# 1

# LEARNING METHODS FOR EVOLVING INTELLIGENT SYSTEMS

Ronald R. Yager

**Abstract:** In this work we describe two instruments for introducing evolutionary behavior into intelligent systems. The first is the hierarchical prioritized structure (HPS) and the second is the participatory learning paradigm (PLP).

# **1.1 INTRODUCTION**

The capacity to evolve and adapt to a changing environment is fundamental to human and other living systems. Our understanding of the importance of this goes back to at least Darwin [1]. As we begin building computational agents that try to emulate human capacities we must also begin considering the issue of systems that evolve autonomously. Our focus here is on knowledge-based/intelligent systems. In these types of systems, implementing evolution requires an ability to balance learning and changing while still respecting the past accumulated knowledge. In this work we describe two instruments for introducing evolutionary behavior into our intelligent systems. The first is the *hierarchical prioritized structure (HPS)* and the second is the *participatory learning paradigm (PLP)*. The HPS provides a hierarchical framework for organizing knowledge. Its hierarchical nature allows for an implicit prioritization of knowledge so that evolution can be implemented by locating new knowledge in a higher place in the

Evolving Intelligent Systems: Methodology and Applications, Edited by Plamen Angelov, Dimitar P. Filev, and Nikola Kasabov

Copyright © 2010 Institute of Electrical and Electronics Engineers

hierarchy. Two important aspects of the structure are considered in this work. The first is the process of aggregating information provided at the different levels of the hierarchy. This task is accomplished by the hierarchical updation operator. The other aspect is the process of evolving the model as information indicating a change in the environment is occurring.

The participatory learning paradigm provides a general learning paradigm that emphasizes the role of what we already know in the learning process. Here, an attribute about which we are learning is not viewed simply as a target being blindly pushed and shoved by new observations but one that participates in determining the validity of the new information.

Underlying both these instruments is a type of nonlinear aggregation operation that is adjudicating between knowledge held at different levels. Central to this type of aggregation is a process in which the privileged knowledge is deciding on the allowable influence of the less-favored knowledge.

# **1.2 OVERVIEW OF THE HIERARCHICAL PRIORITIZED MODEL**

In [2-5] we described an extension of fuzzy modeling technology called the hierarchical prioritized structure (HPS), which is based on a hierarchical representation of the rules. As we shall subsequently see, this provides a rich framework for the construction of evolving systems. The HPS provides a framework using a hierarchical representation of knowledge in terms of fuzzy rules is equipped with machinery for generating a system output given an input. In order to use this hierarchical framework to make inferences, we needed to provide a new aggregation operator, called the hierarchical updation (HEU) operator, to allow the passing of information between different levels of the hierarchy. An important feature of the inference machinery of the HPS is related to the implicit prioritization of the rules; the higher the rule is in the HPS, the higher its priority. The effect of this is that we look for solutions in an ordered way, starting at the top. Once an appropriate solution is found, we have no need to look at the lower levels. This type of structure very naturally allows for the inclusion of default rules, which can reside at the lowest levels of the structure. It also has an inherent mechanism for evolving by adding levels above the information we want to discard.

An important issue related to the use of the HPS structure is the learning of the model itself. This involves determination of the content of the rules as well the determination of the level at which a rule shall appear. As in all knowledge-based systems, learning can occur in many different ways. One extreme is that of being told the knowledge by some (usually human) expert. At the other extreme is the situation in which we are provided only with input–output observations and we must use these to generate the rules. Many cases lie between these extremes.

Here we shall discuss one type of learning mechanism associated with the HPS structure that lies between these two extremes, called the *DELTA* method. In this we initialize the HPS with expert-provided default rules and then use input–output

observations to modify and adapt this initialization. Taking advantage of the HPS we are able to introduce exceptions to more general rules by giving them a higher priority, introducing them at a higher level in the hierarchy. These exceptions can be themselves rules or specific points. This can be seen as a type of forgetting mechanism that can allow the implementation of dynamic adaptive learning techniques that continuously evolve the model.

# **1.3 THE HPS MODEL**

In the following, we describe the basic structure and the associated reasoning mechanism of the fuzzy systems modeling framework called the hierarchical prioritized structure (HPS).

Assume we have a system we are modeling with inputs V and W and output U. At each level of the HPS, we have a collection of fuzzy *if-then* rules. Thus for level *j*, we have a collection of  $n_i$  rules:

If V is 
$$A_{ii}$$
 and W is  $B_{ii}$ , then U is  $D_{ii}$   $i = 1, ..., n_i$ 

We shall denote the collection of rules at the *j*th level as  $R_j$ . Given values for the input variables,  $V = x^*$  and  $W = y^*$ , and applying the standard fuzzy inference to the rules at level *j*, we can obtain a fuzzy subset  $F_j$  over the universe of *U*, where  $F_j(z) = \frac{1}{T} \sum_{i=1}^{n_j} \lambda_{ji} D_{ji}(z)$  with  $\lambda_{ji} = A_{ji}(x^*) \wedge B_{ji}(y^*)$  and  $T = \sum_{i=1}^{n_j} \lambda_{ji}$ . Alternatively, we can also calculate  $F_j(z) = \text{Max}_i[\lambda_{ji} \wedge D_{ji}(z)]$ . We denote the application of the basic fuzzy inference process with a given input,  $V = x^*$  and  $W = y^*$ , to this sub-rule base as  $F_j = R_j \bullet$ . Input.

In the HPS model, the output of level *j* is a combination of  $F_j$  and output of the preceding level. We denote the output of the *j*th level of the HPS as  $G_j$ .  $G_j$  is obtained by combining the output the previous level,  $G_{j-1}$ , with  $F_j$  using the hierarchical updation (HEU) aggregation operator subsequently to be defined. The output of the last level,  $G_n$ , is the overall model output *E*. We initialize the process by assigning  $G_0 = \emptyset$ .

