3. Learning Modeling and Control

3.1 Need for Learning

The inflexibility of many industrial robots has been one of the principal reasons for the reduction in the growth rate of robotic industrial applications [Albus, 1990].

By introducing learning elements into the control systems, the plants will become more flexible and better able to deal with complex, real-world environments. Learning is an integral part of any IC system and exists at many levels of abstraction. In the dynamic control techniques, learning algorithms have been studied in the adaptive control field for many years [Åström and Wittenmark, 1989, Narendra and Thathacher, 1989], and these have been complemented by research into adaptive neural and fuzzy networks [Wang, 1994]. Higher in the IC architecture, learning can be as simple as internally updating its world model, or complex learning/exploring systems may be used [Sutton, 1990]. Each task requires an appropriate learning system, as the problem structures are generally different. Therefore, learning systems should be aimed at particular problems and should always use the maximum amount of relevant *a priori* information.

Within the specific context of dynamic processes, learning may be required for the following reasons:

- <u>prior</u> knowledge about the plant's structure is unavailable or only partially known;
- <u>time-varying</u> plant;
- <u>partially known</u> or time-varying operational environment;
- <u>improve</u> the performance of the plant over a wide range of operating conditions;
- <u>increased</u> flexibility; and
- <u>reduced</u> design costs.

It should be noted that other techniques such as gain scheduling can provide similar benefits if the problem is structured appropriately.

Learning control techniques generally use a basic model that is inherently nonlinear. This is an important point, because it enables *global* plant models to be constructed rather than the locally linear models used in adaptive control. Therefore, there is no need for the continuous adaptation which is necessary to compensate for changing operating points once a satisfactory model has been learned.

3.2 Learning and Adaptation

To adapt is "to change a behaviour to conform to new circumstances" [Åström and Wittenmark, 1989], whereas learning generally implies a gaining or transfer of knowledge. Adaptive control techniques have been well developed over the past thirty years and many convergence (and stability) rules and theories have been developed for linear plants, and under certain circumstances these can also be applied to various nonlinear processes as well. Learning algorithms are generally aimed at ill-defined processes and use heuristics, for instance to construct rule-base systems.

3.3 Learning Algorithms

Many different algorithms can be used within learning control schemes and their knowledge structures generally reflect the type of application. The three most popular types of learning algorithms are currently artificial neural networks [Miller *et al.*, 1990], fuzzy logic [Wang, 1994], and expert systems [Hunt, 1992]. These categories are not distinct as there exist strong relationships between fuzzy and expert systems which incorporate uncertainty, as well as between fuzzy and some associative memory neural systems and some expert systems can be implemented within a neural network architecture.

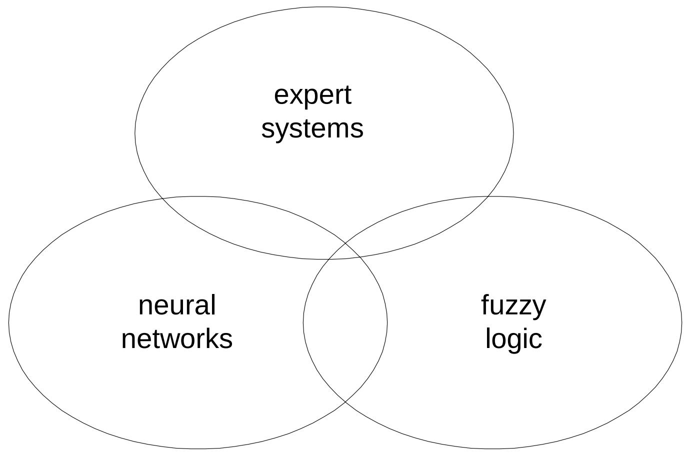

Fig. 3.1. Three learning algorithms with inter-relationships.

Central to the problem of learning is the ability of the algorithm to *generalize* correctly from a limited number of training samples, which means that the algorithm must *interporate* and locally *extrapolate* with sufficient accuracy.

Most learning algorithms can be classified according to their *modeling*, *learning*, and *validation* properties. The modeling capabilities of an algorithm determine the range of nonlinear functions which it can reproduce exactly and any implicit smoothness assumptions made by the network. The learning rule used does not generally affect the underlying modeling capabilities of the algorithm, although the chosen model structure influences its rate of convergence and can even determine the type of learning rule that should be used. Finally, any practical application of a learning algorithm requires convergence, stability and correctness tests which can verify what is being learned. If an algorithm learns, it also forgets and it must be verified that the behavior being stored is desirable.