[Tam89] Tam D.C. and Perkel D.H. (1989). Quantitative modeling of synaptic plasticity. *Computational Models of Learning in Simple Neural Systems*, Hawkins R.D. & Bower G.H. eds. Academic Press. pp. 1-30.

[Thom86] Thompson R.F. (1986). The neurobiology of learning and memory. *Science*, **233**, pp. 941-947.

[Waib89] Waibel A., Sawai H. and Shikano K. (1989). Modularity and scaling in large phonemic neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-37, pp. 1888-1898.

[Whit89] Whitley D. and Hanson T. (1989). Optimizing neural networks using faster, more accurate genetic search. *Proc. Third International Conference on Genetic Algorithms*, J.D. Shafer ed., Morgan Kaufmann, pp. 391-396.

[Will89] William R.J. and Zipser D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Networks*, vol. 1, no. 2, pp. 270-280.

[Wins90] Winson J. (1990). The meaning of dreams. *Scientific American*, Nov. 1990, pp. 86-96.

[Lipp89] Lippman R. (1989). Review of neural networks for speech recognition. *Neural Computation* vol. 1, no. 4., pp. 1-38.

[Mel90] Mel B. and Koch C. (1990). Sigma-pi learning: On radial basis functions and cortical associative learning. In D. Touretzky, ed., *Advances in Neural Information Processing 2*, pp. 474-481, Morgan Kaufmann.

[Mins69] Minsky M., Papert S. (1969). *Perceptrons*. Cambridge, MA. MIT Press.

[Pavl88] Pavlides C., Greenstein J., Grudman M. and Winson J. (1988). Long-term potentiation in the dendate gyrus is induced preferentially on the positive phase of Θ-rhythm. *Brain Research*, vol. 439, pp. 383-387.

[Pear89] Pearlmutter B.A. (1989). Learning state space trajectories in recurrent neural networks. *Neural Computation* vol. 1, no. 2, pp. 263-269.

[Pete90] Peterson C. (1990). Parallel distributed approaches to combinatorial optimization: benchmark studies on traveling salesman problem. *Neural Computation*, vol. 2, no. 3, pp. 261-269.

[Pogg89] Poggio T. and Girosi F. (1989). A theory of networks for approximation and learning. MIT AI Memo No. 1140.

[Pome89] Pomerleau D.A. (1989). ALVINN: an autonomous land vehicle in a neural network. *Advances in Neural Information Processing Systems 1*, ed. D.S. Touretzky, Morgan Kaufmann, pp. 305-313.

[Rose62] Rosenblatt F.E. (1962). *Principles of Neurodynamics*. New York. Spartan.

[Rume86a] Rumelhart D.E., McClelland J.L. (eds.) (1986). *Parallel Distributed Processing*, volume 1. Bradford Books, MIT Press.

[Rume86b] Rumelhart D.E., Hinton G.E. and Williams R.J. (1986). Learning internal representations by error propagation. Chapter 8 of *Parallel Distributed Processing*, volume 1. Rumelhart D.E. and McClelland J.L. (eds.), Bradford Books, MIT Press.

TR-CMU-CS-84-119, Carnegie-Mellon University, Dept. of Computer Science.

[Hint89] Hinton G.D. (1987). Connectionist Learning Procedures. *Artificial Intelligence*, vol. 40, Sept. 89, pp. 185-234.

[Holl75] Holland J. (1975). *Adaptation in Natural and Artificial Systems.* University of Michigan Press.

[Hopf83] Hopfield J.J., Feinstein D. and Palmer R. (1983). Unlearning has stabilizing effects in collective memories. *Nature 304*, pp. 158-159.

[Hopf84] Hopfield J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. U.S.A.*, **81**, pp. 3088-3092.

[Hopf89] Hopfield J.J. and Tank D.W. (1989). Neural architectures and biophysics for sequence recognition. In *Neural Models of plasticity*, J.H. Byrne & W.O. Berry, eds., pp. 363-377.

[Jaco88] Jacobs R.A. (1988). Increased rates of convergence through learning rate adaptation. *Neural Networks*, **1**, pp.295-307.

[Jord90] Jordan M.I. (1990). Motor learning and the degrees of freedom problem. In M. Jeannerod, (ed.), *Attention and Performance*, XIII, Hillsdale, NJ: Erlbaum.

[Klop89] Klopf A.H. (1989). Classical conditioning phenomena predicted by a drive-reinforcement model of neuronal function. In *Neural Models of plasticity*, J.H. Byrne & W.O. Berry, eds., pp. 104-132.

