to *yes* on the other (only *+:slip* is active), and *don't know* in between (neither collector is very active). If both the collectors receive comparable and strong activation then both collectors can be active, despite mutual inhibition. This signals a contradiction.

Links from the collector nodes to the enabler node of a relation convert a dynamic assertion of a relational instance into a query about the assertion. Thus the system continually seeks an explanation for active assertions. The weight on the link from *+:slip* (*-:slip*) to *?:slip* is proportional to the system's propensity for seeking explanations and inversely proportional to the probability of occurrence of a positive (negative) instance of $slip$.

Nodes are computational abstractions and correspond to *small ensembles of cells*, and a connection between nodes corresponds to several connections from cells in one ensemble to cells in the other. Phasic nodes, of which role nodes are an example, respond only to synchronous activity and fire only in synchrony with their inputs. Enabler and collector nodes, however, can integrate activity over a broader time window (see [10]).

The *dynamic* encoding of a relational instance corresponds to a *rhythmic* pattern of activity wherein bindings between roles and entities are represented by the *synchronous* firing of appropriate role and entity nodes [12, 11] With reference to Figure 1, the dynamic representation of the relational instance *(fall: ⟨fall-pat=John⟩, ⟨fall-loc=Hallway⟩)* (i.e., "John fell in the Hallway") will involve the synchronous firing of *+:John* and *fall-pat*, and the synchronous firing of *+:Hallway* and *fall-loc*. The entities *+:John* and *+:Hallway* will fire in distinct phases.

SHRUTI encodes two types of facts in its long-term memory: episodic facts (E-Facts) and taxon facts (T-facts). These facts provide closure between the enabler node and the collector nodes. While an E-fact corresponds to a specific instance of a relation, a T-fact corresponds to a distillation or statistical summary of various instances of a relation and can be viewed as coding *prior probabilities*. T-facts can be conditioned on the type of role-fillers (e.g., the T-fact *buy(Person,Car)* encodes how likely it is that a person would buy a car).

2.1.2 Encoding of Types and Instances

This is illustrated at the right of Figure 1. The focal cluster of each entity, *A* consists of a *?:A* and a *+:A* node. In contrast, the focal cluster of each type, *T* consists of a pair of *?* (*?e:T* and *?v:T*) and a pair of + nodes (*+e:T* and *+v:T*). The pair of *v* nodes and the pair of *e* nodes signify universal and existential quantification,

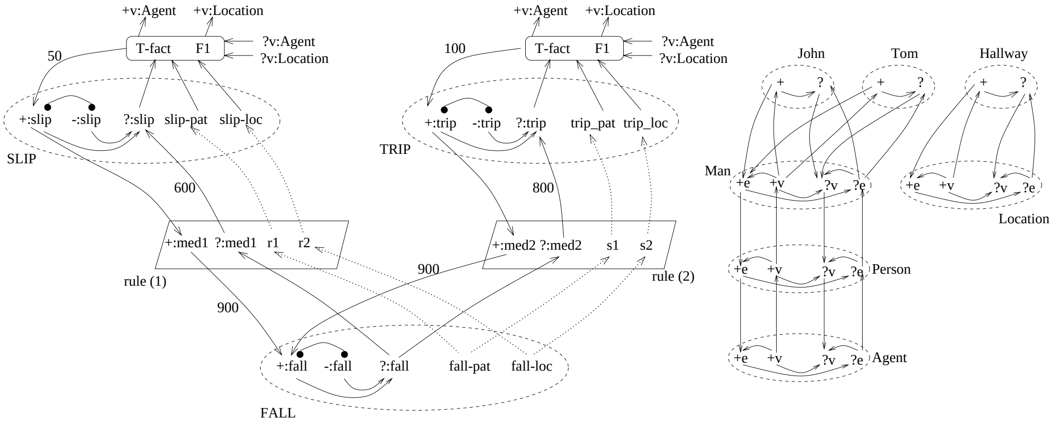

Figure 1: An example SHRUTI network encoding a subset of the knowledge base for the “John fell” example.

respectively. The activation *levels* of *?:A, ?v:T, and ?e:T* nodes signify the strength with which information about entity *A*, the type *T*, and an instance of type *T*, respectively, is being sought. Similarly, the activation *levels* of *+:A*, *+v:T*, and *+e:T* signify the degree of belief that *A*, *T*, and an instance of type *T*, respectively, play appropriate roles in the current situation.

