# Combining belief and utility in a structured connectionist agent architecture

Carter Wendelken and Lokendra Shastri
International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704
{carterw,shastri}@icsi.berkeley.edu

### Abstract

The SHRUTI model demonstrates how a system of simple, neuron-like elements can encode a large body of *relational* causal knowledge and provide the basis for rapid inference. Here we show how a representation of utility can be integrated with the existing representation of belief, such that the resulting architecture can be used to reason about values and goals and thereby contribute to decision-making and planning.

## Introduction

To understand how the brain creates the mind, one could work mainly from the top down, characterizing mental processes, or from the bottom up, trying to understand the capabilities of neurons and simple circuits. In developing the SHRUTI model we have pursued both these approaches simultaneously in order to understand how networks of neurons can perform complex cognitive tasks. In past work, we have demonstrated how such networks can make predictive and explanatory inferences with respect to a large body of causal knowledge. In this paper, we show how the SHRUTI architecture can be extended to represent and reason not only about beliefs but also about utilities, values and goals. The resulting model uses a single causal structure to seek explanations, make predictions, and identify expected utilities of world states and actions.

## The SHRUTI architecture

First we present the basic elements of the SHRUTI architecture. The model is described in considerably more detail in [Shastri, 1999, Shastri and Ajjanagadde, 1993, Shastri and Wendelken, 2000]. SHRUTI is a neurally plausible (connectionist) model that demonstrates how a network of neuron-like elements could encode a large body of structured knowledge and perform a variety of inferences within a few hundred milliseconds. SHRUTI suggests that the encoding of relational information (frames, predicates, etc.) is mediated by neural circuits composed of *focal clusters* and that the dynamic representation and communication of relational instances involves the transient propagation of *rhythmic* activity across these clusters. A role-entity binding is represented in this rhythmic activity by the *synchronous* firing of appropriate cells.

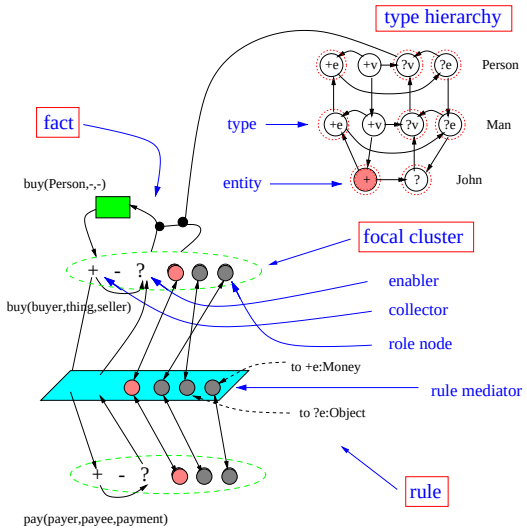


Figure 1: Diagram showing core elements of the SHRUTI model, including relational focal clusters, a fact, a rule, and a simple type hierarchy.

A focal cluster is a collection of nodes with varying functionality all subserving a common representation. A relational focal cluster consists of a positive (+) and a negative (-) collector node, an enabler (?) node, and role nodes. The activity of the positive (negative) collector node reflects the amount of evidence collected in support of belief (disbelief) in the given relation. Activity of the enabler (?) node reflects the strength with which information about the relation is being sought. A link from collector to enabler ensures that the system automatically seeks explanation for what it believes. Role bindings are represented by synchronous firing of relational role nodes with nodes in a connectionist type hierarchy. A relational cluster with active role bindings represents a relational instance. Rules are encoded with links that enable the propagation of rhythmic activity from one relational focal cluster to the next. Specifically, a rule is formed by linking the antecedent collector to the consequent collector, the consequent enabler to the antecedent enabler, and matching role nodes in both directions, through an intervening focal cluster

termed the *rule mediator*. Type restriction and instantiation of unbound variables are handled via connections between the rule mediator structure and the type hierarchy. Long-term facts are encoded in SHRUTI as temporal pattern matching circuits. *Episodic facts* (E-facts) are tuned to particular relational instances and represent specific knowledge or memories, while *taxon facts* (T-facts) are typically responsive to a range of relational activations and represent more general statistical knowledge about the world.

