

THE CONNECTIONIST MODEL QNET AND ITS COMBINATION WITH GENETIC ALGORITHMS

Philip Van Loocke

ABSTRACT

This paper starts with an exposition of a new connectionist model. The model generalizes more classical fully connected symmetric networks. One of its main characteristics is that the state of a unit is characterized by a vector of amplitudes rather than by a single scalar activation value. The rules that express the evolution of the amplitudes of a particular unit are expressed in terms of the amplitudes of the other units. The chance to observe a particular unit as being active in a particular frequency, however, is proportional with the square of the corresponding amplitude. Because of this as well as some other parallels with quantum theory, the model has been called QNET. QNET operates in different frequencies at once, and the operations in different frequencies interact with each other in a way that is desirable from a cognitive point of view. For instance, when confronted with a problem, QNET can find different solutions in different frequencies. When different solutions are found, this often indicates that more classical networks fail to find a solution at all, since they then converge to a spurious mixture of solutions. We go on to consider how this type of network can be combined with genetic algorithms. We point out that a combination of both methods leads to a technique that integrates the benefits of genetic algorithms with the ones of neural networks.

1. *Introduction*

About a decade ago, symmetric neural nets like the schema model were introduced in order to explain how subconceptual units generate cognitive processes (Rumelhart *et al.*, 1986).¹ Meanwhile, this model and related Hopfield-type models have been applied successfully to pattern recognition problems, and to various problems in which large numbers of soft constraints have to be fulfilled. Other types of symmetric neural nets, like the Boltzmann net, have been used for non-supervised learning. When-

ever a network has symmetric connections, it can be characterized by an energy function. An asynchronous updating rule always leads to a local or global minimal value of this function. Depending on the nature of the problem that is solved, a global minimal value corresponds to a pattern that is recognized, to a solution for a soft constraint problem, to a trajectory for a Travelling Salesman Problem, and so on. The QNET-method can fruitfully be applied to all these cases; the only condition that must be satisfied is that a problem can be solved by a network with symmetric connections.

The structure of this paper is as follows. In section 2, we describe the units of QNET and we specify how they perform their update operations. Section 3 explains the basic properties of QNET. For instance, we show that a single distributed input pattern can lead to a retrieval of different relevant memory patterns in different frequencies. This happens often in cases where a Hopfield-model would converge to a spurious mixture of solutions. Section 4 contains another illustration of these properties. It makes use of a network in which a simple set of distributed sentences is stored. In section 5, we go on to describe how QNET can be integrated with a genetic algorithm approach.

2. *The units and the updating rule of QNET*

In QNET, every unit has a state that is characterized by a vector rather than by a single activation value. We will denote the state vector of a unit i as $(a_i^1, \dots, a_i^f, \dots, a_i^k)$. The quantity a_i^f is called the amplitude of the i -th unit in the f -th frequency. Every unit i of the network is initialized in such a way that the sum of its squared amplitudes equals one: $(a_i^1)^2 + \dots + (a_i^k)^2 = 1$.

The updating rule of QNET is randomly asynchronous. At every time step t , one unit i is chosen to update its amplitudes. Consider a unit j that sends communication to i . Then, the update procedure contains three steps:

- i. In unit j , a single frequency f is selected. This frequency is selected with a probability equal to $(a_j^f)^2$.
- ii. The information that is sent by j is the quantity a_j^{f-nl} , where nl is a system parameter. This quantity is weighted by the connection w_{ji} from j to i . Unit i adds this quantity to its previous amplitude in the f -th fre-

quency and may add a bias:

$$\Delta a_i^f = w_{ji}(a_j^f - nl) + \mu b_i$$

This operation is repeated for all units j that communicate to i . After the contributions from all units j have been included, a simple non-linear operation is applied in unit i . This operation ensures that amplitudes saturate for large positive values and that they do not become negative:

if $a_i^f > sat$ then $a_i^f = sat$

if $a_i^f < 0$ then $a_i^f = 0$

The parameter sat is a system parameter.

iii. After ii. took place, the amplitudes in unit i are normalized. More specifically, every amplitude is multiplied by a factor $((a_i^1)^2 + \dots + (a_i^f)^2 + \dots + (a_i^k)^2)^{-1/2}$.

The normalization in step (iii) guarantees that the probabilistic rule in (i) remains meaningful. Rules (i) and (ii) can be read as follows: the chance that unit j is observed in frequency f is proportional with the square of the amplitude of j in f . The magnitudes of the changes in amplitudes in i , however, are governed by the non-squared amplitudes in j . This principle has an obvious analogy in quantum theory. For instance, in case of a quantum oscillator, the terminology can be literary maintained: the probability to observe a quantum oscillator in a particular frequency is proportional to the square of the amplitude of this frequency in the decomposition of the state vector. The equations that govern the interactions between oscillators, however, do not contain probabilities but amplitudes.¹ The analogy between QNET and elementary quantum theory is amplified by the fact that, in QNET, a discrete (and finite) number of quantities is associated with every unit. Also in quantum theory, the state of a system is typically composed of a non-continuous, discrete spectre of pure states (on condition that the quantum system is localized in a finite space).