[Lars86] Larson J., Wong D. and Lynch G. (1986). Patterned stimulation at the theta frequency is optimal for induction of long-term potentiation. *Brain Research*, **368**, pp. 7-35.

[LeCu89] Le Cun Y., Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W. and Jackel L.D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, vol. 1, no. 4, pp.541-551.

[Davi89] Davis L. (1989). Mapping neural networks into classifier systems. *Proc. Third International Conference on Genetic Algorithms*, J.D. Shafer ed., Morgan Kaufmann, pp. 375-378.

[Done89] Donegan N.H., Gluck M.A. and Thompson R.F. (1989). Integrating behavioral and biological models of classical conditioning. *Computational Models of Learning in Simple Neural Systems*, Hawkins R.D. & Bower G.H. eds. Academic Press. pp. 109-156.

[Gall89] Galland C.C. and Hinton G.E. (1989). Deterministic learning in networks with asymmetric connectivity. Tech. Rep. CRG-TR-89-6.

[Gard87] Gardner D. (1987). Synaptic diversity characterizes Biological Neural Networks. *Proc. IEEE First International Conference on Neural Networks*, San Diego, CA, pp. IV-17 - IV-22.

[Gold88] Goldberg D. (1988). *Genetic Algorithms in Machine Learning, Optimization, and Search.* Addison-Wesley.

[Hawk89a] Hawkins R.D., Bower G.H. (eds.) (1989). *Computational Models of Learning in Simple Neural Systems.* Academic Press.

[Hawk89b] Hawkins R.D. (1989). A biologically based computational model for several simple forms of learning. *Computational Models of Learning in Simple Neural Systems*, Hawkins R.D. & Bower G.H. eds. Academic Press. pp. 65-108.

[Hawk89c] Hawkins R.D. (1989). A simple circuit model for higher-order features of classical conditioning. In *Neural Models of plasticity*, J.H. Byrne & W.O. Berry, eds., pp. 74-93.

[Hawk83] Hawkins R.D., Abrams T.W., Carew T.J. and Kandel E.R. (1983). A cellular mechanism of classical conditioning in Aplysia: Activity-dependent amplification of presynaptic facilitation. *Science*, **219**, pp. 400-404.

[Hint84] Hinton G.E., Sejnowski T.J. and Ackley D.H. (1984). Boltzmann machines: Constraint satisfaction networks that learn. Tech. Rep.

References

[Bair90] Baird B. (1990). Learning with synaptic nonlinearities in a coupled oscillator model of olfactory cortex. To appear in *Analysis and Modeling of Neural Systems*, F.H. Eeckerman ed.

[Bart91] Bartha G.T., Thompson R.F., and Gluck M.A. (1991). Sensorimotor learning and the cerebellum. To appear in *Visual Structures and Integrating Functions*, M. Arbib & J. Ewert eds, Springer-Verlag.

[Beck89] Becker, S. and Le Cun, Y. (1989). Improving the convergence of back-propagation learning with second-order methods. In Touretzky, Hinton and Sejnowski eds., *Proc. of the 1988 Connectionist Summer School*, pp. 29-37, San Mateo. Morgan Kaufmann.

[Beng89] Bengio Y., Cardin R., De Mori R. & Merlo E. (1989). Programmable execution of multi-layered networks for automatic speech recognition. *Communications of the Association for Computing Machinery*, vol. 32, no. 2, pp. 195-199.

[Bott90] Bottou L., Fogelman Soulie F., Blanchet P. and Lienard J.S. 1990. Speaker-independent isolated digit recognition: multilayer perceptrons *vs* dynamic time warping. *Neural Networks* 3 (4) : 453-465.

[Byrn87] Byrne J.H. (1987). Cellular analysis of associative learning. *Physiological Review*, **67**, pp. 329-439.

[Byrn89] Byrne J.H., Gingrich K.J. and Baxter D.A. (1989). Computational capabilities of single neurons: relationship to simple forms of associative and nonassociative learning in *Aplysia. Computational Models of Learning in Simple Neural Systems*, Hawkins R.D. & Bower G.H. eds. Academic Press. pp. 31-63.

[Crow89] Crow T. (1989). Associative learning, memory and neuromodulation in *Hermissenda.* In *Neural Models of plasticity*, J.H. Byrne & W.O. Berry, eds., pp. 1-21.

that depends on the behavior of variables, such as a postsynaptic potential, which have been observed physiologically. The parameters θ of the learning function(s) $\Delta w(\theta)$ are adapted in order for those variables to fit as well as possible the observed data for those neural and behavioral phenomena.