## Probabilistic reasoning

Previous work has shown that the inferential behavior of SHRUTI does not, in most cases, stray far from a probabilistic ideal [Wendelken and Shastri, 2000]. With appropriate assignment of link weights, a simple rule structure can be shown to compute probabilities correctly in both the forward and backward direction. A set of evidence combination functions allows for flexible combination of evidence from multiple sources, while maintaining a relatively simple connectionist structure in which each antecedent communicates with the consequent via a single weighted link [Shastri and Wendelken, 1999]. Explaining away occurs via inhibitory interconnections between antecedents, so false patterns of circular reasoning are not introduced.

Inference in SHRUTI is essentially an anytime algorithm. Unlike in a belief net, responses to a query are generated almost immediately, based on the prior information stored for the queried relation. As inference is allowed to progress, early estimates are repeatedly refined as more and more evidence is brought in from further up or down the causal chain. In a neural system, the depth to which this search for evidence occurs would be limited, such that only evidence within a certain distance (along any casual chain) would be considered. Presumably, this depth could be modulated by attention or other factors. Importantly, this is a model which scales up naturally to large domains without performance loss (with reference to a parallel network of nodes and links).

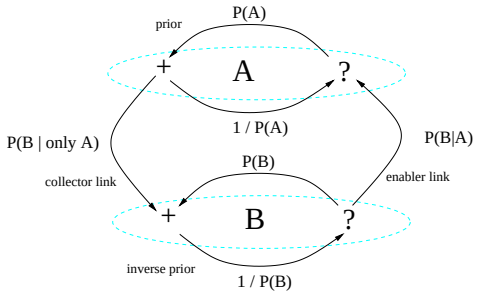


Figure 2: An illustration of the link weights for a simple rule (roles not shown). If B is believed true (+:B active with value 1.0) then activity at +:A will equal P(A—B)

## Representing utility in SHRUTI

SHRUTI's representation of utility [von Neumann and Morgenstern, 1947] is analogous to its representation of belief. This consists primarily of a set of utility nodes associated with each relational focal cluster, reward facts denoting reward and punishment, value facts denoting learned utility values, probabilistically weighted utility-carrying connections between relations, and various modulatory mechanisms that affect utility flow differently in different situations. Thus belief and utility in SHRUTI are tightly integrated, sharing much of the same structure, and are not separate modules in any conventional sense.

### Utility nodes

Recall that the representation of beliefs in SHRUTI is built around relational focal clusters, which contain several different types of nodes including positive and negative collectors, an enabler, and role nodes. Alongside these nodes representing belief, there are additional nodes representing associated utility. Thus there is a utility node tied to each of the two collectors, with activation range [-1,1]. These nodes are denoted by $+ and $-; positive activity of $+ ($-) indicates that positive utility value is associated with the truth (falsity) of the relation, while negative activation value of $+ ($-) indicates that negative utility is associated with the truth (falsity) of the relation. Links from each utility node to the enabler node ensure that whenever something is marked as having utility, it is automatically investigated by the system.

Activation of a relational utility node can indicate that reward is currently being experienced, or that it is expected. In either case, it reflects not only reward that is directly associated with its relation (as, for example, satisfying a sweet tooth is associated with eating cake), but also sources of reward that are more distantly related (such as potential weight gain). In this respect, the utility node is comparable to the value funcion of traditional reinforcement learning; however, utility node activity is transient and cannot by itself represent any permanent learned value associated with a relation instance (how this information is maintained will be described shortly). Instead, activity at a relational utility node reflects the combination of more permanent representations of value with the transient factors that make up current context.