3. Properties of QNET

The properties of QNET can be explained most clearly if we differentiate between three types of initial amplitude patterns. Suppose that we give input-information in the first q frequencies, with $q < k$. This means that

q amplitude patterns are defined for the input units of the network. We notice that the normalization condition requires that input units that have amplitude zero in the first q frequencies receive at least some nonzero amplitudes. For such units, we put the initial amplitude in every frequency larger than q equal to $nl = (1/k-q)^{(1/2)}$.

Suppose that a unit is 'hidden'. In case of QNET, this means that none of the first q amplitude patterns attributes a value to this unit. Then, it is initialized with amplitude $(1/k)^{(1/2)}$ in all frequencies. The system parameter nl that appears in step ii of the updating rule is put equal to this value. Then, nl is a noise parameter: it is a residual amplitude in every frequency of a unit that does not represent information.

Units are allowed to have amplitudes in k frequencies. For some processes, it is useful to confine the communication between the units to k_{eff} frequencies, with $k_{\text{eff}} < k$. This means that, whenever a unit sends communication in a frequency between k_{eff} and k , this information is ignored by the receiving unit.

With these preliminaries, we differentiate between three types of input.

(1) *The input information is presented in a single frequency f only.*

Without loss of generality, we can assume that f is the first frequency. If the initial amplitude pattern in the first frequency is binary, then units that are active in the first amplitude pattern have amplitude zero in the other frequencies. Units that are not active in the first frequency have amplitude $(1/k-1)^{(1/2)}$ in the remaining $k-1$ frequencies.

In the illustrations of the present section, we use a network with 225 units which are organized in a 15×15 matrix. We store 10 patterns in the network. For ease of visualization, the three first patterns are the ones of Figure 1. The next seven patterns are random patterns with a mean activity of $1/2$. The connections in the network are put in accordance with the pseudo-Hebbian Hopfield rule. If we denote the activity of the p -th memorized pattern in unit i V_i^p , this rule prescribes that the connection w_{ij} between unit i and unit j is given by $w_{ij} = e \cdot S_p(2V_i^p - 1)(2V_j^p - 1)$. The value of k is put to 15.

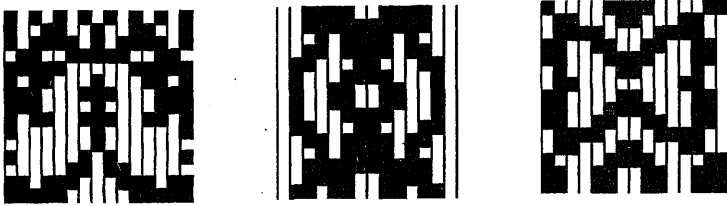


Figure 1. The three first patterns memorized by a network with 225 units.

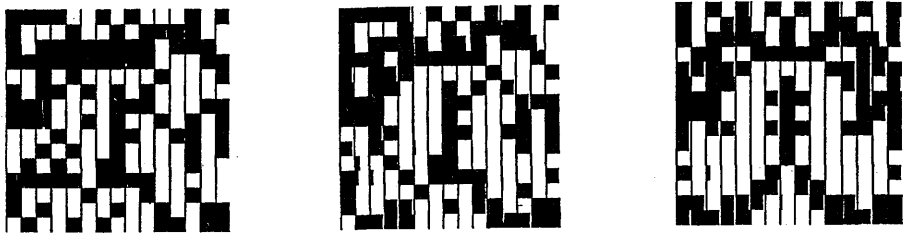


Figure 2. The activation pattern of the schema model after 0, 75 and 450 updates.

When $k_{\text{eff}}=1$, the present type of input pattern reduces QNET to a Hopfield model. Suppose that the first pattern in Figure 1 is distorted 75 successive times. At every distortion, the activation of one randomly chosen unit is replaced by its opposite. If the resulting pattern is presented to QNET, then the first pattern is retrieved after 450 updatings (Figure 2; for ease of visualization, we adapted the random generator in such a way that every unit was selected for an update during a cycle of 225 updatings).

In Figure 3, we show a run of a QNET with $k_{\text{eff}}=2$. As Figure 3 shows, although the information is presented in the first frequency only, a pattern starts to form in the second frequency. After about 450 updates, it becomes clear that this pattern is the mirror image of the pattern that is retrieved in the first frequency.

In order to explain this, we notice that the mirror image of every memorized pattern is also an attractor of the Hopfield model (Amit *et al.*, 1985). During the first updates, the second frequency is occupied by small fluctuations only. If a unit j sends communication to another unit i , and if this communication concerns the second frequency, then the quantity that is communicated is relatively small due to the appearance of nl in step (ii) of the updating procedure. Due to the normalization, fluctuations in the second frequency that would support the first pattern have no chance to persist. Fluctuations that favour the mirror image, on the other hand, are allowed to grow until the mirror-attractor is reached.