2.1.3 Encoding of Rules

A rule is encoded via a mediator focal cluster (shown as a parallelogram) that mediates the flow of activity between the antecedent and the consequent clusters. The mediator consists of a collector and an enabler node and as many role-instantiation nodes as there are distinct variables in the rule. The enablers of the consequent relations are connected to the enablers of the antecedent relations via the enabler of the mediator. The appropriate (+/–) collectors of the antecedent relations are linked to the appropriate (+/–) collectors of the consequent relations via the collector of the mediator. Each of these enabler and collector links for a rule has a weight which can be specified by a knowledge engineer and/or learned via supervised learning. The roles of the consequent relations are linked to the roles of the antecedent relations via appropriate role-instantiation nodes in the mediator. This linking reflects the correspondence between antecedent and consequent roles specified by the rule.

If a role-instantiation node receives activation from one or more consequent role nodes, it simply propagates the activity onward to the connected antecedent role nodes. If on the other hand, it receives activity only from the mediator enabler, it sends activity to the *?:e* node of the type specified in the rule as the type restriction for this role. This causes the *?:e* node of this type to become active in an unoccupied phase. The *?:e* node of the type conveys activity in this phase to the role-instantiation node which in turn propagates this activity to connected antecedent role nodes. This interaction between the mediator and the type hierarchy, in effect, creates activity corresponding to “Does there exist some role filler of the specified type?”

2.1.4 Mutual Exclusion and Collapsing of Phases

Entities in the type hierarchy can be part of a *phase-level* mutual exclusion cluster (ρ-mex cluster). The + node of every entity in a ρ-mex cluster has inhibitory links to and from the + node of all other entities in the cluster. As a result of the mutual inhibition, only the most active entity within a ρ-mex cluster can remain active in any given phase. A similar ρ-mex cluster can be formed by mutually exclusive types. Mutual exclusion also occurs in the type hierarchy as a result of inhibitory connections from the + nodes of a type (or an entity) to the ? nodes of all its siblings. This inhibition leads to an "explaining away" phenomenon. If for example, the type query “Is it a Person?” (i.e., activation of *?e:person*) leads to the queries “Is it a Man?” and “Is it a Woman?”, then strong support received by *+e:Woman* reduces the strength of the query *?e:Man*. In essence, the query “Is it a Man?” is no longer considered important by the system since it was seeking a person and it has already found a woman.

SHRUTI allows separate phases to coalesce into a single phase, or new phases to emerge, as a result of inference. The latter is realized by the allocation of new phases resulting from the interaction between role-instantiation nodes in mediators and the type hierarchy. The unification of phases is realized in the current implementation by collapsing of phases based on activity within an entity cluster or within a focal cluster. In the first case, phase collapsing occurs whenever a single entity dominates multiple phases (for example if the same entity comes to be the answer to multiple queries). In the second case, phase collapse occurs if two unifiable instantiations of a relation arise within a focal cluster. For example, an assertion *+:fall(John,Hallway)* alongside the query $\exists$ *x:man ?:fall(x,Hallway)* (Did a man fall in the Hallway) will result in a merging of the two phases for “a man” and “John”.

> SHRUTI’s ability to flexibly instantiate entities and collapse them into a single entity during inference is due to its use of temporal synchrony to represent dynamic bindings.

2.1.5 Short-term Potentiation

If *?:P*, the enabler node of P receives activity from a mediator enabler node and concurrent activity from one of P’s collector nodes, then the weight of the link from the mediator enabler to *?:P* increases for a short-duration.[2] With reference to the “John fell” example, this increase in weight has the following functional significance (refer to Figure 1): The activation arriving at *?:slip* from *?:med1* means that “John slipped” is being sought as a possible explanation of “John fell”. The concurrent arrival of activity from *+:slip* would mean that at this very time “John slipped” is also being asserted. Under these circumstances, it is highly likely that “John slipped” may indeed be the explanation of “John fell”. The increase in weight of the link from *?:med1* to *?:slip* marks *slip* as a more likely explanation for *fall* under the existing circumstances.