### Reward facts

Some relations have *reward facts* (R-facts) tied to them, designating certain relational instances as goals. Reward facts represent the source of reward and punishment in the system. Activation of a positive reward fact indicates the attainment (real or imagined) of some reward, while activation of a negative reward fact indicates the suffering (real or imagined) of some punishment. Like episodic facts in the belief system, reward facts are temporal pattern matching circuits that respond only when the specified set of role-fillers are active. In this case, activation of a relational collector along with synchronous activation of role nodes and appropriate type

node role fillers leads to activation of an associated fact node, which in turn leads to activation of that relation's appropriate utility node. Many different reward facts can be linked to a single relation; for example, a relation like $eat(x)$ might have associated with it positive reward facts such as $eat(Cake)$ as well as negative reward (punishment) facts such as $eat(Dirt)$.

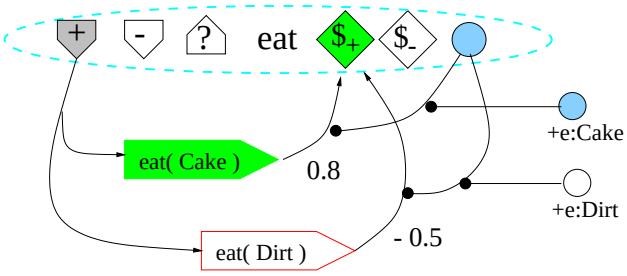


Figure 3: Two reward facts for the relation eat(x)

Research with rats and brain-stimulus reward suggests that both idiosyncratic and common currency representations of utility exist in the brain [Shizgal, 1998]. The representation of utility as activation values of relational utility nodes is a common currency representation which allows the activity of one node to be directly compared to the activity of another in order to guide decision-making. This is vital in order to allow successful decision making that takes into account disparate sources of reward and punishment. More domain-specific representations of utility must also exist, since the relative weighting of utilities from different sources can vary. The utility of eating, for example, is greatly influenced by degree of hunger, while the utility of play is not. Reward facts represent the connection between the common currency and the more domain-specific representations of utility. In order to model the latter, we allow that the weights on reward facts might vary depending on some internal state of the agent.

## Value facts

While relational utility nodes represent value estimates in the current context, and reward facts represent basic goals, the task of storing learned value estimates rests with the *value facts*, or V-facts. Value facts are similar in form to reward facts, but instead of directly representing reward, they represent predicted future reward. For both value facts and reward facts, utility values are stored as link weights (specifically, as the weight on the link leading from the fact node to the associated relational utility node). The value fact associated with a relation plays a similar role to the value function in traditional reinforcement learning, and the update function for a value fact, depending as it does on local reward and maximization (or some other combination) of utility values of possible consequents, closely resembles the Bellman equation [Bellman, 1957]. Note, however, that value updates in SHRUTI depend only on activity of a few connected predicates, and not on the entire system state. Because of the similarity in the Bellman equation and SHRUTI's value-updating algorithm, the latter has been termed Causal Heuristic Dynamic Programming (CHDP) [Thompson and Cohen, 1999]. Like taxon facts in the belief system, value facts hold a statistical summary of past activity. They too are associative, meaning that matching of relational activity to the fact is stronger with more role matches, but is not necessarily blocked by a single role mismatch; this helps with generalization of value to multiple related instances.

A typical relation has many value facts associated with it, some very specific and some quite general. In this way, particularly important or salient items are explicitly encoded, whereas novel or less important items can fall back on more general representations. For the hypothetical agent for which eating cake is a paramount goal, $find(Cake)$ should be a highly-rewarding value fact. Eating other things may still be beneficial, so the more general $find(Food)$ may also appear as a weaker value fact; finding anything is more often good than bad, so even the most general value fact $find(Thing)$ might appear in the agent's internal representation. When the agent with these value facts happens upon a dollar bill, it will immediately perceive this as a positive situation according to the value of the $find(Thing)$ value fact. If finding money turns out to be significantly more rewarding than finding that average-value random thing, then this should be learned and explicitly represented as a new value fact.