QNET with $k_{\text{eff}}=2$ appears to have slightly larger basins of attraction when compared to the Hopfield model. This effect is strengthened significantly when a different type of input presentation is used. Suppose that frequency 1 is initialized like in Figure 3, but that the second frequency receives as input the mirror image of this amplitude pattern (Figure 4). Then, QNET still recovers the memorized pattern after 133 distortions, whereas the Hopfield model recovers the pattern for 96 distortions maximally, compared with 102 for QNET with initial information in a single frequency. The reason for the increase of the basin of attraction is that, from the onset, the second frequency can cooperate with the first one. When the amplitudes in the second frequency approach the mirror image of the first stored pattern, the normalization condition stimulates the amplitudes in the first frequency to converge to the first stored pattern, and vice versa.



Figure 4. Amplitude patterns for other stimulus presentation after 0, 450 and 1800 updtings.

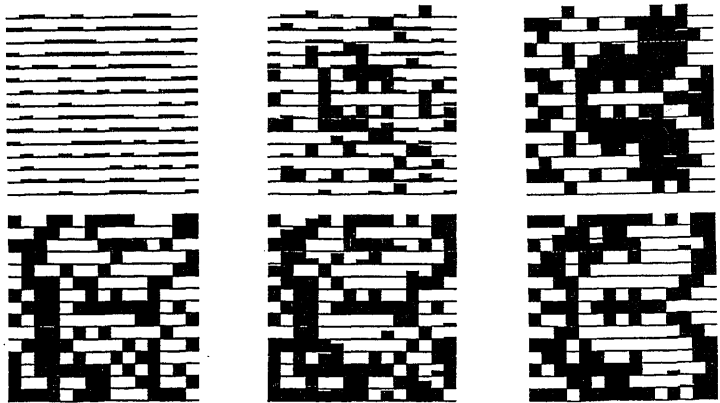


Figure 3. The amplitude patterns of QNET in the first two frequencies after 0, 75 and 450 updtings.

(2) *An identical input pattern is shown in q frequencies, with $q > 1$.* In this situation, the updating process may lead to two types of outcomes. First, a single pattern that is compatible with the input may be found in one or more frequencies. Second, if different patterns are compatible with the input, they will be retrieved in different frequencies. At this instance, we notice that the system parameter *sat* can be used to encourage retrieval of different patterns in different frequencies. Consider a frequency in which no attractor has been found yet, and suppose that a particular unit has a significantly positive amplitude in this frequency. Suppose that the same unit also participates in an attractor that has become realized in another frequency. In the latter frequency, this unit will in general experience stronger support than in the former one. However, if *sat* is not too high, it follows from step (ii) in the updating rule that the unit can remain active in the former frequency too, in spite of the normalization process. Hence, the search process in the former frequency can go on, also when an attractor has already been found in the latter one (in the illustrations that are described in the present paper, the value of *sat* has been put equal to 1).

We start with an illustration in which QNET finds a single solution only for an input of type (2). Consider the first memorized pattern, and suppose that it is distorted fifteen times. We put $q=3$, which means that the pattern is presented in three frequencies. If a unit is active in the distorted version of the first memorized pattern, then it is initialized with an amplitude $(1/3)^{(1/2)}=0.58$ in the first three frequencies. For $k=15$, a unit that is not active in the distorted pattern is initialized with an amplitude $(1/12)^{(1/2)}=0.29$ in the twelve remaining frequencies.

Figure 5 shows the evolution of the amplitudes in the first four frequencies. We observe that, in all three frequencies in which information was given, the memorized pattern is retrieved. Notice that in the fourth frequency, the mirror image of the memorized pattern is retrieved. Due to the normalization condition, this fact enhances the stability of the patterns in the first three frequencies.

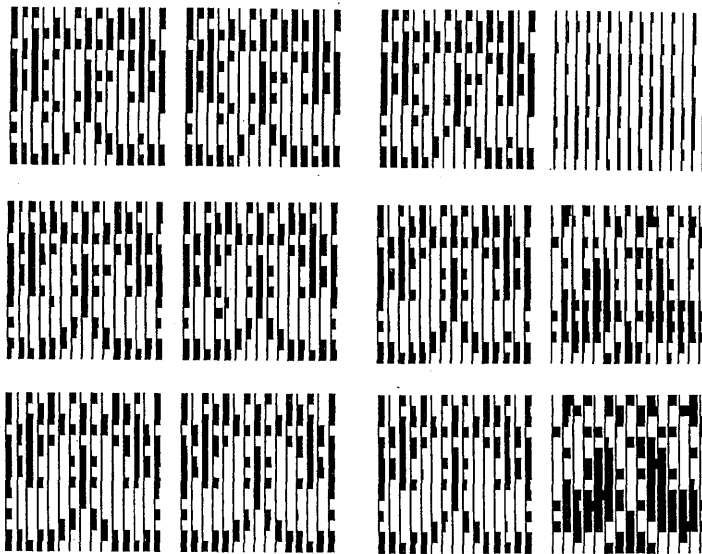


Figure 5. Presentation of a pattern with 15 random distortions in three frequencies. The upper figures show the amplitude patterns after 0 updates in the first, the second, the third and the fourth frequency. The middle figures show the same amplitude patterns after 450 updates; the bottom line shows the amplitudes after 2250 updatings.