If *+:P* (*-:P*) receives activity from one of its T-facts and concurrent activity from a mediator collector node, then the weights of the links from the mediator collector to *+:P* (*-:P*) and from the active T-facts to *+:P* (*-:P*) increase for a short-duration. With reference to the “John fell” example, this increase in weights has the following functional significance (refer to Figure 1): The activation arriving from *+:med1* at *+:fall* means that “John fell” is being predicted as a possible consequence of “John slipped”. The concurrent arrival of activity at *+:fall* from *?:fall* (via a T-fact) would mean that at this very time “John fell” is also being sought as a possible explanation of some event (this is why *?:fall* is active). Under these circumstances, it is highly likely that the event “John fell” actually occurred and is an effect of “John slipped”. The increase in weight of the link from *?:med1* to *+:fall* marks *fall* as a more likely effect of *slip* under the existing circumstances.

2.1.6 “Explaining away” in the Causal Model

A “explaining away” phenomena also occurs in the causal model as a result of inhibitory connections between rules which share the same consequent. For the structure shown in Figure 1, there is for example an inhibitory link (not shown) from *+:med1* to the link from *?:fall* to *?:med2*. As a result, a strong activation of *+:slip* reduces the activation flowing from *?:fall* into *?:trip*. In essence, if the system is seeking an explanation for fall, then a strong belief in slipping is taken to be a sufficient explanation of falling, and hence, the search for tripping acquires lesser significance.[3]

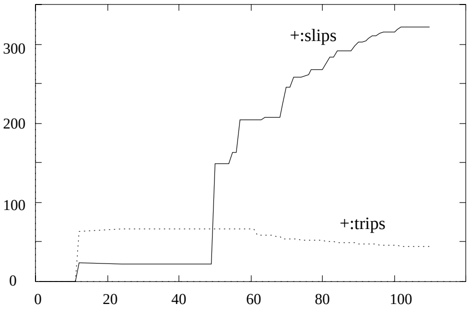

Figure 2: The activation trace of collector nodes *+:slip* and *+:trip* during the processing of the “John fell” story. X-axis records the number of cycles. Each cycle may correspond to ∼50-100 msecs.

> Taken together, the short-term associative increase in weights and the inhibitory interactions leading to the explaining away phenomena, provide a powerful and neurally plausible mechanism that enable SHRUTI to prefer coherent explanations over non-coherent ones.

3 Simulation Result

The activation trace resulting from the processing of the "John fell" story is shown in Figures 2 and 3. Figure 2 shows the actual activation levels of the *+:slip* and *+:trip* nodes as the story is processed by SHRUTI. Figure 3 depicts the activation trace of a larger *subset* of nodes. The depiction in this figure, however, has been simplified to highlight key aspects of the network behavior. In particular, several nodes have been omitted, some intermediate cycles have been omitted and the activation levels of collector and enabler nodes have been discretized to four levels. Please note that due to simplifications made to Figure 3, the time scales along the x-axis in Figures 2 and 3 are not the same. To minimize confusion, we will refer to the times in Figure 2 as cycles and in Figure 3 as steps.

A sentence is conveyed to SHRUTI by activating the

---

[2] This is modeled after the biological phenomena of short-term potentiation (STP) [2].

[3] The use of inhibitory connections for explaining away is motivated in part by [1].

+ node of the appropriate relation and establishing role-entity bindings by the synchronous activation of the appropriate role and entity nodes. The sentences are presented in sequence and after each sentence presentation, the network is allowed to propagate activity for a fixed number of cycles. For example, the first sentence (S1) is communicated to SHRUTI in step 1 (cycle 0) by activating the node *+:fall*, the nodes *fall-pat* and *+:John* in synchrony, and the nodes *fall-loc* and *+:Hallway* in synchrony. The firing of nodes *+John* and *+:Hallway* occupy distinct phases — ρ_1 and ρ_2, respectively.