## Communication of utilities

Links connect utility nodes of different relations in the same way that they connect belief nodes. These links run parallel to the belief system connections, but in the consequent to antecedent (backward) direction. Figure 4 provides a simple illustration of these connections: for the rule $A \wedge B \Rightarrow C$, there are utility connections from the utility nodes of $C$, through the rule mediator, back to those of $A$ and $B$. Weights on these connections are similar to the weights on the collector-collector links. Their purpose is to introduce probability into the calculations of value, such that the value estimate at some antecedent relation is based on both the value of its consequent (activity at its utility node) and the probability that it will be reached (weight on the connecting link). For the rule $A \Rightarrow C$, where the utility node of C ($:$C$) has a value of $\alpha$, the utility node of A ($:$A$) should obtain the value $\alpha \times P(C|A)$.

This structure has the effect that assertion of a particular goal, via activation of a utility node, leads in the simplest case to assertion of its potential causes as subgoals, via spreading activation backwards along the causal chain. Belief in some relation, represented as activation of a collector node, leads to internal reward or punishment (activation of a reward fact) or recognition that such reward or punishment is likely (activation of a value fact) if there is an intact causal chain leading from that relation to some goal relation.

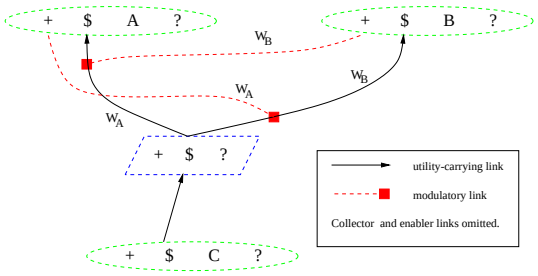


Figure 4: A diagram showing structure of utility connections for a two-antecedent rule.

## Utility modulation

The model of utility propagation described so far is perfectly adequate for simple cause-effect relationships or chains of these. However, with multiple-antecedent or multiple-consequent rules, or with multiple rules involving a common relation, additional mechanisms must be introduced. Consider first a rule with two antecedents, such as $find(x) \wedge edible(x) \Rightarrow eat(x)$. The utility of finding something, which is derived from the utility of eating something, depends directly on whether or not that thing is edible. Thus, there should be an interaction between the two antecedents such that if $edible(x)$ is false, then the propagation of utility from $eat(x)$ to $find(x)$ is at least partially blocked. The reverse holds true as well - utility of a thing being edible depends not only on the utility of eating it, but also on whether or not it has been found.

The interaction described here is appropriate for the *and*-combination, but different interactions should occur when different relations hold between the antecedents and the consequent. For example, when antecedents are combined with an *or* function, then belief in the truth of one should tend to discount the propagation of utility to the others. In this case, when one cause is established, then redundant causes are no longer particularly useful. For the *avg* (weighted average) function, each antecedent contributes independently to the total, and so belief in the truth of one antecedent should have no impact on the perceived utility of another.

In general, the utility value at an antecedent relation should reflect the value of any associated consequents times the extent to which truth of the antecedent affects truth of the consequent. For a rule with antecedents $A$ and $B_1$ through $B_n$ and consequent $C$, this might be stated as "What difference does $A$ make, in the context $B_1$ ... $B_n$, for the attainment of $C$", or in terms of probabilities, $P(C|A,B_1 = b_1,\ldots,B_n = b_n) - P(C|\neg A,B_1 = b_1,\ldots,B_n = b_n)$.

If the above expression is expanded for each different combination function, an interesting result is obtained, namely, that it is possible to compute it exactly for each different combination function using only the existing weight on the utility link along with a single additional weight from each associated antecedent. This relative simplicity of the resulting connectionist structure is important, since it lends plausibility to the notion that such a mechanism could be learned in the brain. Results for three combination functions, *and*, *or*, and *avg*, are shown below. The connectionist structure that computes these functions is shown in figure 4.

| ECF | $:A/$:C |
|---|---|
| *and* | $W_A \cdot \prod_{i=1}^{n}(1-(1-b_i)W_{B_i})$ |
| *or* | $W_A \cdot \prod_{i=1}^{n}(1-b_i W_{B_i})$ |
| *avg* | $W_A$ |

Action focal clusters are given special treatment within this framework. Since the agent has control over whether or not an action is performed, activity of an action collector does not modulate the utility values flowing to any sibling antecedents. Also, while activity of an action's utility node indicates that the action is beneficial or harmful, activity of its enabler simply indicates that the action is potentially relevant.