The key to inference mechanism in the HPS is the HEU aggregation operator  $G_j = \gamma(G_{j-1}, F_j)$ , where

$$G_i(z) = G_{i-1}(z) + (1 - \alpha_{i-1})F_i(z)$$

Here,  $\alpha_{j-1} = \text{Max}_{z}[G_{j-1}(z)]$ , the largest membership grade in  $G_{j-1}$ . See Figure 1.1.

Let us look at the functioning of this operator. First we see that it is not pointwise in that the value of  $G_j(z)$  depends, through the function  $\alpha_{j-1}$ , on the membership grade of elements other than z. We also note that if  $\alpha_{j-1} = 1$ , no change occurs. More generally, the larger  $\alpha_{j-1}$  the less the effect of the current level. Thus, we see that  $\alpha_{j-1}$  acts as a kind of choking function. In particular, if for some level j we obtain a situation in which  $G_j$  is normal, and has an element with membership grade one, the process of aggregation stops.



Figure 1.1. Hierarchical prioritized structure.

It is also clear that  $G_{j-1}$  and  $F_j$  are not treated symmetrically. We see that, as we get closer to having some elements in  $G_{j-1}$  with membership grade one, the process of adding information slows. The form of the HEU essentially implements a prioritization of the rules. The rules at the highest level of the hierarchy are explored first; if they find a good solution, we look no further at the rules.

Figure 1.2 provides an alternative view of the HPS structure.



Figure 1.2. Alternative view of HPS.

## 1.4 AN EXAMPLE OF HPS MODELING

We shall illustrate the application of this structure with the following example

**Example :** Consider a function W = F(U, V) defined on U = [0,10] and V = [0,10]. Refer to Figure 1.3 for the following discussion. We shall assume that in the white areas the value of the function is *small* and in the black areas the value of the function is *large*. The figure could, for example, be representative of a geospatial mapping in which W is the altitude and the black areas correspond to a mountain range.

We can describe this functional relationship by the following three-level HPS structure:

**Level-1:** If U is *close to five*, then W is *small*. (Rule 1) **Level-2:** If  $((U-5)2+(V-5)^2)^{0.5}$  is *about two*, then W is *large*. (Rule 2) **Level-3:** If U and V are *anything*, then W is *small*. (Rule 3)

For our purposes, we define the underlined fuzzy subsets as follows:

$$Small = \left\{\frac{0.3}{5}, \frac{0.6}{6}, \frac{1}{7}, \frac{0.6}{8}, \frac{0.3}{9}\right\} \text{ and } Large = \left\{\frac{0.3}{21}, \frac{0.6}{22}, \frac{1}{23}, \frac{0.6}{24}, \frac{0.3}{25}\right\}$$
$$close \ to \ five(U) = e^{-(U-5)^2} \ \text{and } about \ five(r) = e^{-(r-2)^2}$$

Let us look at three special cases.

1. U = 5 and V = 6. Here rule one fires to degree 1. Hence the output of the first level is  $G_1 = \left\{\frac{0.3}{5}, \frac{0.6}{6}, \frac{1}{7}, \frac{0.6}{8}, \frac{0.3}{9}\right\}$ . Since this has maximal membership grade equal to one, the output of the system is  $G_1$ .



Figure 1.3. Structure of F(U, V).

- 2. U = 6 and V = 6. Here the firing level of Rule 1 is 0.02 and the output of the first level is  $G_1 = \left\{\frac{0.2}{5}, \frac{0.2}{6}, \frac{0.2}{7}, \frac{0.2}{8}, \frac{0.2}{9}\right\}$  and has maximal firing level 0.2. Applying the input to Rule 2, we get a firing level of 1. Thus  $F_2 = \left\{\frac{0.3}{21}, \frac{0.6}{22}, \frac{1.3}{23}, \frac{0.6}{24}, \frac{0.3}{25}\right\}$ . Thus  $G_2(z) = \left\{\frac{0.2}{5}, \frac{0.2}{7}, \frac{0.2}{8}, \frac{0.2}{9}, \frac{0.24}{21}, \frac{0.46}{22}, \frac{0.3}{23}, \frac{0.46}{24}, \frac{0.24}{25}\right\}$  and therefore  $\alpha_2 = 0.8$ . Applying the input to Rule 3 three we get firing level 1. Thus  $F_2 = \left\{\frac{0.3}{21}, \frac{0.6}{22}, \frac{1.3}{23}, \frac{0.6}{24}, \frac{0.3}{25}\right\}$ . Since  $G_3(z) = G_2(z) + (1 0.8)$   $F_3(z)$  we get  $G_3 = \left\{\frac{0.25}{25}, \frac{0.312}{6}, \frac{0.4}{7}, \frac{0.31}{8}, \frac{0.26}{22}, \frac{0.24}{23}, \frac{0.46}{24}, \frac{0.3}{25}, \frac{0.46}{24}, \frac{0.24}{25}\right\}$ . Defuzzifying this value we get W = 16.3.
- 3. U=9 and V=8. In this case, the firing level of Rule 1 is 0; thus  $G_1 = \emptyset$ . Similarly, the firing level of Rule 2 is also 0, and hence  $G_1 = \emptyset$ . The firing level of Rule 3 is one, and hence the overall output is *small*.