Activation from the focal cluster for *fall* reaches the mediator structures of rules (1) and (2) shown in Figure 1. Consequently, nodes *r1* and *r2* in the mediator for rule (1) become active in phases ρ_1 and ρ_2, respectively. Similarly, nodes *s1* and *s2* in the mediator of rule (2) become active in phases ρ_1 and ρ_2, respectively. At the same time, the activation from *+:fall* activates *?:fall* which in turn activates the enablers *?:med1* and *?:med2* (the activity of mediator nodes, and role nodes of *slip* and *trip* is not depicted in Figure 3). The activation from nodes *r1* and *r2* reaches the roles *slip-pat* and *slip-loc* in the *slip* focal cluster, respectively. Activation also reaches *trip-pat* and *trip-loc*. In essence, the system has created new bindings for the *slip* and *trip* relations. These bindings together with the activation of the nodes *?:slip* and *?:trip* encode two queries: "Did John slip in the hallway?", and "Did John trip in the hallway?". At the same time, activation travels in the type hierarchy and activates the nodes *?v:Man*, then *?v:Person*, and then *?v:Agent* in phase ρ_1, and the *?v:Location* node in phase ρ_2. The coincident activity of *slip-pat* and *?v:Agent* node, and the coincident activity of the *slip-loc* and *?v:Location* nodes leads to the firing of the T-fact F1 associated with slip. The activation of F1 causes activation from *?:slip* to flow to *+:slip*. The T-fact F2 associated with trip also becomes active in an analogous manner and conveys activation from *?:trip* to *+:trip*. The level of these activations is a measure of the prior probabilities that a person may slip or trip. At this time, "John tripped" is believed to be a more likely explanation of "John fell" than "John slipped."

While the activation spreads "backwards" from the *fall* focal cluster as described above, activation also travels "forwards" to the *hurt* focal-cluster (not shown in Figure 1) and leads to the prediction that John got hurt.

The introduction of sentence S2 in step 6 (Figure 3) (cycle 40 Figure 2) results in the instantiation of *clean* with the bindings *(⟨clean-agt=+:Tom⟩, and ⟨clean-loc=+e:Location⟩)*. As a result, Tom gets active in phase ρ_3 and *+e:Location* in phase ρ_4. Note that now we have two instantiations of a location. The second instantiation gets merged with the first (Hallway) as a result of phase merging described in Section 2.1.4. This happens in step 8 (see activity of *+e:location* in Figure 3). At this time, *+:wetFloor* also becomes active as a result of activity arriving from *+:clean* via the mediator of the rule "cleaning leads to a wet floor" (not shown in Figure 1). By step 10 (Figure 3) *+:slip* becomes more active as a result of the high activation of *+:wetFloor*. The effect of "explaining away" kicks in and causes the activation of *+:trip* to go down by step 12. The strength of *+:slip* increases even further due to (i) the potentiation of links from the mediator for the rule "walking on a wet floor may cause slipping" (not shown in Figure 1), (ii) the potentiation of the link from *?:med1* to *?:slip*, and (iii) the effect of explaining away. The effect of these changes on the activation levels of *+:slip* and *+:trip* may be seen more vividly in the detailed trace shown in Figure 2.[4]

S3 is introduced in step 14 (cycle 80) with the binding *(⟨ hurt-pat=+e:Man⟩)*. This leads to *+e:Man* becoming active in phase ρ_4 and a second dynamic instantiation of *hurt* (in addition to the earlier instantiation resulting from the inference *hurt(John)*). These two instantiations get merged immediately, and phase ρ_4 gets merged with ρ_1 (John) in step 15 as a result of the phase merging described in Section 2.1.

4 Evidence combination

The problem of evidence combination arises even in the limited example discussed above. This probolem, however, can become far more complex in real-world situations. It becomes apparent as the system is used to model increasingly complicated domains that there is a need for a significant degree of flexibility in the manner in which evidential values are combined.

There are many places in SHRUTI where activity converging on a node from different sources must be combined to determine an output value for the node. The combination of collector activity from multiple antecedents and also across multiple rules, and of enabler activity from multiple consequents and multiple rules, are prime examples of this. At the locations where evidence from facts is incorporated, in the influence of collector activity on an enabler node, and in propagation of activity through the type hierarchy, as well as in a number of other situations, evidence combination is also being performed.

Evidence combination in SHRUTI takes the form of a set of evidence combination functions, or ECFs. At each point in the network where evidence must be combined, a particular ECF is chosen. In selecting the range

---

[4] If sentence S2 were delayed, the activity in *slip* would lead to the instantiation of an instance of *clean* with an entity of type *agent* being instantiated as a potential filler of the role *clean-agt*. This entity, however, would get unified with *Tom* upon the introduction of S2.