## Distribution and recombination

Just like beliefs, utilities from different sources must be combined. In general, the same approach is used here as with calculation of belief - a range of simple evidence combination functions are available and can be inserted into the connectionist structure as appropriate. Because many rewards are generally better than one, combination functions selected should generally have the property that a combined utility value is greater than any of the individual utilities; summation and *or* are two likely candidates. However, using such a combination function leads to a difficult problem when we allow multiple paths to exist between two relations. Consider the scenario, illustrated in figure 5 where exploration can lead to finding fruit or finding game, and that either of these consequents can lead to the goal relation of being able to eat. Utility associated with eating is propagated in full to both $findFruit$ and $findGame$ (assuming an *or*-combination and that neither is currently true), and from each it is further propagated back to *explore*. Now if *explore* has the sort of combination function described above, it can obtain a local utility value greater than that originating at the goal *eat*. This is clearly an unacceptable situation, and it comes from the fact that locally there is no information to distinguish between utility arriving from different sources (which should be added together) and utility values that originate from the same source (which should not).

One solution to this problem might be to disallow multiple paths between relations. Indeed, this is the solution adopted for belief nets to solve essentially the same problem. However, connections between relations are assumed to be learned from experience based only on local information; it is difficult to imagine any plausible mechanism by which learning of multiple paths could be inhibited when these provide the best fit for experience. Another solution would be to reduce the amount of utility distributed along each path according to the number

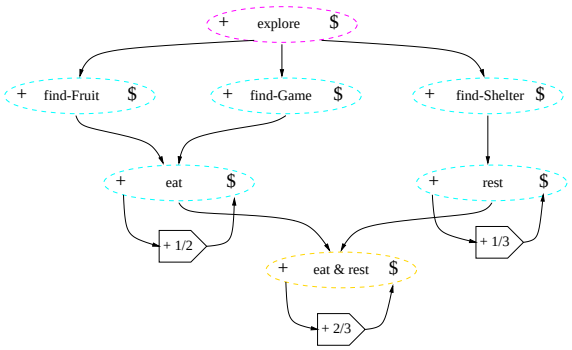


Figure 5: A proposed solution to the problem of utility combination.

of such paths; in this case that would mean that only half of utility at the *eat* relation is propagated to $findFruit$ and $findGame$. But this clearly leads to an underestimation of utility along each. Finally, we might abandon the use of summation and similar combination functions for utility and instead use something like *max*. This solves the problem at hand but also makes it impossible to productively combine multiple sources of utility. The "commom currency" representation of utility becomes somewhat modified; utilities can be directly compared within this framework but can no longer be directly combined. Instead, reward combinations must be explicitly represented in order to be used. This is illustrated in figure 5, where the basic goals *eat* and *rest* are supplemented by a combination goal *eat&rest*.

## Simulation example

The operation of the network is illustrated here. The screen capture from the Shruti-Agent Simulator in Figure 6 shows a simple network representing the caveman's dilemma of whether to hunt or gather. Successful hunting yields the greatest reward (represented by the reward fact $eat(Game)$). Gathering, on the other hand, is more reliable, but only productive during the right season. We examine the propagation of beliefs and utilities around this simple network in detail. First, suppose that the caveman agent is hungry, and hence reward facts related to eating are fully active. Eating game or eating fruit are the current active goals of the system. Activity from the reward facts flows to multiple banks of the eat relation and from there back to $kill(Game)$ and $find(Fruit)$. The agent has realized that either killing game or finding fruit would be useful eventualities. Alongside the propagation of utility, a querying belief state is also being transmitted from relation to relation; this is represented in the activity of the enabler nodes. Since neither eventuality is thought to be true of the current world state, there is no competitive modulation of utility values; thus, $kill(Game)$ has the full 0.8 value from the $eat(Game)$ reward fact while $find(Fruit)$ has the full 0.6 from $eat(Fruit)$.