# **1.5 HIERARCHICAL UPDATION OPERATOR**

Let us look at some of the properties of this hierarchical updation operator  $\gamma$ . If *A* and *B* are two fuzzy sets of *Z*, then we have  $\gamma(A, B) = D$ , where  $D(z) = A(z) + (1 - \alpha) B(z)$  with  $\alpha = \text{Max}_{z \in Z}(A(z))$ . This operator is not pointwise as  $\alpha$  depends on A(z) for all  $z \in Z$ . This operator is a kind of disjunctive operator; we see that  $\gamma(A, \emptyset) = A$  and  $\gamma(\emptyset, B) = B$ . This operator is not commutative  $\gamma(A, B) \neq \gamma(B, A)$ . An important illustration of this is that while  $\gamma(Z, B) = B$ , we have  $\gamma(A, Z) = D$ , where  $D(z) = A(z) + (1 - \alpha)$ . The operator is also nonmonotonic. Consider  $D = \gamma(A, B)$  and  $D' = \gamma(A', B)$ , where  $A \subseteq A'$ . Since  $A(z) \leq A'(z)$  for all *z*, then  $\alpha \leq \alpha'$ . We have  $D'(z) = A'(z) + (1 - \alpha') B(z)$  and D(z) = $A(z) + (1 - \alpha) B(z)$ . Since  $A \subseteq A'$ , monotonicity requires that  $D'(z) \geq D(z)$  for all *z*. To investigate the monotonicity of  $\gamma$  we look at  $D'(z) - D(z) = A'(z) - A(z) + B(z)(\alpha - \alpha')$ .

Thus, while A'(z) - A(z) = 0, we have  $(\alpha - \alpha') \le 0$ , and therefore there is no guarantee that  $D'(z) \ge D(z)$ .

We can suggest a general class of operators that can serve as hierarchical aggregation operators. Let *T* be any *t*-norm and *S* be any *t*-conorm [6]. A general class of hierarchical updation operators can be defined as D = HEU(A, B), where  $D(z) = S(A(z), T(1 - \alpha, B(z)))$  with  $\alpha \ge \text{Max}_z(A(z))$ .

First, let us show that our original operator is a member of this class. Assume *S* is the bounded sum, S(a, b) = Min[1, a + b] and *T* is the product, S(a, b) = ab. In this case,  $D(z) = \text{Min}[1, A(z) + \overline{\alpha}B(z)]$ . Consider the term  $A(z) + \overline{\alpha}B(z)$ . Since  $\alpha = \text{Max}_z[A(z)]$ , then  $\alpha = A(z)$  and therefore  $A(z) + \overline{\alpha}B(z) \le \alpha + (1 - \alpha)$   $B(z) \le 1$ . Thus  $D(z) = A(z) + (1 - \alpha) B(z)$ , which was our original suggestion.

We can now obtain other forms for this HEU operator by selecting different *S* and *T*. If  $S = Max(\lor)$  and  $T = Min(\land)$ , we get

$$D(z) = A(z) \lor (\bar{\alpha} \land B(z))$$

If S is the algebraic sum, S(a, b) = a + b - ab and T is the product, then

$$D(z) = A(z) + \bar{\alpha}B(z) - \bar{\alpha}A(z)B(z) = A(z) + \bar{\alpha}\bar{A}(z)B(z)$$

If we use S as the bounded sum and T as the Min, we get

$$D(z) = \operatorname{Min}[1, A(z) + \bar{\alpha} \wedge B(z)]$$

Since  $\alpha \ge A(z)$ , then  $A(z) + \overline{\alpha} \land B(z) \le \alpha + (1 - \alpha) \land B(z) \le \alpha + (1 - \alpha) \le 1$ ; hence we get

$$D(z) = A(z) + \bar{\alpha} \wedge B(z)$$

More generally, if S is the bounded sum and T is any t-norm, then

 $D(z) = \operatorname{Min}[1, A(z) + T(\bar{\alpha} \wedge B(z))]$ 

Since  $T(\bar{\alpha} \wedge B(z)) \leq \bar{\alpha} = 1 - A(z)$ , then

$$D(z) = A(z) + T(\bar{\alpha}, B(z))$$

#### **1.6 THE DELTA METHOD OF HPS LEARNING**

In the preceding, we have described the inference mechanism associated with the hierarchical prioritized structure. We have said nothing about how we obtained the rules in the model. The issue of the construction of the HPS model is an important one. The format of the HPS model allows many different methods for obtaining the model.

In this section we shall outline a dynamic learning approach for the construction of an HPS that allows the system to continuously learn and evolve. We call this the *default-exception-learning-that's-adaptive* (*DELTA*) method for HPS. In this approach we initialize the HPS by providing a default representation of the relationship we are trying to model. With this default relationship we allow the system builder to provide an initial model of the system that will be augmented as we get more data about the performance of the actual system. This default model can be as simple or as complex as the designer's knowledge of the system can support. In this approach the augmentation of the model will be one in which we add specific observations and rules to the HPS. The addition of knowledge to the structure will be driven by observations that are exceptions to what we already believe the situation to be. The exceptions will be captured and stored at the top level of the hierarchy. Groups of exceptions shall be aggregated to form new rules, which will stored at the next level of the hierarchy.

We shall use a three-level HPS model as shown in Figure 1.4. For ease of explanation we shall assume a model having a single input. The extension to multiple inputs is straightforward.

The construction of the structure is initialized with the first and second levels being empty. The third level is initialized with our default information about the structure of the relationship between the input and output variables *V* and *U*. In particular, the third level



Figure 1.4. Exception-based hierarchy.

contains default rules of the form

If V is A then U is f(V)

In the above, f(V) is some prescribed functional relationship and A is a fuzzy subset indicating the range of that default rule. The knowledge in the default can be any manifestation of the prior expectation of the system modeler. It could be a simple rule that says U = b for all values of V, a linear relationship that says  $U = k_1 + k_2 V$  for all values of V, or a collection of more complex rules based on some partitioning of the input space.