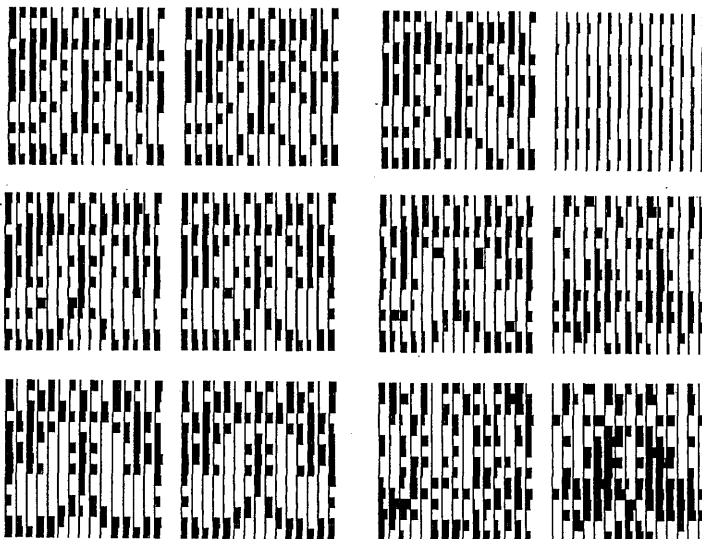


Figure 6. Presentation of a pattern with 45 random distortions in three frequencies. The middle figures show the same amplitude patterns after 450 updates; the bottom line shows the amplitudes after 2250 updatings.

Figure 6 shows what happens for an input pattern that was distorted 45 times. This time, the two first frequencies succeed in retrieving the memorized pattern, but the fluctuations that are inherent in the random updating process draw the pattern in the third frequency away from the first memorized pattern. After 2250 updates, when the pattern has been retrieved in the first two frequencies, the third frequency has not reached stability yet. In this sense, nothing is retrieved in this frequency. When the input pattern is subject to 75 random distortions, the first pattern is retrieved in a single frequency only; the second and the third frequency do not reach stability after 2250 updates.

Next, we consider a more interesting situation. Consider a pattern that overlaps with the two first memorized patterns. Such a pattern can be obtained, for instance, when one takes the units that are active in both patterns. We present the resulting pattern to QNET in the first three frequencies. Again, we put $k=15$, so that a unit that does not receive input information has an initial amplitude $(1/12)^{(1/2)}$ in the remaining twelve frequencies. The network run that is illustrated in Figure 7 was made for $k_{\text{eff}}=6$. Figure 7 shows that QNET retrieves the first memorized pattern in frequencies 1 and 3, whereas the second memorized pattern is retrieved in frequency 2. The amplitudes in the fourth frequency converge to the mirror image of the first pattern. The amplitudes in the fifth frequency do not reach stability after 2250 updates, and the ones of the sixth frequency become extinct.

The capacity to retrieve different solutions in different frequencies differentiates QNET from other models, which often lead to spurious mixtures of different low energy patterns in comparable input situations² (we verified that, if the present initial amplitude patterns are used to define an initial pattern of activation, the Hopfield model converges to a stable spurious state).³

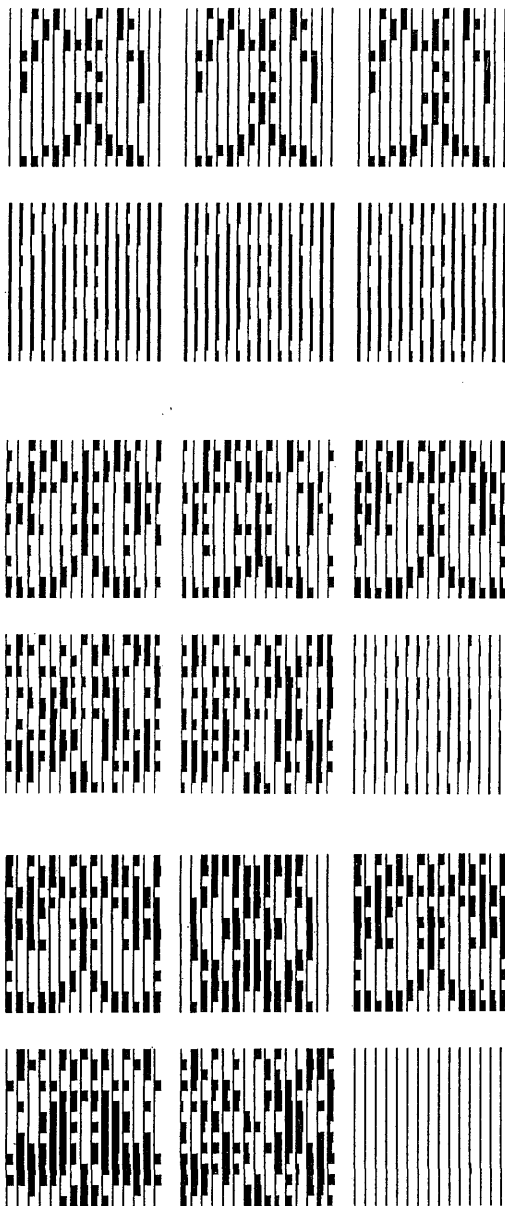


Figure 7. The amplitudes in the first six frequencies are shown after 0, 450 and 2250 updates (in every series of six amplitude patterns, the amplitude patterns 1 to 3 depicted from left to right on the first row; pattern 4 to 6 can be seen on the lower row).