Utility value propagates further back to the *hunt* relation, this time modified by the uncertainty of hunting,

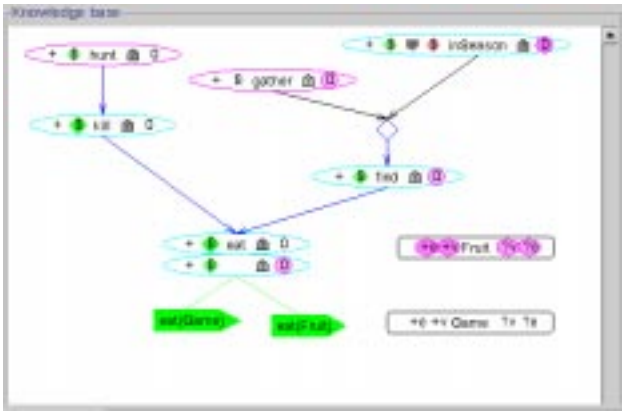

Figure 6: A captured moment from the simulation of the caveman scenario. Activity of -:inSeason blocks the propagation of utility to $:gather, resulting in a higher valuation of the hunt action.

such that $hunt(Game)$ has associated with it a utility of 0.4. In order for gathering fruit to be perceived as useful, the agent must have some knowledge that the fruit is in season. Suppose first that the query $inSeason(Fruit)$ is answered in the negative, either as a result of immediate knowledge of the agent or of further reasoning along paths not illustrated here. Then, according to the equation for distribution of utility values around an *and*-combination given above, and by means of a simple inhibitory mechanism, the flow of utility to the *gather* relation is blocked. Simlarly, If *inSeason* is uncertain, utility propagation to *gather* will be partially blocked. In either case, the *hunt* action, with a higher utility, will be favored. This is the situation illustrated in figure 6 and indicated by a numeral one in figure 7. If on the other hand the agent is reasonably certain that fruit is in season, then sufficient utility will propagate from the $find$ relation and *gather* will obtain a higher utility value than *hunt*, marking it as the preferred action.

Figure 8 illustrates a larger domain wherein the possibility of moving to a location where food can be found is included, as is the possiblility of being injured while hunting. When the assumption is made that *skilled* is true, (i.e. caveman is a skilled hunter), utility and belief propagate in this network such that $moveTo(RiverBank)$ (i.e. go to where the game is) is marked as a useful action.

## Conclusion

We have demonstrated that SHRUTI, a neurally plausible model of knowledge representation and reasoning, can be enhanced to deal effectively with utilities, values, and goals. The resulting connectionist machinery is sufficient to guide an agent through a wide range of decision-making tasks, such as those illustrated in the previous examples. However, there is a class of decision problems for which the model presented here is inadequate.

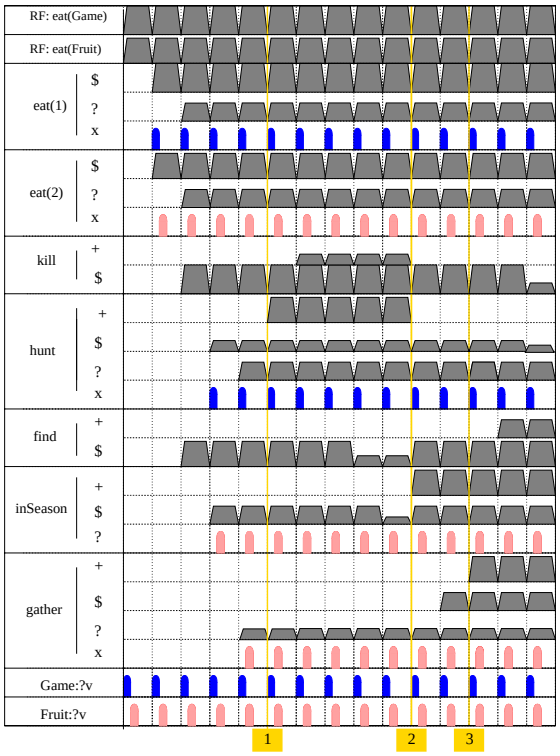


Figure 7: A stylized trace of node activations during execution of the caveman scenario.