A more complicated architecture is the sensorimotor circuit responsible for the classically conditioned, rabbit eyelid closure response, studied for example in [Thom86, Done89, Bart91]. Much less is known about cortical architecture and function of components of the brain responsible for learning of complex tasks, such as learning to recognize sequences. However, it is possible to design models that are more plausible than the connectionist speech recognition recognition models (*e.g.* [Beng89, Waib89]). For example, the model for sequence and speech recognition proposed in [Hopf89] attempts to respect the electrophysiology and anatomy of mammalian brains.

5 Conclusion

We present an original approach to neural modeling, based on the idea of searching *with learning methods* for a parametric synaptic learning rule that is biologically plausible, as well as yielding networks that can learn to perform difficult tasks. The networks architecture, as well as the learning function, can be designed with constraints derived from biological considerations. The method presented may help to bridge the gap between two approaches in neural networks research: mathematically-derived (*e.g.*, connectionist models [Rume86a]) and biologically-faithful models (*e.g.*, [Hawk89a]). This is not only important to improve biological modeling of brain function and learning, but it may also help to discover better artificial neural networks algorithms. By constraining the search for these algorithms to biologically plausible models, we increase the probability that these models will perform and generalize well, when facing tasks on which the brain performs well. It should be noted that the method of learning the learning rule can be viewed as a recursive process. If the resulting learning rule performs better than the original learning method used to learn that rule (*e.g.* gradient descent or genetic algorithms), then that learning rule should be used in a further iteration as a learning method in order to improve itself.

An example of a biological model that could be embedded as a module of $\Delta w(\theta)$ is the generalized Hebbian rule proposed in [Done89]:

$$\Delta w_i = \epsilon \, y_{pre(i)} \, k \tag{11}$$

where ϵ is a positive valued learning rate parameter, $y_{pre(i)}$ is the presynaptic activity (at synapse i), and k is a reinforcing signal. That signal can either be the postsynaptic potential $x_{post(i)}$ or the activity of a special training signal, for example a neurofacilitator signal $y_{modul(i)}$, as in the case of the pairing-specific presynaptic facilitation observed in *Aplysia* [Hawk83], or in the case of neuromodulators that may be related to associative learning in *Hermissenda* [Crow89]. In general, several modulating signals should be allowed, corresponding to multiple neuromodulators, as in Figure 1.

Another example of a synaptic modification model that could inspire the design of $\Delta w(\theta)$ is the one proposed in [Klop89]. In this model, synaptic changes depend upon *changes* in pre- and post- synaptic activity, instead of depending upon the absolute levels of these variables. This could be implemented in the design of Figure 1, by allowing multiple delays (and/or recurrence) in the subfunction modules.

More physiologically detailed models of synaptic plasticity can also be considered. For example, it would be interesting to explore the advantages of including more detailed models, describing molecular mechanisms involved in synaptic plasticity, such as the models described in [Tam89] or [Byrn89].

The synaptic modification rule by itself is not enough to ensure that the network will be able to perform the target tasks. It is also necessary to investigate various architectural alternatives, as discussed in section 2. For example, it seems clear that a recurrent architecture is needed, in particular to solve the credit-assignment problem.

Well-studied network architectures can be used to constrain the synaptic learning rule to be biologically plausible. The *Aplysia* circuit models described in [Hawk89c] demonstrate various neural and behavioral phenomena, such as habituation, sensitization, classical conditioning, extinction, second-order conditioning, blocking and latent inhibition. Some of the networks used to constrain $\Delta w(\theta)$ can be trained so that they demonstrate those phenomena (see Section 3). This can be obtained by using a learning criteria