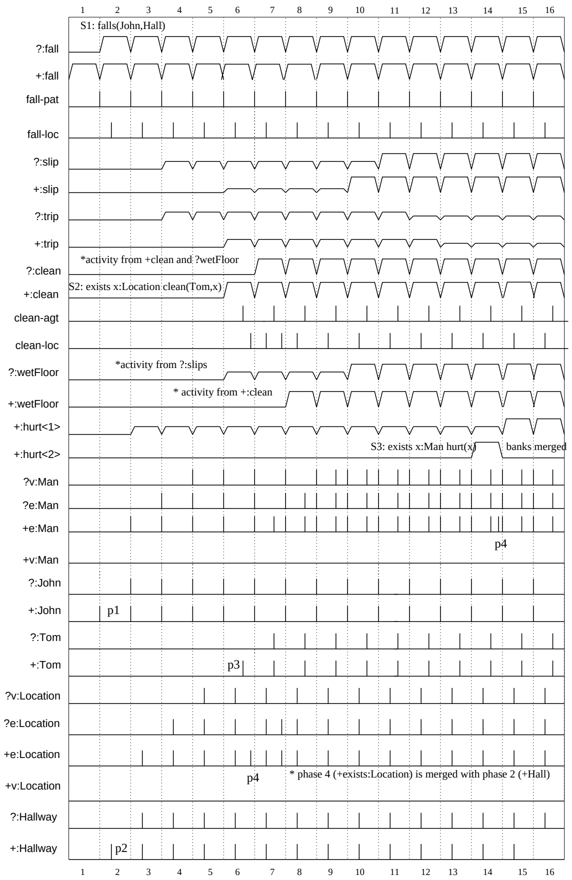

Figure 3: Schematized activation trace of selected nodes.

of functions, the goal was to have a set large enough to adequately model real-world data, but small enough such that the choice between functions for a particular situation is relatively simple. Moreover, these functions should be computationally simple, or at least decomposable into very simple parts, such that the biological plausibility of the system is not sacrificed. The set of functions developed is not intended to cover all possible relations, but instead to be sufficiently flexible so as to capture the vast majority of practical situations.

4.1 Background

An obvious source of inspiration for this undertaking is fuzzy logic, where a multitude of functions have been developed to combine fuzzy membership values in different ways [13]. Of particular note are the classes of binary operators known as T-norms and S-norms. These represent, respectively, general forms of fuzzy set intersection and union. The extension of T-norm and S-norm operators to handle combination of multiple evidence values is generally straightforward, and these two categories are prime candidates for evidence combination functions in SHRUTI.

In neural networks, greater representational complexity is achieved by adding more and more nodes and interconnections, keeping the combination function at the nodes very simple. The commonest form of evidence combination in this context is the sigmoid-sum; this function has a number of properties which make it appealing for use in neural nets. Although other functions can certainly be used as well, it would not make sense with a standard neural net to pick and choose particular functions for particular nodes, since the nodes have no special meaning. In a structured connectionist system like SHRUTI, however, nodes are meaningful and the network structure is relatively fixed, so it is useful to push more flexibility into the combination functions than is either necessary or possible in standard neural nets. Belief nets also utilize a form of evidence combination, found in the conditional probability tables associated with each node. In the case of a full CPT, the flexibility of combination is high but so are the storage and computational demands. The often used noisy-OR function [8] reduces these demands but when used universally, as is often the case, is overly restrictive. Other means of reducing complexity of the CPT, such as encoding it with a tree structure, demonstrate a different approach to evidence combination than that envisioned here [4].

4.2 Combination Functions

The combination functions developed for SHRUTI can be thought of as forming a continuum, with *and* at one end and *or* at the other. In between these two extremes are four basic categories of functions: *soft-and* (with values up to min), *soft-min* (ranging from min to average), *soft-max* (ranging from average to max), and *soft-or* (ranging from max and up). Although specific functions have been chosen to represent each of these categories, many of the functions developed for fuzzy logic could be used here. As a general rule, antecedents or consequents which are correlated will be combined into a single multiple-antecedent or multiple-consequent rule in SHRUTI, whereas uncorrelated factors will reside in separate rules. This means that for the former case, evidence combination functions should allow for this correlation, while in the latter assumptions of independence are usually justified. It is proposed that most meaningful combinations of evidence can be characterized as belonging to one of these four basic categories, on the basis of the necessity or sufficiency and also degree of correlation of their inputs.

Link weights can play an important and context-dependent role in many of these functions. The standard use of link weights is to multiply them with the input values prior to doing evidence combination. However, instead of simply affecting values before they are combined, weights can also be used as additional function parameters, with different interpretation for different functions. The use of link weights in this manner, elaborated for each of the ECFs below, provides a significant degree of flexibility in the kinds of relations that can be represented. While this use of link weights appears to run counter to biological intuitions, it is possible to replace each of the so-weighted combination functions with an expanded network which involves only very simple combinations and which employs link weights in a more standard manner.