The HPS model will evolve based on observations presented to it, especially observations that are exceptions to what we already believe. In particular, the information in levels 1 and 2 will be obtained from the observations presented to the model. As we shall see, level 1 will contain facts about individual observations that are exceptions to what we already believe. Level 2 shall contain rules that aggregate these exceptions. The aggregation process used here is very much in the spirit of the *mountain clustering method* [7–10] introduced by Yager and Filev.

In Figure 1.5, we provide a flow diagram of the basic learning mechanism used in this approach. In the following, we describe the basic mechanism for the construction of this type of HPS. An observation (x, y) is presented to the HPS model. We calculate the output for the input x, and denote this  $y^*$ . We then compare this calculated output with the desired output. If y and  $y^*$  are close to each other, we can disregard this data and assume it doesn't provide any learning. If y and  $y^*$  are not close, we use this data to modify the HPS. More specifically for the pair  $(y, y^*)$  we calculate the value  $Close(y, y^*) \in [0,1]$  indicating the degree of closeness of the observed value and the calculated value. If  $Close(y, y^*) \ge \alpha$ , a threshold level, we disregard the data. If  $Close(y, y^*) < \alpha$ , we use this data to update the model. We denote for this observation  $\mathcal{P} = 1 - Close(y, y^*)$  as a measure of this



Figure 1.5. Schematic of learning process.

observation's ability to count as an exception, its strength of exception. We add to the top level of the current HPS model this observation in the form of a point rule,

#### If V is x, then U is y

For simplicity, we shall denote this rule as the **point**(x, y). We further associate with this rule a value M, which we initialize as  $\mathcal{P}$ , its strength of exception. As we shall see, this M value will be used in the same way as the mountain function is used in the mountain method to help in the aggregation of point exceptions to form exception rules. We next update the M value for all the other exception rules in the top level of the HPS. For any point rule, if V is  $x_i$ , then U is  $y_i$ , in the top level we update  $M_i$  as

$$M'_i = M_i + \mathcal{P}e^{-\text{Distance}((x,y)-(x_i,y_i))}$$

Thus we see that as a result of experiencing an observation that is considered an exception we add this observation to a current model and modify the *M* value of all other exceptions by adding to them a value proportional to the strength of the current exception modulated by its distance to the current exception.

We next check to see whether the addition of this new exception has caused a accumulation of the exceptions that can be gathered to form an exception rule; here we use the M values.

Specifically, we find the data point in the top level that now has the highest M value. Let us denote this value as  $\hat{M}$  and assume it occurs for the point  $(\hat{x}, \hat{y})$ . If  $\hat{M} \ge \beta, \beta$  being a threshold value for exception rule formation, we create a new rule of the form

If V is about  $\hat{x}$ , then U is about  $\hat{y}$ 

where, as we have noted above,  $\hat{x}$  and  $\hat{y}$  are the coordinates of the data point with the largest *M* value. This new rule is added to the second level of HPS. Thus we see that a collection of exceptions close to each other focused at  $(\hat{x}, \hat{y})$  form an exception rule at the second level. We emphasize that it is with the aid of the *M* function that we measure the power of a exception point in the first level to be the nucleus of an exception rule in the second level.

The final step is the cleansing and reduction of the top level by eliminating the individual exception rules that are now accounted for by the formulation of this new rule at the second level. We first modify our function M at each point (x, y) in the top level to form M', where

$$M'(x,y) = M(x,y) - \hat{M}e^{-\text{Distance}((x,y) - (\hat{x},\hat{y}))}$$

We next eliminate all point rules for which  $M'(x, y) \le 1 - \alpha$ .

Further, we let  $\hat{A}$  and  $\hat{B}$  be the fuzzy subsets about  $\hat{x}$  and  $\hat{y}$ . For each exception point in the top level, we calculate  $\hat{A}(x_i)$  and  $\hat{B}(y_i)$  and let  $t_i = \text{Min } (A(x_i), B(y_i))$ . We then eliminate all exceptions for which  $t_i \ge \lambda$ , a threshold for exception cleansing.

It should be noted that the above procedure has a number of parameters affecting our actions. In particular we introduced  $\alpha$ ,  $\beta$ , and  $\lambda$ . It is with the aid of these parameters that we are able to control the uniqueness of the learning process. For example, the smaller we make  $\alpha$ , the more rigorous our requirements are for indicating an observation as an exception; it is related to our sensitivity to exceptions. The parameter  $\beta$  determines openness to the formulation of new rules. The choice of these parameters is very much in the same spirit as choice of the learning rate used in the classical gradient learning techniques such as back propagation. Experience with the use of this exception-based machinery will of course sharpen our knowledge of the effect of parameter selection. At a deeper level the selection of these parameters should be based on how we desire the learning to perform and gives us a degree of freedom in the design of our learning mechanism, resulting, just as in the case of human learning, in highly individualized learning.

It is important to emphasize some salient features of the DELTA mechanism for constructing HPS models. We see this has an adaptive-type learning mechanism. We initialize the system with current user knowledge and then modify this initializing knowledge based on our observations. In particular, as with a human being, this has the *capability* for continuously evolving. That is, even while it is being used to provide outputs it can learn from its mistakes. Also we see that information enters the systems as observations and moves its way down the system in rules very much in the way that humans process information in the face of experience.