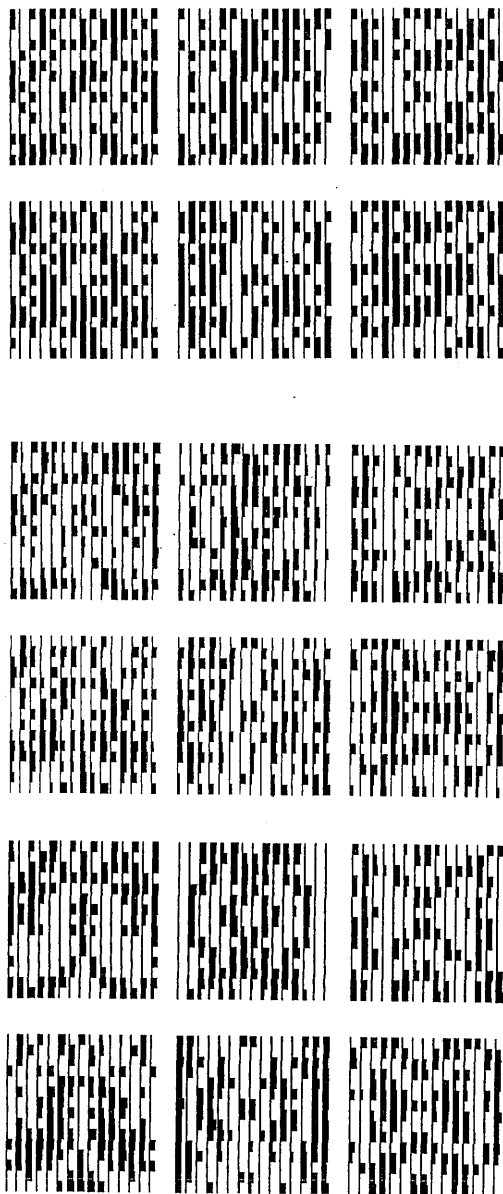


Figure 8. The amplitudes in the first six frequencies are shown after 0, 450 and 2250 updates (in every series of six amplitude patterns, the amplitude patterns 1 to 3 depicted from left to right on the first row; pattern 4 to 6 can be seen on the lower row.

(3) *Different initial amplitude patterns are presented to different frequencies.*

Suppose that the initial amplitude-patterns in different frequencies correspond to noisy versions of different memorized patterns. Then, as a result of the updating operations, these patterns will be retrieved in the respective frequencies. For instance, suppose that the first three memorized patterns of the present example are distorted 75 times each, and that the resulting patterns are presented to the first three frequencies. In order to increase the basin of attraction of the memorized patterns, we present in the fourth, the fifth and the sixth frequency the mirror images of these patterns (Figure 8). For this initialization, every unit has a strictly positive amplitude in exactly three frequencies. Hence, all strictly positive amplitudes are given the value $(1/3)^{(1/2)}$. We put $k_{\text{eff}}=k=6$. Figure 8 shows the amplitude patterns in six frequencies after 0, 450 and 2250 updates. We observe that the first three patterns as well as their mirror images are retrieved.

This property entails that, although QNET is a single distributed network with a single connection matrix, it can act like different schema-models that run in parallel. However, as has been illustrated, processes in different frequencies are not independent. For instance, they encourage each other to converge to a low-energy attractor instead of to a spurious state.

4. *Another illustration of QNET.*

As a second illustration of QNET, we consider a simple set of sentences. We represent sentences in a format that is familiar from particular connectionist approaches to the variable binding problem (Shastri & Ajjanagadde, 1993). In this format, a verb or a relation is represented by a number of units that equals its number of arguments. For instance, 'to give' has three arguments: a subject, a receiver and an object; 'to wear' has two arguments, and so on. Consider the following four sentences that relate to what has happened on Mary's party:

1. John gives a red rose to Mary
2. Jim gives an expensive book to Mary
3. Bill gives delicious champaign to Mary
4. Lucy wears a white dress

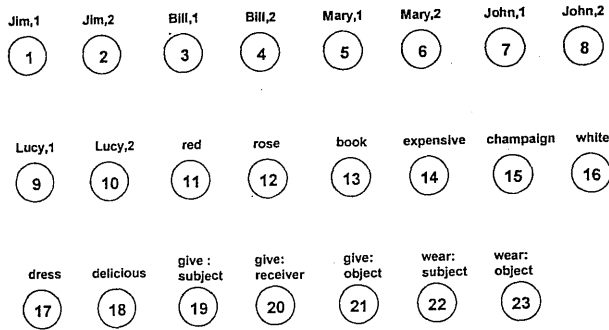


Figure 9. The units that participate in the representation of the example sentences.