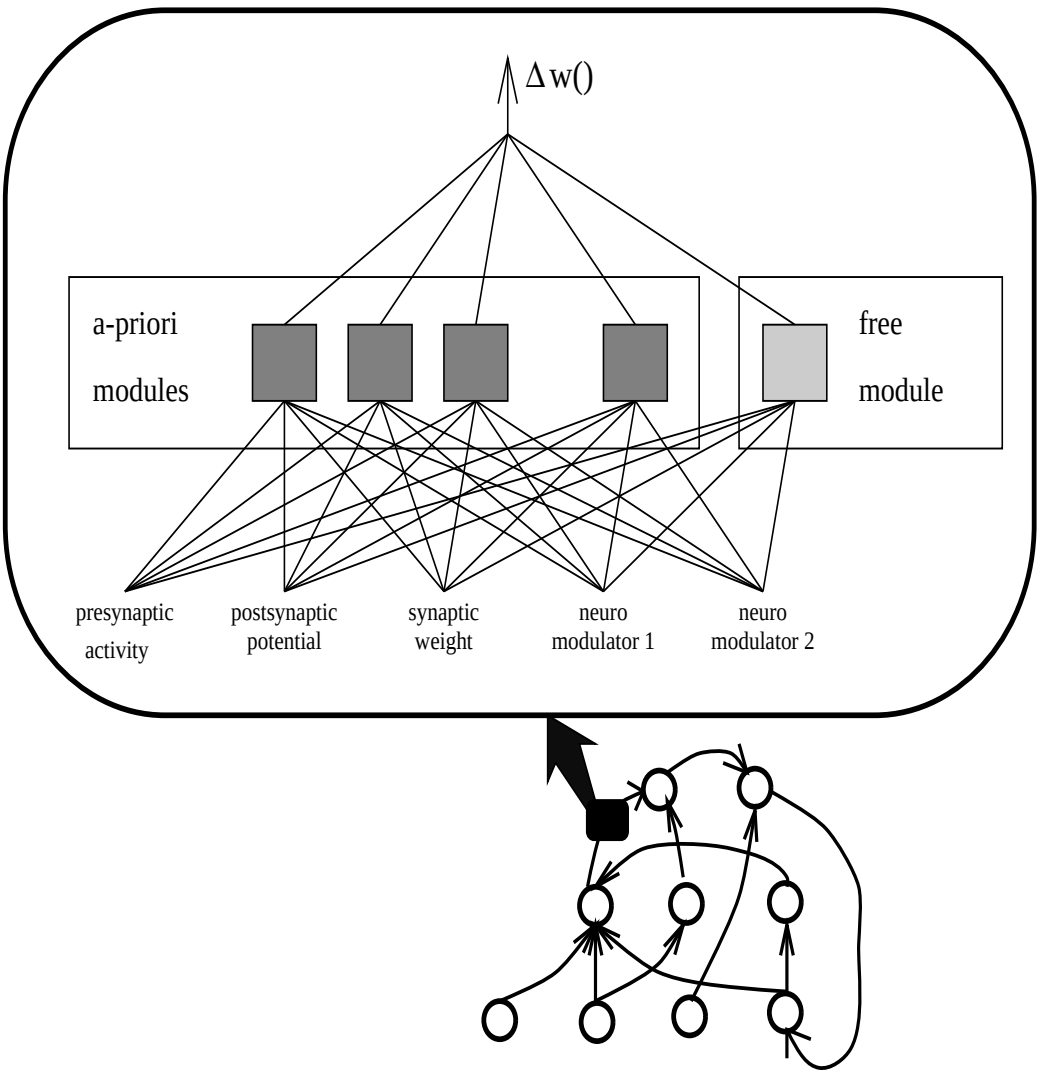

Figure 1: *Proposed structure of the synaptic learning function $\Delta w(\theta)$: it contains multiple a-priori modules, each representing a different known or hypothesized synaptic plasticity mechanism. A free module may be added to allow the resulting $\Delta w(\theta)$ function to be more powerful.*

with respect to a parameter θ_j:

$$\frac{\partial w_i(t+1)}{\partial \theta_j} = \alpha \dot{g}(h(\beta w_i(t)) + \Delta w_i(t)) \, (\dot{h}(\beta w_i(t)) \beta \frac{\partial w_i(t)}{\partial \theta_j} + \frac{\partial \Delta w_i(t)}{\partial \theta_j}) \quad (10)$$

where $\dot{g}()$ and $\dot{h}()$ denote the first derivatives of $g()$ and $h()$ respectively. The parameters α and β could themselves be tuned with a learning algorithm (similar to learning algorithms in Section 3). These parameters could be different for different types of synapses or neurons (allowing an architecture in which inhibitory neurons are organized differently from excitatory neurons).

A biologically plausible implementation of excitatory and inhibitory synapses should allow a different parameter set θ for the $\Delta w(\theta)$ function of excitatory and inhibitory synapses. A generalization of this idea is to allow multiple $\Delta w(\theta)$ functions for the diverse types of synapses that are observed. Various types of synapses, neurotransmitter, pre-, epi- and post-synaptic mechanisms were for example modeled in *Aplysia* [Gard87].