# 1.7 INTRODUCTION TO THE PARTICIPATORY LEARNING PARADIGM

Participatory learning is a paradigm for computational learning systems whose basic premise is that learning takes place in the framework of what is already learned and believed. The implication of this is that every aspect of the learning process is effected and guided by the learner's current belief system. Participatory learning highlights the fact that in learning we are in a situation in which the current knowledge of what we are trying to learn participates in the process of learning about itself. This idea is closely related to Quine's idea of *web of belief* [11, 12]. The now-classic work by Kuhn [13] describes related ideas in the framework of a scientific advancement. With the



Figure 1.6. Partial view of a prototypical participatory learning process.

participatory learning paradigm we are trying to bring to the field of computational intelligence some important aspects of human learning. What is clear about human learning is that it manifests a noncommutative aggregation of information; the order of experiences and observations matters. Typically, the earlier information is more valued. Participatory learning has the characteristic of protecting our belief structures from wide swings due to erroneous and anomalous observations while still allowing the learning of new knowledge. Central to the participatory learning paradigm is the idea that observations conflicting with our current beliefs are generally discounted.

In Figure 1.6, we provide a partial systemic view of a prototypical participatory learning process that highlights the enhanced role played by the current belief system. An experience presented to the system is first sent to the acceptance or censor component. This component, which is under the control of the current belief state, decides whether the experience is compatible with the current state of belief; if it is deemed as being compatible, the experience is passed along to the learning components, which use this experience to update the current belief. If the experience is deemed as being too incompatible, it is rejected and not used for learning. Thus we see that the acceptance component acts as a kind of filler with respect to deciding which experiences are to be used for learning. We emphasize here that the state of the current beliefs participates in this filtering operation. We note that many learning paradigms do not include this filtering mechanism; such systems let all data pass through to modify the current belief state.

Because of the above structure, a central characteristic of the PLP (participatory learning paradigm) is that an experience has the greatest impact in causing learning or belief revision when it is compatible with our current belief system. In particular, observations that conflict too much with our current beliefs are discounted. The structure of the *participatory learning system (PLS)* is such that it is most receptive to learning when confronted with experiences that convey the message "What you know is correct except for this little part." The rate of learning using the PLP is optimized for situations in which we are just trying to change a small part of our current belief system. On the other hand, a PLS when confronted with an experience that says "You are all wrong; this is the truth" responds by discounting what is being told to it. In its nature, it is a conservative learning system and hence very stable. We can see that the participatory learning environment uses sympathetic experiences to modify itself. Unsympathetic observations are discounted as being erroneous. Generally, a system based on the PLP uses the whole context of an observation (experience) to judge something about the credibility of the observation with respect to the learning agent's beliefs; if it finds the whole



Figure 1.7. Fully developed prototypical participatory learning process.

experience credible, it can modify its belief to accommodate any portion of the experience in conflict with its belief. That is, if most of an experience or observation is compatible with the learning agent's current belief, the agent can use the portion of the observation that deviates from its current belief to learn.

While the acceptance function in PLP acts to protect an agent from responding to "bad" data, it has an associated downside. If the agent using a PLP has an incorrect belief system about the world, it allows this agent to remain in this state of blissful ignorance by blocking out correct observations that may conflict with this erroneous belief model. In Figure 1.7, we provide a more fully developed version of the participatory learning paradigm that addresses this issue by introducing an arousal mechanism in the guise of a critic.

The arousal mechanism is an autonomous component not under the control of the current belief state. Its role is to observe the performance of the acceptance function. If too many observations are rejected as being incompatible with the learning agent's belief model, this component arouses the agent to the fact that something may be wrong with its current state of belief; a loss of confidence is incurred. The effect of this loss of confidence is to weaken the filtering aspect of the acceptance component and allow incoming experiences that are not necessarily compatible with the current state of belief to be used to help update the current belief. This situation can result in rapid learning in the case of a changing environment once the agent has been aroused. Essentially, the role of the arousal mechanism is to help the agent get out of a state of belief that is deemed as false.

Fundamentally, we see two collaborating mechanisms at play in this participatory learning paradigm. The primary mechanism manifested by the acceptance function and controlled by the current state of belief is a conservative one; it assumes the current state of belief is substantially correct and requires only slight tuning. It rejects strongly incompatible experiences and doesn't allow them to modify its current belief. This mechanism manifests its effect on each individual learning experience. The secondary mechanism, controlled by the arousal mechanism, being less conservative, allows for the possibility that the agent's current state of belief may be wrong. This secondary mechanism is generally kept dormant unless activated by being aroused by an accumulation of input observations in conflict with the current belief. What must be

emphasized is that the arousal mechanism contains no knowledge of the current beliefs; all knowledge resides in the belief system. It is basically a scoring system calculating how the current belief system is performing. It essentially does this by noting how often the system has encountered incompatible observations. Its effect is not manifested by an individual incompatible experience but by an accumulation of these.

## **1.8 A MODEL OF PARTICIPATORY LEARNING**

As noted in the preceding, participatory learning provides a paradigm for constructing computational learning systems; as such, it can used in many different learning environments. In the following, we provide one example of a learning agent based on the PLP to illustrate some instantiation of the ideas described above [14]. While this is an important example of learning in a quantitative environment, we note that the PLP can be equally useful in the kind of symbolic learning environments found in the artificial intelligence and machine learning community.