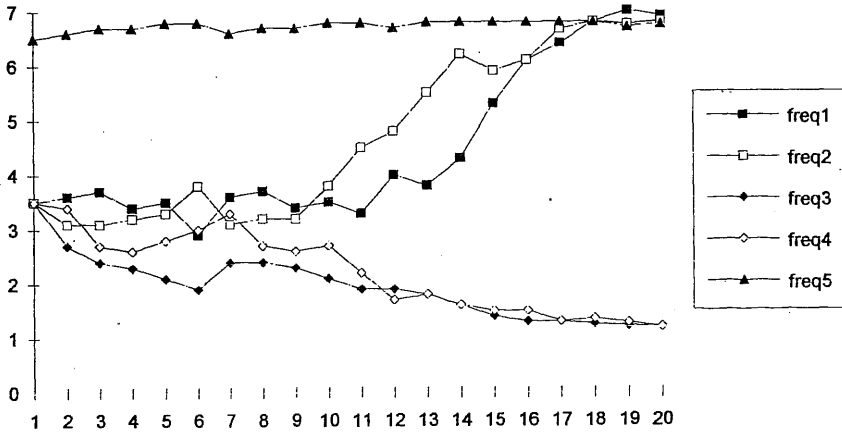


Figure 10. Summed square differences between the values of the fourth sentence and the first five amplitude patterns.

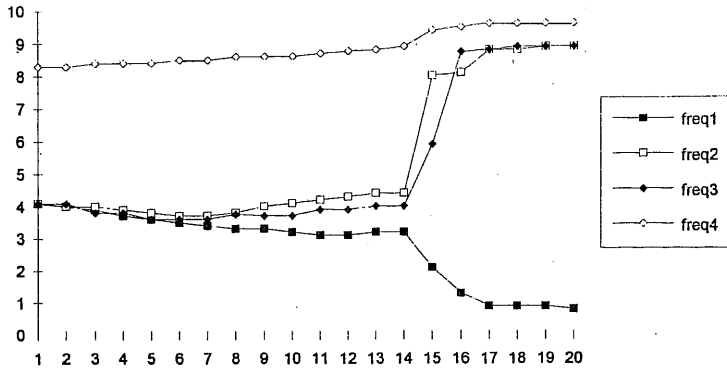


Figure 11a

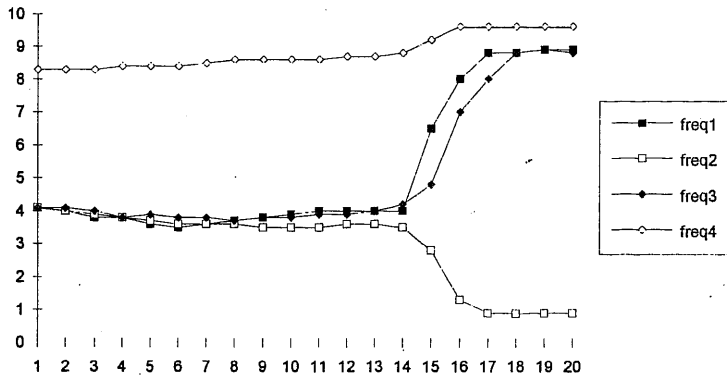


Figure 11b

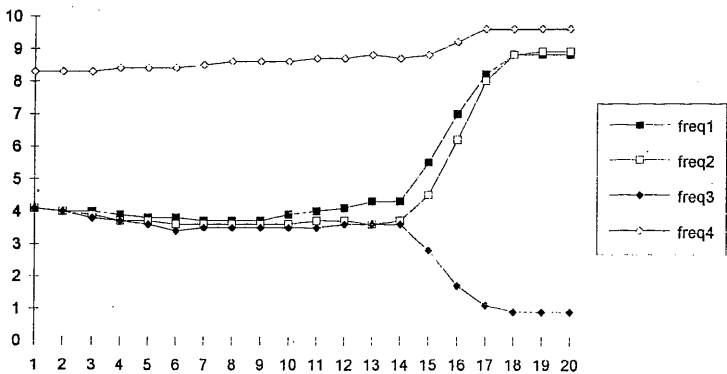


Figure 11c

Figure 11. Summed square differences between the values of the first sentence and the first four amplitude patterns (Figure 11a); the same quantities but for the second and the third sentence are shown in Figure 11b and Figure 11c.

We number units as indicated in Figure 9. Then, the four sentences can be mapped on the following sets of units:

1: 5, 6, 7, 8, 11, 12, 19, 20, 21

2: 3, 4, 5, 6, 13, 14, 19, 20, 21

3: 1, 2, 5, 6, 15, 18, 19, 20, 21

4: 9, 10, 16, 17, 22, 23

We notice that we associated two units with a person. In this way, we obtain representations that are slightly more distributed. As a consequence, we can keep using the pseudo-Hebbian Hopfield rule to store sentences 1-4 in a network with 23 units (with representations that are more local, the number of memorized patterns in proportion to the total number of units becomes too high for the Hopfield rule, and we should use the projection rule instead). Suppose now that we ask the QNET-network what Lucy was wearing. The basic procedure is familiar from the schema-model (Rumelhart, Smolensky et al., 1986): we clamp the units 'Lucy' (units 9 and 10), 'to wear: subject' (unit 22) and 'to wear: object' (unit 23), and we wait and see if the network turns on any other unit. In the run that is illustrated, we clamped units 9, 10, 22 and 23 with amplitude $(1/4)^{(1/2)}$ in the first four frequencies. Other units were initialized with amplitude zero in these frequencies, and amplitudes $(1/11)^{(1/2)}$ in the remaining eleven frequencies (we put $k=15$ and $k_{\text{eff}}=5$).