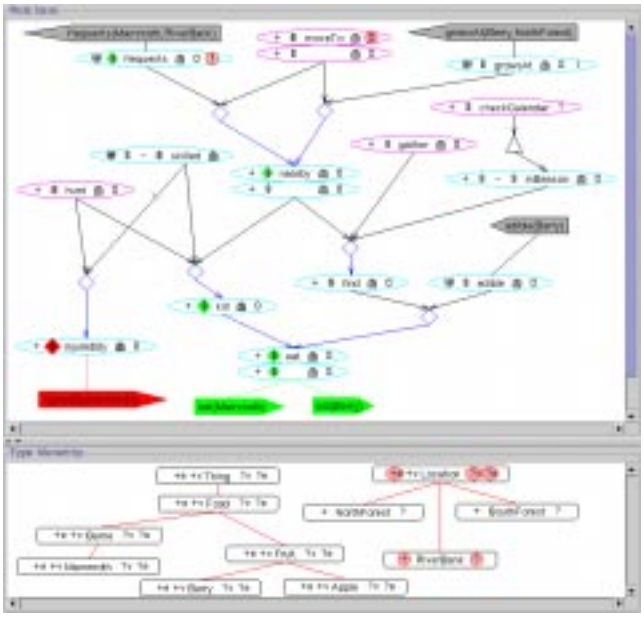

Figure 8: An expanded version of the caveman scenario.

In order to deal effectively with complex decision tasks, a measure of higher-level control must be introduced. Extensions to the model described here that enable it to perform complex decision-making and planning are described elsewhere [Wendelken and Shastri, 2002, Garagnani et al., 2002].

## References

[Bellman, 1957] Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

[Garagnani et al., 2002] Garagnani, M., Shastri, L., and Wendelken, C. (2002). A connnectionist model of planning via back-chaining search. In *Proc. 24th Conf. of the Cognitive Science Society*.

[Shastri, 1999] Shastri, L. (1999). Advances in SHRUTI - a neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony. *Applied Intelligence*, 11.

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

[Shastri and Wendelken, 1999] Shastri, L. and Wendelken, C. (1999). Soft computing in SHRUTI. In *Proc. 3rd Int. Symposium on Soft Computing*, pages 741–747, Genova, Italy.

[Shastri and Wendelken, 2000] Shastri, L. and Wendelken, C. (2000). Seeking coherent explanations - a fusion of structured connectionism, temporal synchrony, and evidential reasoning. In *Proc. 22nd Conf. of the Cognitive Science Society*, Philadelphia.

[Shizgal, 1998] Shizgal, P. (1998). *Foundations of hedonic psychology: Scientific perspectives on enjoyment and suffering*, On the neural computation of utility: implications from studies of brain stimulation reward.

[Thompson and Cohen, 1999] Thompson, B. and Cohen, M. (1999). Naturalistic decision making and models of computational intelligence. In A. Jagota et al. editors, *Connectionist Symbol Processing: Dead 0r Alive?*, volume 2 of *Neural Computing Surveys*, pages 1-40. http://www.icsi.berkeley.edu/ jagota/NCS.

[von Neumann and Morgenstern, 1947] von Neumann, J. and Morgenstern, O. (1947). *Theory of Games and Economic Behavior*. Princeton University Press.

[Wendelken and Shastri, 2000] Wendelken, C. and Shastri, L. (2000). Probabilistic inference and learning in a connectionist causal network. In *Proc. 2nd Int. Symposium on Neural Computation*.

[Wendelken and Shastri, 2002] Wendelken, C. and Shastri, L. (2002). Decision-making and control in a structured connectionist agent architecture. In *submitted*.