In the illustration of PL that follows, we have a context consisting of a collection of variables, x(i), i = 1 to n. Here we are interested in learning the value of this collection of variables. It is important to emphasize the multidimensionality of the environment in which the agent is doing the learning. Multidimensionality, which is present in most real-world learning experiences, is crucial to the functioning of the participatory learning paradigm since the acceptability of an experience is based on the compatibility of the collection of observed values as a whole with the agent's current belief. For simplicity we assume that the values of the  $x(i) \in [0,1]$ . The current state of the system's belief consists of a vector  $V_{k-1}$ , whose components  $V_{k-1}(i)$ , i = 1 to n, consist of the agent's current belief about the values of x(i). It is what the agent has learned after k - 1 observation. The current observation consists of a vector  $D_k$ , whose components  $d_k(i)$ , i = 1 to n, are observations about the variable x(i). Using a participatory learning type of mechanism, the updation of our current belief is the vector  $V_k$  whose components are

$$V_k(i) = V_{k-1}(i) + \alpha \rho_k^{(1-a_k)}(d_k(i) - V_{k-1}(i))$$
(I)

Using vector notation, we can express this as

$$V_k = V_{k-1} + \alpha \rho_k^{(1-a_k)} (D_k - V_{k-1})$$
(Ib)

In the above,  $\alpha \in [0,1]$  is a basic learning rate. The term  $\rho_k$  is the compatibility of the observation  $D_k$  with the current belief  $V_{k-1}$ . This is obtained as

$$\rho_k = \left(1 - \frac{1}{n} \sum_{i=1}^n |d_k(i) - V_{k-1}(i)|\right)$$

It is noted that  $\rho_k \in [0,1]$ . The larger  $\rho_k$ , the more compatible the observation is with the current belief. One important feature that needs to be pointed out is the role that the

multidimensionality plays in the determination of the compatibility  $\rho_k$ . The system is essentially looking at the individual compatibilities as expressed by  $|d_k(i) - V_{k-1}(i)|$  to determine the overall compatibility  $\rho_k$ . A particularly notable case is where for most of the individual variables we have good compatibility between the observation and the belief but a few are not in agreement. In this case, since there is a preponderance of agreement we shall get a high value for  $\rho_k$  and the system is open to learning. Here, then, the system has received a piece of data that it feels is a reliable observation based on its current belief and is therefore open to accept it and learn from the smaller part of the observation, which is not what it believes. We shall call such observations *kindred* observations. The agent's openness to kindred observations plays an important part in allowing PL-based systems to rapidly learn in an incremental fashion [15].

The term  $a_k$ , also lying in the unit interval, is called the *arousal rate*. This is obtained by processing the compatibility using the formula

$$a_k = (1 - \beta)a_{k-1} + \beta(1 - \rho_k) \tag{II}$$

Here  $\beta \in [0,1]$  is a learning rate. As pointed out in [14],  $\beta$  is generally less than  $\alpha$ . We see that  $a_k$  is essentially an estimate of the negation, one minus, the compatibility. Here we note that a low arousal rate, small values for  $a_k$ , is an indication of a good correspondence between the agent's belief and the external environment. In this case, the agent has confidence in the correctness of its belief system. On the other hand, a value for  $a_k$  closer to one arouses the agent to the fact that there appears to be some inconsistency between what it believes and the external environment.

We see from equation (Ib) that when  $a_k \approx 0$ , the learning from the current observation  $D_k$  is strongly modulated by  $\rho_k$ , the compatibility of  $D_k$  with the current belief  $V_{k-1}$ . If the compatibility  $\rho_k$  is high, we learn at a rate close to  $\alpha$ ; if the compatibility  $\rho_k$  is low, we do not learn from the current observation. On the other hand, when  $a_k \approx 1$ , and therefore the agent is concerned about the correctness of its belief, the term  $\rho_k^{(1-a_k)} \approx 1$  is independent of the value  $\rho_k$  and hence the agent is not restrained by its current belief from looking at all observations.

We note that the updation algorithm (I) is closely related to the classic *Widrow-Hoff learning rule* [16].

$$V_k(i) = V_{k-1}(i) + \alpha(d_k(i) - V_{k-1}(i))$$
(W-H)

The basic difference between (I) and (W-H) is the inclusion of the term  $\alpha \rho_k^{(1-a_k)}$ instead of simple  $\alpha$ . This results in a fundamental distinction between the classic leaning model (W-H) and the PL version. In the classic case, the learning rate  $\alpha$  is generally made small in order to keep the system from radically responding to erroneous or outlier observations. In the PL version, the basic learning rate  $\alpha$  can be made large because the effect of any observation incompatible with the current belief is repressed by the term  $\rho_k^{(1-a_k)}$  when  $a_k$  is small. This effectively means that in these PL systems, if a relatively good model of the external environment is obtained, the system can very rapidly tune itself and converge. On the other hand, the classic model has a much slower rate of convergence. An often-used strategy when using the classic learning algorithm (I) is to let  $\alpha$  be a function of the number of observations k. In particular, one lets  $\alpha$  be large for low values of k and then lets it be smaller as k increases. In the PLP, it would seem that here we should treat the learning parameter  $\beta$  in a similar manner.

What we emphasize here is that we determined the acceptability of an observation by using the pointwise compatibilities of the features of the observation with the agent's current belief about the values of the features. More complex formulations for the determination of acceptability can be used within the PLP. For example, in many real learning situations the internal consistency of the features associated with an observation (experience) plays a crucial role in determining the credibility of an observation. The knowledge of what constitutes internal consistency, of course, resides in the current belief systems of the learning agent. When a person tells us a story about some experience he had, a crucial role in our determining whether to believe him is played by our determination of the internal consistency of the story. At a more formal level, in situations in which we are trying to learn functional forms from data, it would appear that the internal consistency of the data in the observation could help in judging whether an observation should be used for learning.

# **1.9 INCLUDING CREDIBILITY OF THE LEARNING SOURCE**

Let us consider the nature of a learning experience. Generally, a learning experience can be seen to consist of two components. The first is the *content of the experience*; we have been dealing with this in the preceding. The second is the *source of the content*. Information about both these components is contained in the agent's current belief system.