In every unit, we calculated the squared differences between the amplitudes in the five first frequencies and the value of the four sentences. These quantities were summed over all units. The resulting five quantities for sentence 4 are plotted in Figure 10. Every step on the horizontal axis of figure 10 corresponds with 23 successive random updatings. As can be seen in Figure 10, sentence 4 becomes realized in both the third and the fourth frequency.³

Next, suppose that QNET is asked if Mary was given something by someone. Then, the units 'Mary' (units 5 and 6), 'give: subject' (unit 19), 'give:receiver' (unit 20) and 'give:object' (unit 21) are clamped. For instance, we can clamp these units in the first three frequencies with amplitudes $(1/3)^{(1/2)}$, with $k_{\text{eff}}=5$. As expected, QNET retrieves the three first sentences. Figure 11a shows the summed square differences between the four first amplitude patterns and the first sentence; Figure 11b and 11c show the same quantities for the second and the third sentences. In the run that is illustrated in Figure 11, sentence 1 is retrieved in frequency 1 (Figure 11a), sentence 2 in frequency 2 (Figure 11b), and sentence

3 in the third frequency (Figure 11c).⁴

A QNET model can store sentences and retrieve in parallel different distributed sentence representations in different frequencies. Also the way in which variables are bound within a sentence can be stored if QNET is enriched with higher order connections. I refer to Van Looke (1996) for a brief discussion of this matter.

5. *QNET and genetic algorithms*

We have seen a number of interesting properties of QNET that are due to the presence of amplitude patterns in different frequencies, and due to the updating rule. Another advantage of this model is that it suggests a link between neural networks and genetic algorithms. Genetic algorithms have been proposed as a method to train backpropagation networks (19xx), and genetic considerations have been integrated in more general neural frameworks (Edelman, 1987), but thus far, this did not result in qualitatively new capacities for neural nets. We line out how an integration of QNET with genetic algorithms can be more fruitful.

Let us briefly recapitulate how a genetic algorithm solves a problem concerning a particular system (Holland, 1975). As a first step, the variables associated with the system are collected in a string of binary or non-binary numbers. Then, such a string is mapped on a real number by a suitably chosen fitness function. The problem is solved when a string is found that maximizes the fitness function. To this end, the genetic procedure of variation and selection is applied.

The algorithm usually starts with a set of randomly generated strings. According to the standard procedure, this set is replaced by a 'next generation' of strings in accordance with two processes. First, a new, preliminary set of strings is formed by letting every string reproduce itself with a probability that is proportional to its fitness value. Next, strings of this set are combined with other strings of the set. This combination may involve a 'crossover', in which both strings exchange string-parts. Subsequently, a random clip with low probability may follow. The resulting set of strings is replaced again by a next generation by the same procedure. This operation is repeated until a population is obtained with at least one string that solves the problem. If the initial set of random strings is sufficiently large, then many optimization problems

can be solved when the number of generations that is considered is reasonably large.

Neural nets of the Hopfield-type can also be used for constraint-satisfaction and optimization problems, but they are subject to a restriction when compared with genetic algorithms. In order for a constraint-satisfaction problem to be solvable by a Hopfield net, its fitness (or 'energy'- or 'harmony'-) function must be expressible as a bilinear form. Genetic algorithms are not subject to this limitation. We give a practical example of a problem where this limitation is critical.

Consider the problem of finding a fractal code for a picture of a tree. Suppose that we try to approximate the tree by a single fractal (i.e. the image is not segmented). A fractal can be characterized by a set of contractive transformations. Every transformation has six parameters. Thus, for instance, a fractal composed by 9 contractive transformations is characterized by 54 continuous parameters. If every continuous parameter is mapped on a discrete set of 16 values, then four bits represent one parameter, and the fractal can be characterized by a binary string of 216 values. In addition, every transformation may be provided with a colour. If, for instance, every colour that is considered is encoded by 6 bits, then the total length of the string becomes 270 bits.

A genetic algorithm that aims at finding the fractal code of the picture then starts with a set of binary random strings of length 270. For every string, the fitness is calculated by generating the corresponding fractal image, and by comparing this image with the image that is given. The more close both images are, the higher the fitness of the string. If the initial number of strings is large, it may be expected that an approximation will be found after a sufficiently high number of generations. In case of this example, calculating the fitness function is a fairly complex (and time consuming) matter. There is no way to express this fitness function in a bilinear form that would be suited for implementation on a Hopfield-style network (I refer to Van Loocke, 1997 for a concrete elaboration of this example).