4.2.1 Soft-And and Soft-Or

At one end of the spectrum of functions are the *and*-like functions, corresponding to the T-norms of fuzzy logic, which are appropriate for combining evidential values which are deemed necessary. The basic *weighted and* is calculated as $\prod_i (1-(1-X_i)W_i)$ (where X_i is the evidential value and W_i is the weight for the ith incoming link). This *and* function is most appropriate for combining independent sources of evidence, such as in the following example rule: ∀*w,x:Person, y:Object*[AND*(canSell(x,y) 1000,wants(w,y) 800)* ⇒ *sells(x,w,y) 500*] where both a potential seller's ability to sell and a potential buyer's desire to buy are necessary and independent prerequisites of a sale actually taking place. The collector node of the mediator for this rule, which combines activity from the antecedents *canSell* and *wants*, will utilize the *and* function. The weight of 500 specified for the consequent means this can only be concluded with half of the maximum possible strength. With the independence assumption relaxed, assuming instead that combined values are positively correlated, a *soft-and* function is appropriate which has a value greater than the product-based *and*. The function chosen for this purpose is the *and(X)/or(X)* function which is similar to the Hamacher product *T*-norm $H(x,y) = xy/(w+(1-w)(x+y-xy))$ (where w is a weight in [0,1]) generalized to n variables [3]. The utilization of link weights brings another dimension to the standard *and* function. Since the main characteristic of the and is that combined values are regarded as necessary, an obvious interpretation for the link weights is that they reflect the degree of necessity. In probabilistic terms, this would be the probability that the consequent is false given that the antecedent is false. This means that lower link weights on the *and* function generally result in higher output values. While this fact may be counterintuitive when considering the network behavior, assignment of degrees of necessity seems quite practical from a knowledge engineering standpoint, and makes the simple and function remarkably flexible. In the above example, the interpretation is that while *canSell* is absolutely necessary in order to draw any conclusion about a sale taking place, *wants* is not.

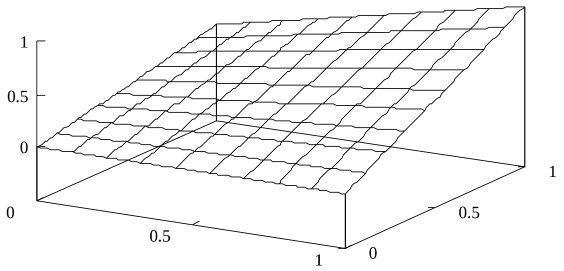

Figure 4: Graph of a weighted *and* function with antecedent weights of 1.0 and 0.6.

Shown above is a graph of a weighted *and* function with two antecedents of weights 1.0 and 0.6 (see Figure 2). The relative importance of the first value is seen in that the function value changes slowly while traveling along the near axis, but rapidly when traveling along the far axis.

At the other end of the spectrum is the weighted *or*, given as $(1-\prod_i (1-X_i * W_i))$. *Or*-like functions, which can be thought of as those having output values at least equal to the maximum input, are used when there is a notion of sufficiency of individual antecedents to affect the consequent. These correspond roughly to the S-norms of fuzzy logic, and many of these fuzzy operators might be adapted to the task. *Or* is the most commonly used function for combining activity from different rules that converge on a particular concept. As *or* assumes that antecedents are independent, a *soft-or* (the complement of the *soft-and*) is provided to handle cases where antecedents are correlated. This is in particular the function of choice for combining enabler activity from multiple consequents, which are most certainly correlated. The general requirement for *soft-or* is that its value be less than the *or* but still greater than the *max*. The natural interpretation of link weights for *or*-like functions is that they represent the degree of sufficiency of the source concept - the probability of the consequent given only the particular antecedent.