In order to decide on the degree of acceptability of a learning experience and its subsequent role in updating the current belief, a participating learning-based system must determine two quantities. The first is the *compatibility of the content* of the experience with the system's current belief system. The second is *the credibility of the source*. The information needed to perform these calculations is contained in the agent's current belief system. A point we want to emphasize is that information about the source credibility is also part of the belief structure of a PL agent in a similar way as information about the content. That is, the concept of credibility of source is essentially a measure of the congruency of the current observation's source with the agent's belief of what are good sources.

Generally, compatible content is allowed into the system and is more valued if it is from a credible source rather than a noncredible source. Incompatible content is generally blocked, and more strongly blocked from a noncredible source than from a credible source.

As we have pointed out, a learning experience consists of content as well as an indication of its source. We noted that the credibility of the source should play a role in the learning process. In the model previously presented we paid no attention to the source. We were implicitly assuming that the source had complete credibility. Let us now consider including information about the source credibility in our learning model based on the participatory learning paradigm.

Here we shall begin to look at ways of including source credibility in the learning process. We assume we have a collection of possible sources of content,  $S = \{S_1, ..., S_n\}$ . In this situation, we associate with each source a value  $C(j) \in [0,1]$  indicating the agent's current belief about the credibility of content provided by source  $S_j$ . This information is stored in the belief system of the agent doing the learning. We can denote this as a vector **C**.

Consider that now we are in a situation in which the *k*th observation is provided by source  $S_j$ . We now provide a form for our updation algorithm (I) that takes into account the learner's perceived credibility of this source, C(j). Here

$$V_{k} = V_{k-1} + \alpha C(j)\rho_{k}^{(1-a_{k})}(d_{k}(i) - V_{k-1}(i))$$
(Ic)

We see that a source with zero credibility has no effect,  $V_k = V_{k-1}$ . On the other hand, in the case when C(j) = 1, we get our original model.

The process of updating the arousal,  $a_K$ , associated with the model is also affected by the credibility of the source. In these cases, we must modify (II) to take into account the credibility of the source. We first note that the original arousal-level updation algorithm was

$$a_k = (1 - \beta)a_{k-1} + \beta(1 - \rho_k)$$

or equivalently

$$a_k = a_{k-1} + \beta(\bar{\rho}_k - a_{k-1})$$

where

$$\bar{\rho}_k = 1 - \rho_k$$

Modifying this to take into account the credibility of the source, we get

$$a_k = a_{k-1} + C(j)\beta(\bar{\rho}_k - a_{k-1}) \tag{IIc}$$

Again we see that sources with zero credibility do not affect our degree of arousal; on the other hand, for those with credibility one, we get the original model.

We note that formula (IIc) can be expressed as

$$a_k = (1 - C(j)\beta)a_{k-1} + C(j)\beta\bar{\rho}_k$$

Implicit in our inclusion of the credibility of the sources has been an assumption that a source's credibility, C(j), is fixed. More generally, the agent using the PLP will learn and update its belief about the credibility of the sources as a result of its learning experiences in a manner similar to the way it learns content. Before providing this generalization, we shall more usefully denote the credibility of source  $S_j$  as  $C_k(j)$ , indicating the credibility of  $S_j$  after the *k*th learning experience.

In the following, we shall provide a formula to allow the updation of the source credibility. In expressing this formulation we shall find it convenient to use the term  $M_{ik}$ 

defined such that  $M_{jk} = 1$  if  $S_j$  is the source of the *k*th observation, and  $M_{jk} = 0$  if  $S_j$  is not the source of the *k*th experience. Using this notation we express our algorithm for modifying the source credibility as

$$C_{k}(j) = C_{k-1}(j) + M_{jk}\lambda\bar{a}_{k-1}(\rho_{k} - C_{k-1})$$
(III)

where  $\bar{a}_{k-1} = 1 - a_{k-1}$  and  $\lambda \in [0,1]$  is a base learning rate. We note that if  $S_j$  is not the source of the *k*th learning experience, then  $M_{jk} = 0$  and  $C_k(j) = C_{k-1}(j)$ . Thus, in this case we do not make any changes in our credibility. If  $M_{jk} = 1$ , then

$$C_k(j) = C_{k-1}(j) + \lambda \bar{a}_{k-1}(\rho_k - C_{k-1}(j)) = (1 - \lambda \bar{a}_{k-1})C_{k-1}(j) + \lambda \bar{a}_{k-1}(\rho_k)$$

Here we see  $\lambda \bar{a}_{k-1}$  is an effective learning rate; it is the product of our base learning rate and the negation of the arousal level. Essentially  $\bar{a}_{k-1}$  is the confidence in the current model. Thus, if  $\bar{a}_{k-1} = 1$ , then  $C_k(j) = C_{k-1}(j) + \lambda \bar{a}_{k-1}(\rho_k - C_{k-1}(j))$ . Here  $\rho_K$  is being used to calculate the current degree of performance of the source.

We make some observations about the learning-updation model for the  $C_k(j)$ , First, we note that if there exists only one source  $S = \{S_1\}$ , then we will get  $C_k(1) = 1 - a_k$ . That is, the credibility of the agent is the negation of the arousal level. Since  $1 - a_K$  is essentially the credibility of the model, we see that the credibility of a sole source and the credibility of the resulting model are equivalent.

We note that in the source credibility updation algorithm (III) we can have different base learning rates for each source. Thus we can have  $\lambda(j)$  instead of  $\lambda$ , where  $\lambda(j)$  is the base rate of learning the credibility of the *j*th agent. Further, if  $\lambda(j) = 0$ , then  $C_k(j) = C_{k-1}(j)$  for all *k*. This implies a fixed assigned credibility for the *j*th source.