Another way to use biological knowledge is to bootstrap the function $\Delta w(\theta)$ so that it initially has access to a set of *a-priori* subfunctions equivalent to known or hypothesized synaptic modification mechanisms (see Figure 1). These *a-priori* subfunctions may themselves be parametrized functions (which is often the case of biological models). The optimization method described in section 3 can be used to search for an optimal combination of these subfunctions, as well as, to tune their parameters. In addition, a *free* subfunction could be included in the system that computes $\Delta w(\theta)$. This free subfunction is initialized with random parameters (like for a backpropagation network initialization) and it is used to perform computations which the *a-priori* subfunctions and their combination cannot provide, but which can help improving the satisfaction of constraints imposed on $\Delta w(\theta)$. This bootstrapping of $\Delta w(\theta)$ ensures that the networks will at least be able to learn, in the situations in which the *a-priori* subfunctions where shown to work. If no free subfunction is integrated in $\Delta w(\theta)$, then one constrains $\Delta w(\theta)$ to use only the synaptic modification mechanisms provided in the *a-priori* subfunctions. A modular approach similar to the one proposed here, which integrates several pre-trained or already known modules with a free module, was successfully used in [Waib89] to reduce the training time.

tic modification function) while the networks are learning, rather than waiting for all the networks to have converged. Once a network has converged, its learning can be restarted, using the current (improved) synaptic modification function. An interesting idea is to apply such an on-line, joint optimization to the genetic algorithm. This is equivalent to using a noisy evaluation function, but moving much faster in the space of the parameters θ.

4 Biological Constraints

Biological plausibility constraints can be imposed on the learning rule. For example the rule can be designed in such a way that it uses only information which is local in time and in space: its inputs are locally measurable quantities, sampled in the near past. The rule can have some internal variables which represent a memory of their past values. However, this mechanism should not use an indefinitely large buffer of their past values, as does for example the time-unfolding algorithm for backpropagation recurrent networks [Rume86b].

A biological constraint that is often neglected in connectionist models is that a synapse can either be excitatory or inhibitory. This constraint could be implemented in many ways. An interesting approach is to do so by forcing $\Delta w(\theta)$ to be bounded to a certain range. For example, this can be obtained with a squashing function. Instead of equation (2), the synaptic weight update is then expressed as follows:

$$g(x) = \frac{1}{1 + \exp^{-x}} \tag{7}$$

$$h(x) = g^{-1}(x) = 2 \arg\tanh(2x - 1) \tag{8}$$

$$w_i(t+1) = \alpha\, g(h(\beta\, w_i(t)) + \Delta w_i(t)) \tag{9}$$

where the parameter β controls the abruptness of the squashing, and the parameter α controls the maximum strength, as well as the sign of the synapse (+ = excitatory, - = inhibitory). In that case, instead of equation (6), one would use the following in order to compute the derivative of the weight w_i

brids have been proposed and could be considered here (see for example [Davi89] or [Whit89]). A coding scheme and a set of genetic operators for $\Delta w(\theta)$ could be designed, based for example on those proposed in [Whit89] to improve neural networks with genetic algorithms.

The advantage of genetic algorithms is that they are quite resistant to the problem of local minima in the optimization criteria. However, this is obtained at the cost of improving simultaneously a whole population of solutions instead of just one.

3.3 Methodology

Both these learning methods require the implementation of several neural networks, doing several different tasks. Indeed, it is very important for these tasks to be as varied as possible, because they will constrain the learning power and the generality of the new learning rule.

Some networks should work on complex learning problems such as pattern or speech recognition problems, to enforce efficiency constraints. There should also be tasks that consist in reproducing behavioral phenomena, such as associative conditioning, to reflect behavioral constraints. And finally, some very small networks may be trained to reproduce neurological phenomena such as habituation, recovery, dishabituation and sensitization.

The synaptic modification function could be implemented as a backpropagation [Rume86a] neural network with constraints, inputs and initial features that are derived from biological considerations, *i.e.*, biological knowledge will be used to bootstrap this function and force it to be plausible (see section 4). This network could have feedback and could be driven by differential equations (see [Pear89] for the computation of the gradient in that case).