4.2.2 Weighted Averages: Soft-min and Soft-max

Covering the range between *min* and *max* are the weighted averages. Weighted averages are appropriate when individual antecedents are neither necessary nor sufficient. For all of these functions the link weights represent degrees of influence, giving the relative ef-

fect of an antecedent value on the output. There are two main functions in this class: the *soft-min* function $((\sum_i X_i^k W_i)/(\sum_i W_i))^{1/k}$ for $k \in (0,1)$, and the *soft-max* function with $k \in (1,\infty)$. It should be noted that *min* and *max* are the limits of the given *soft-min* and *soft-max* functions, as $k \rightarrow 0$ and $k \rightarrow \infty$, respectively, so this whole range from *min* to *max* is really only one functional form with a varying parameter. *Soft-min* is used when it is necessary that most of the evidence for the antecedents be available in order to conclude the consequent, but unlike *and* no single piece is required. Combining evidence about the symptoms associated with a particular syndrome is a place where *soft-min* can be appropriate. A syndrome is a specific set of co-occurring symptoms and so in deciding whether a particular syndrome is present, lack of evidence for one of the particular symptoms should weigh heavily against a positive conclusion. But it should still be possible to conclude that a syndrome is present even if evidence for one of its symptoms is absent, so any *and*-like function would not be quite appropriate and *soft-min* is the function of choice. With *soft-max*, only a fraction of the potential evidence is sufficient to lead to strong activity in the consequent, but unlike *or* no single piece is alone enough. The following rule provides a reasonable example usage of *soft-max*:
∀*x:Person*[SOFTMAX*(tall(x) 500, athletic(x) 800, practiceDaily(x,Basketball))* ⇒ *goodAt(x,Basketball)*]
Here each factor can contribute significantly to the result, but none is really sufficient to draw much of a conclusion absent some other support.

5 Conclusion

We have discussed how causal and referential coherence can arise within a neurally plausible system as a result of spontaneous activity in a network. The network's structure reflects the causal model of the environment and when the nodes in the network are activated to reflect a given state of affairs, the network spontaneously combines evidence, seeks coherent explanations, and makes likely predictions. The time taken to perform an inference is simply proportional to the depth of the causal derivation and is otherwise independent of the size of the causal model. The state of coherence is reflected as reverberation around *closed loops*. The reverberating pattern of rhythmic activity also codes dynamic bindings via synchronous activity. Coherence arises in SHRUTI as a result of (i) flexible evidence combination, (ii) inhibitory interactions among sibling entities, types and rules, (iii) short-term increase in link weights resulting from short-term potentiation, and (iv) the dynamic merging and instantiation of entities.

References

[1] Ajjanagadde, V. (1991) Abductive reasoning in connectionist networks. TR WSI 91-6, Wilhelm-Schickard Institute, University of Tubingen, Tubingen, Germany.

[2] Bliss, T.V.P. and Collingridge, G.L. (1993) A synaptic model of memory: long-term potentiation in the hippocampus. *Nature* 361, 31–39.

[3] Driankov, D. Helendoorn, H. and Reinfrank, M. 1996. *An Introduction to Fuzzy Control*. Springer-Verlag: Berlin.

[4] Friedman, N. and Goldszmidt, M. (1996) Learning Bayesian Networks with Local Structure. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence.*

[5] Keenan, J. M., Baillet, S. D., & Brown, P. (1984) The Effects of Causal Cohesion on Comprehension and Memory. *Jrnl of Verbal Learning and Verbal Beh.*, 23, 115-126.

[6] Kintsch, W. (1988) The Role of Knowledge Discourse Comprehension: A Construction-Integration Model. *Psych. Rev.*, Vol. 95, 163-182.

[7] McKoon, G., & Ratcliff, R. (1980) The Comprehension Processes and Memory Structures Involved in Anaphoric Reference. *Jrnl. of Verbal Learning and Verbal Beh.*, 19, 668-682.

[8] J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.

[9] Potts, G. R., Keenan, J. M., & Golding, J. M. (1988) Assessing the Occurrence of Elaborative Inferences: Lexical Decision versus Naming. *Journal of Memory and Language*, 27, 399-415.

[10] Shastri, L. (1999) Advances in SHRUTI — A neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony., *Applied Intelligence*. To appear.

[11] Shastri, L. & Ajjanagadde V. (1993) From simple associations to systematic reasoning. *Behavioral and Brain Sciences*, 16:3 p. 417-494.

[12] von der Malsburg, C. (1981) The correlation theory of brain function. Internal Report 81-2. Department of Neurobiology, Max-Planck Institute for Biophysical Chemistry, Gottingen, Germany.

[13] Zadeh, L. 1996. *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A Zadeh*, G. Klir & B. Yuan (eds), World Scientific: Singapore, River Edge, N.J.