# 1.10 TYPES OF LEARNING SOURCES

In the following, we shall try to identify and classify some types of sources of content that are involved in learning experiences. The first type of source is *direct sensory experiences*. These are related to observations we make with our own sensory organs. Examples of this are seeing an auto accident, being at a baseball game and watching the Yankees win, smelling alcohol on somebody's breath, or hearing John tell Mary "I love you."

The second type we shall denote as an *authority*. Examples of learning experiences from an authority are being told something by another person, reading something in a book or obtaining it from the Internet, and hearing news on the radio or seeing it on TV. The basic idea is that the contents have been processed by some other *cognitive* agent. For these types of sources the content can be subject to "interpretation" by the processing agent.

A third source of content is *electromechanical sensor* data. This is data obtained through some electromechanical device such as the speedometer on your car, a thermometer, or the types of screens air traffic controllers use. Numerous other types of devices can be listed. Here the contents have been processed by a neutral *physical* device.

Another source of content is what we shall call *reflection*. Here we mean the conscious rational manipulation of information already in an agent's belief system that brings to the agent's awareness knowledge that is implicit in the current belief system. Deduction, induction, and reasoning are examples of this source. It can be seen as a kind of reorganization of knowledge in the agent's belief system. An interesting example occurs when the agent becomes aware of a conflict in his system. This source of content here is clearly distinct from the preceding three in that in the first three the source of the new content comes from outside the agent. In the case of reflection, the source of the content is internal to the agent. An additional aspect of what we have in mind with this reflective source of information that we want to emphasize without getting too deep into philosophical issues is the conscious *rational* aspect of this process.

Another source of content for our learning experiences is what we shall denote as *mystic*. Examples of this would be information coming from the subconscious, dreams, hallucinations, being told by God, and what some people call *gut feeling*. What we want to point out here is that for the most part this can be considered as *internally sourced* information; however, for some of these, such as being told by God, it is not clear whether it is from external or internal sources. Further, for some people, this type of content can be considered as "empty." However, for others, this is a valid type of learning experience. This category of learning is becoming an increasingly important one in our time since one objective of many security systems is to try to understand, predict, and manipulate the actions of terrorists, many of whom are religious fundamentalists who construct their belief system using this type of source.

We feel that various types of agents can be modeled by formalizing the credibility that they attach to these different categories of sources of learning experiences.

# 1.11 CONCLUSION

In this work we described two instruments for introducing evolutionary behavior into intelligent systems. The first was the hierarchical prioritized structure (HPS) and the second was the participatory learning paradigm (PLP). We saw that underlying both these instruments is a type of nonlinear aggregation operation that is adjudicating between knowledge held at different levels. Central to this type of aggregation is a process in which the privileged knowledge is deciding on the allowable influence of the less-favored knowledge.

# 1.12 REFERENCES

- 1. Darwin, C., The Origin of the Species, London, 1859.
- 2. Yager, R. R., "On a Hierarchical Structure for Fuzzy Modeling and Control." *IEEE Trans. on Systems, Man and Cybernetics* 23, 1189–1197, 1993.
- 3. Yager, R. R., "On the Construction of Hierarchical Fuzzy Systems Models." *IEEE Trans. on Systems, Man and Cybernetics Part C: Applications and Reviews* 28, 55–66, 1998.

- Rozich, R., T. Ioerger, R. R., Yager, "FURL—A Theory Revision Approach to Learning Fuzzy Rules." *IEEE World Congress on Computational Intelligence*, Honolulu, Hawaii: *Proceedings of Conference on Fuzzy Systems*, 791–796, 2002.
- Yager, R. R., "Organizing Information Using a Hierarchical Fuzzy Model." In *Systematic Organisation of Information in Fuzzy Systems* (P., Melo-Pinto, H., Teodrescu, T., Fukuda, eds.), NATO Science Series. IOS Press: Amsterdam, 2003, pp. 53–69.
- Klement, E. P., R. Mesiar, E., Pap, *Triangular Norms*. Dordrecht, Kluwer Academic Publishers, 2000.
- 7. Yager, R. R., D. P., Filev, "Approximate Clustering via the Mountain Method." *IEEE Trans. on Systems, Man and Cybernetics* 24, 1279–1284, 1994.
- 8. Yager, R. R., D. P., Filev, "Generation of Fuzzy Rules by Mountain Clustering," *Journal of Intelligent and Fuzzy Systems* 2, 1994, pp. 209–219.
- Rickard, J. T., R. R. Yager, W., Miller, "Mountain Clustering on Non-uniform Grids Using p-Trees," *Fuzzy Optimization and Decision Making* 4, 2005, pp. 87–102.
- Yager, R. R., D. P., Filev, "Summarizing Data Using a Similarity-Based Mountain Method," Information Sciences 178, 2008, pp. 816–826.
- 11. Quine, M.V.O., "Two Dogmas of Empiricism," Philosophical Review 60, 1951, pp. 20-43.
- 12. Quine, M. V. O., From a Logical Point of View. Cambridge: Harvard Press, 1953.
- 13. Kuhn, T. S., The Structure of Scientific Revolutions. University of Chicago Press: 1962.
- 14. Yager, R. R., "A Model of Participatory Learning." *IEEE Trans. on Systems, Man and Cybernetics* 20, 1990, pp. 1229–1234.
- Yager, R. R., D. Z., Zhang, "Effective Suggestion Based on Participatory Learning," *Expert* Systems: Research & Applications 7, 1994, pp. 423–432.
- 16. Widrow, B., M. E., Hoff, "Adaptive Switching Circuits," *IRE Western Electric Show and Convention Record*, Part 4, 1960, pp. 96–104.