With the gradient descent method, it has been shown experimentally [Bott90] that modifying the parameters of neural networks continuously (on-line learning) improves convergence time significantly over the batch learning method (modifying the parameters only after all training examples have been shown). Consequently, it would probably be more efficient to update θ (*i.e.*, the synap-

An advantage of the gradient descent approach to learning the synaptic modification rule is that there exists a large body of work on non-linear optimization using gradient descent in a similar situation, namely with the backpropagation algorithm. Another advantage is its firm mathematical grounding. A disadvantage is the weakness of the standard algorithm when facing local minima of the criterion function. Most frequently, the local minima are almost as "good" as the global minima [Rume86b] . Otherwise, several strategies can be considered to cope with the problem. One of them consists in using on-line learning with an adaptive and decreasing learning rate ($\epsilon(t)$ in equation (1)). Another solution is to consider another learning algorithm, or a hybrid of gradient descent with another learning algorithm, such as genetic algorithms.

3.2 Genetic Algorithms

An interesting alternative to gradient based optimization methods is the set of optimization methods called "genetic algorithms" (see [Holl75] and [Gold88]). Genetic algorithms are learning algorithms inspired from several features of biological evolution. They consider a population of solutions to a problem, such as the synaptic modification function, encoded in artificial "chromosomes". Each member of the population is evaluated using an evaluation function, *e.g.*, how well the synaptic modification function represented by the chromosomes improves the optimization criteria. The population undergoes reproduction until a satisfactory performance is attained. During reproduction, one or more "parents" are stochastically chosen to reproduce. This choice favors parents with highest evaluation, *i.e.*, best performance of the evaluation function. Operators are applied to the chromosomes of the parents to produce children that are inserted into the population. Standard operators are mutation (stochastically modifying one piece of information in a chromosome) and crossover (combining corresponding pieces of information from two parent chromosomes). Domain knowledge can be exploited to create operators which improve the efficiency of the optimization procedure [Whit89].

It should be noted that interesting gradient-descent/genetic-algorithm hy-

For example, a biologically plausible synaptic change function could have as local arguments a measure of presynaptic activity (y_{pre}), the postsynaptic potential (x_{post}), the strength of the synapse (w_i) and a measure of the activity of a facilitatory neuron (or of the concentration of a diffusely acting neuromodulator) (y_{modul}) that modulates the synaptic plasticity:

$$\Delta w_i(t) = \Delta w(x_{post(i)}(t), y_{pre(i)}(t-1), w_i(t), y_{modul(i)}(t), \theta) \tag{4}$$

To perform gradient descent on θ one computes the following derivative:

$$\frac{\partial E}{\partial \theta_j} = \sum_i \frac{\partial E}{\partial w_i(t)} \frac{\partial w_i(t)}{\partial \theta_j} \tag{5}$$

where $\frac{\partial w_i(t)}{\partial \theta_j}$ can be computed recursively ($\frac{\partial w_i(0)}{\partial \theta_j} = 0$):

$$\frac{\partial w_i(t)}{\partial \theta_j} = \frac{\partial w_i(t-1)}{\partial \theta_j} + \frac{\partial \Delta w_i(t)}{\partial \theta_j} \tag{6}$$

Hence the approximating function $\Delta w(\theta)$ is chosen so that it is differentiable with respect to the parameters θ_j. When $\frac{\partial E}{\partial \theta_i}$ is computed, one can use equation (1) (or a more efficient variation of gradient descent [Beck89, Jaco88]) to update the parameters θ_i and perform gradient descent in the criterion E.

To avoid that θ be estimated in function of a particular synapse, neuron or network performing a particular task, it is important that

1. the function $\Delta w(\theta)$ and the parameter set θ that defines it be the same for all (or a large number of) synapses, and that

2. θ be estimated simultaneously with a population of networks learning to perform different tasks, (with all networks sharing the same - or a small set of - learning rule $\Delta w(\theta)$).

The networks learn to perform a task by modifying their parameters (synaptic weights w_i) with a synaptic modification function $\Delta w(\theta)$. This function also evolves, but at a slower rate.

We will consider in the next subsections some iterative learning methods that are probably well suited for this task, namely gradient descent and genetic algorithms.

3.1 Gradient Descent

Gradient descent is a particular kind of hill-climbing optimization. It can be applied when the optimization criteria is a function of some parameters, and it is possible to compute the first derivative of this criteria with respect to these parameters. The gradient descent rule says that each parameter should be modified (increased or decreased) in the direction of the gradient. Suppose we want to minimize an error E that depends on parameters θ_i. Then in order to minimize E, the parameters should be updated with

$$\theta_i(t+1) = \theta_i(t) - \epsilon \frac{\partial E(t)}{\partial \theta_i(t)} \tag{1}$$

where ϵ is a very small but positive real number. There exist numerous variants of this method, to make it faster and more resistant to local minima, for example with the use of second-order information and an adaptable $\epsilon(t)$ [Beck89, Jaco88].

Consider a network of neurons and synapses (with strength w_i), and an optimization criteria E, which is a function of the behavior of the network, and that is to be minimized. Let us assume that $\frac{\partial E}{\partial w_i}$ can be computed. See for example the backpropagation algorithm [Rume86b] to compute that derivative, and [Pear89] for its computation in the case of a continuous recurrent network. Let us consider here for the sake of simplicity a discrete time model (a continuous time model can also be considered, with a similar approach). Let the synaptic weight update at time t be defined as follows:

$$w_i(t+1) = w_i(t) + \Delta w_i(t) \tag{2}$$

and let $\Delta w_i(t)$ be a function of local observable quantities, as well as, of a *set of parameters* θ shared by all (or a lot of) synapses:

$$\Delta w_i(t) = \Delta w(local\ variables, \theta) \tag{3}$$

poorly informative, or limited to a few neurons only. The learning mechanism then has to solve the *credit assignment* problem: how to change the behavior of hidden neurons so that they will perform "better" the next time, where "better" is defined in terms of a learning criterion (for example more positive reinforcement and less negative reinforcement).

In this paper, we do not propose answers to the architectural questions, nor to the reinforcement/supervision problems. Instead, we describe a tool to help search for and refine the synaptic learning rule, given some architectural constraints and a learning criterion.

3 Learning the learning rule

We consider an automatic method of finding and improving, with learning methods, a synaptic modification function that attempts to satisfy several constraints. To this end, we make the following assumptions:

1. The same rule is used in many neurons. A modification of the rule in order to improve it, somewhere in the network, is reflected by the same modification everywhere. This constraint can be relaxed to several rules for several types of neurons or synapses.

2. There exists an input/output mapping that corresponds to the learning rule. This means that the synaptic modification consistently depends on some physical quantities, and not just on chance [3].

3. This mapping can be approximated with a parametric function. Most functions can be approximated, if they present a certain degree of smoothness [Pogg89]. In that case, it is possible to approximate a learning rule by searching for parameters that respect as much as possible the constraints imposed on that rule, such as biological and efficiency constraints.

other parts of the brain, from the same part of the brain in the near future, or from unconditioned stimuli/response.

[3] However, a stochastic process may be involved in the computation of the synaptic change, for example as a variable noise term.

Studies of several forms of learning (*e.g.* classical conditioning and sensitization) suggest that they are related and that complex forms of learning could be obtained by combining mechanisms of elementary forms of learning [Hawk89b]. This helps justifying a model of synaptic change that is a functional combination of several lower level mechanisms (see section 4).

Unfortunately, a better understanding of synaptic plasticity is probably not sufficient to account for learning in biological neural networks, The brain is not a uniform homogeneous network. Different regions of the brain are organized differently, have a different connectivity, as well as different roles. The nervous system includes several types of neurons (*e.g.*, inhibitory and excitatory), as well as several types of synapses. In addition, other factors seem to influence synaptic plasticity (see for example theta rhythm [Lars86] and its relation to learning [Pavl88] and dreaming [Wins90]). Some artificial neural network models require control mechanisms. For example, in the Boltzmann Machine [Hint84] and the Contrasted Hebbian Learning model [Hopf83, Gall89], learning is driven by the difference in neural behavior, between an "awake" phase (inputs fixed by the environment) and a "sleep" [1] phase (free inputs, noisy activity of the neurons). These examples support the contention that a better understanding of the role of brain architecture and organization is important to model neural networks capable of exhibiting complex behavior.

Another important aspect of learning is the definition of *what* the network attempts to learn (learning criterion) and *how* interneurons (hidden neurons in connectionist parlance) learn what functions to perform, in order to satisfy that goal. In many biological models, as well as in artificial neural network models, it is assumed that a direct supervision is provided to the network. In the former models, supervision often takes the form of unconditioned stimuli (US) driving the synaptic change. In the latter models, it is assumed that desired outputs or teacher outputs are available. In a biologically realistic situation, the network has access to a reinforcement (positive or negative feedback), and/or a supervision (teacher[2], unconditioned stimuli) which are

[1] *Awake* and *sleep* are our choice of words, for the purpose of an analogy with a control mechanism for sleep and dream in mammals.

[2] If a teacher is available, it must be assumed that another part of the brain can provide it. A plausible supervision could come from predicting the outcome of processing from

synapse in the network, as in [Will89]).

Some models have recently been proposed that seek to narrow the gap between biological models and connectionist models (see for example [Bair90, Mel90, Bart91]). This is also the goal of the approach presented here. The basic hypothesis of this approach is that it is possible to search with learning algorithms for a synaptic learning rule, and to constrain this rule in order to:

- yield networks that can solve some difficult AI problems,
- respect some aspect of biological knowledge about synaptic mechanisms,
- yield networks that can reproduce some behavioral phenomena.

The proposed method of automatically finding the learning rule relies on the idea of considering the synaptic modification rule as a parametric function, which has *local* inputs, and is the same in many neurons. The parameters that define this function can be optimized with known learning methods. For this optimization, we give particular consideration to gradient descent and genetic algorithms. In both of these cases, estimation of the synaptic modification rule consists of a joint global optimization of the synaptic modification rule, as well as of networks that learn to perform some tasks with this rule. The architecture of the networks, as well as the learning function can be designed with constraints derived from biological considerations. The proposed methodology can be used as a tool to explore the missing pieces of the puzzle of neural networks learning.

2 Learning in Neural Networks

Studies on animal learning increasingly agree that learning involves changes in synaptic connections [Hawk89a, Byrn87]. Synaptic transmission is a complex process, in which changes in efficacy can be due to a number of factors, sometimes acting in concert, both at presynaptic and postsynaptic sites.

1 Introduction

One of the major goals of both biological neural networks modeling and artificial neural networks research is to discover better learning rules, to yield networks that can learn more difficult tasks, such as tasks that the brain can handle. Most researchers now accept that animal learning involves changes of synaptic efficacy [Byrn87]. However, biological knowledge of synaptic plasticity is limited. Because of the complexity of synaptic mechanisms, physiologically detailed models of neural learning are generally limited to simple learning situations. These models abstract some details of neural function, based upon hypotheses about neural function and plasticity.

Alternatively, several researchers have proposed abstract models - called artificial neural networks models, or "connectionist" models [Rume86a] - that are remotely inspired by biological systems. Connectionist models use fairly simple mechanisms for both the neuron operation and the modification of synapses. They can be used to solve some difficult learning problems, including the problem ("hard learning", [Hint89]) of training hidden layers of the network when reinforcement or supervision is only available to some neurons (output units). Early artificial neural networks models [Rose62] were not able to solve that problem [Mins69]. More recent and powerful abstract models have been used for automatic speech recognition (see [Lipp89] for review), handwritten character recognition [LeCu89], optimization [Hopf84, Pete90], robotics [Jord90, Pome89] and other artificial intelligence problems. However, as has been pointed out by Hinton [Hint89], a mathematically derived algorithm such as backpropagation [Rume86b] is not plausible as a biological model for many reasons. Synapses used in backpropagation models permit both forward and backward signals, a functionality not demonstrated to exist in biological synapses. Furthermore, the neurons have to propagate error derivatives through the axon backwards, as well as propagating activity level forwards. In both cases different type of input-output functions are involved, which is even less biologically plausible. In addition, that model is not a real time model since an extension to networks with feedback is biologically unrealistic, either because it is not local in time (it requires a stack of past neural activity, as in [Pear89]) or because it is not local in space (it requires the computation of the mutual influence of each neuron's activity upon each

Learning a synaptic learning rule

Yoshua Bengio [*] Samy Bengio [†]

Technical Report #751,
Département d'Informatique et de Recherche Opérationelle,
Université de Montréal,
November 19, 1990

Abstract

This paper presents an original approach to neural modeling based on the idea of searching, *with learning methods*, for a synaptic learning rule which is biologically plausible, and yields networks that are able to learn to perform difficult tasks. The proposed method of automatically finding the learning rule relies on the idea of considering the synaptic modification rule as a parametric function. This function has *local* inputs and is the same in many neurons. The parameters that define this function can be estimated with known learning methods. For this optimization, we give particular attention to gradient descent and genetic algorithms. In both cases, estimation of this function consists of a joint global optimization of (a) the synaptic modification function, and (b) the networks that are learning to perform some tasks. The proposed methodology can be used as a tool to explore the missing pieces of the puzzle of neural networks learning. Both network architecture, and the learning function can be designed within constraints derived from biological knowledge.

[*]McGill University, School of Computer Science, 3480 University street, Montréal, Qc, Canada, H3A 2A7

[†]Université de Montréal, Département d'Informatique et de Recherche Opérationelle, C.P. 6128, succursale A, Montréal, Qc, Canada, H3C 3J7