As far as the form of a fitness function is concerned, a genetic algorithm is a more general method than is a neural network. Conversely, a neural network is good at a number of tasks for which a genetic algorithm is not or not equally well suited. A neural network can memorize a massive number of patterns that it has learned in accordance with a learning algorithm. To some extent, one may include information about

previous performances in a genetic algorithm also: along with random patterns, the initial population of the algorithm may contain patterns that appeared useful in related tasks. However, this memory property is much more primitive, and much more subject to trial and error-procedures than is the systematic neural net approach. Therefore, one may wonder if it is possible to combine the memory capacity of neural networks with the more general optimization capacity of genetic algorithms.

The QNET model suggests such a combination. Let us turn back to the example that has been referred to a few lines higher. Suppose that strings that were able to code images of trees are memorized in a Hopfield network in accordance with, for instance, a Boltzmann learning procedure or in accordance with one of its faster variants. Suppose that a new image is presented for which a fractal code must be searched. To start with, a number of random strings are generated and a genetic algorithm is allowed to produce p generations, with p a number that is relatively low. Subsequently, the strings that resulted after p generations are mapped on the different frequencies of a QNET-model. This model is allowed to update its amplitudes during a number of q cycles, with q again a relatively low number. During this part of the process, the memories of the net attract the amplitude patterns. Then, the strings that have appeared in the respective frequencies are read, and they are used again in a genetic procedure that generates p generations. The resulting strings are mapped on the frequencies of the QNET-model, and so on. In case a code has been found, it can be added to the set of training stimuli that determines the connections of the network.

6. Discussion

The binding problem of connectionism can be solved if a network includes units that support different amplitudes or 'markers' instead of a single activation value. Shastri and Ajjanagadde (1993) suggest that, from a physiological point of view, such markers correspond to temporal variables. This is consonant with the fact that the importance of temporal variables has been suggested regularly in recent physiological literature. This holds, for instance, for colour vision (Mc Clurkin *et al.*, van Esch *et al.* 1980), texture (Richmond *et al.*, 1989) audition (Ghitza, 1992), pain (Emmers, 1981) taste (Di Lorenzo & Hecht, 1993) and so on (an

overview can be found in Cariani, 1995). Also the possibility that temporal variables help solving the binding problem has been put forward at different instances in this literature (Eckhorn *et al.*, 1990; Engel *et al.*, 1991). If it is assumed that temporal variables realize bindings, then different possibilities remain. For instance, the bindings may be expressed by frequencies or by phases. In the latter case, units are bound if they fire in synchrony (this option is defended in Shastri and Ajjanagadde, 1993).

Relative to these discussions, QNET is a model on an abstract level. The terminology 'frequencies' in the context of the QNET-units has been chosen because of the fact that QNET has been inspired on particular elementary quantum systems. The present study of QNET does not predict a particular neuro-physiological realization of these 'frequencies'. Rather, we studied the computational properties of QNET on a level that allows for different physical realizations. It must also be clear that QNET is not a quantum computer in the sense of Deutsch (1985) or Lockwood (1989), nor does it predict quantum interactions in the brain of the type that Penrose (1989, 1994) has in mind. QNET is a classical model that is inspired on some ideas that also occur in elementary quantum theory, but its simulation or its physical realization does not depend, for instance, on the use of SQUIDS. A quantum computer would operate in many parallel worlds. QNET operates in many parallel frequencies, but all of them belong to the same world.

Section 5 of this paper suggests a way to reconcile QNET with genetic algorithms. Genetic methods in the cognitive domain have not only gained attention within a context of artificial intelligence applications, but also in a context of more theoretical developments. In special, different authors proposed to describe the dynamics of concepts in terms of evolving memes. These descriptions usually do not assert that concepts must evolve according to present-day artificial genetic algorithms. However, explanations often include suggestions in this direction, and discuss, for instance, properties of fitness landscapes for memes. In view of the merits that connectionism has acquired during the past ten years, it is relevant to notice that genetic approaches are not in contradiction with connectionist models, but that they can give a wider scope to connectionist models.

NOTES

1. Slightly more technically, the state of a quantum system is a vector of a Hilbert space. A state that corresponds to an unequivocal frequency is called a pure state. A state of a quantum oscillator can always be written as a weighted combination of pure states. The coefficients in this combination are amplitudes, and the squares of these coefficients give the observation probabilities. The state of a quantum system obeys the Schrodinger equation. This equation describes the evolution of a system and the interactions between different systems. The equation contains the state-vector, but not the observation probabilities. In an operation that frequently occurs in quantum calculations, the state vector is replaced by its decomposition in pure states. Then, the Schrodinger equation results in an equation for amplitudes.
2. To explain this QNET-property, suppose that, as a consequence of a suitably chosen input pattern, the same memorized patterns are partially active in more than one frequency. For instance, suppose that there is a set of units which participate in memory patterns 1 and/or 2, and suppose that these units are active in frequency 1 as well as in frequency 2. Now consider a unit that participates in the representation of memory pattern 1 but not in the representation of memory pattern 2. Suppose that, in this unit, one of both frequencies has a slightly higher amplitude than the other frequency. For instance, suppose that the amplitude in the first frequency in this unit is slightly higher. Then, due to the squares that appear in step (i) of the updating rule, in its communication with other units, this unit supports the first memory pattern significantly stronger in the first frequency. For the same reason, it communicates significantly less information that concerns its amplitude in the second frequency. Hence, it will not help to draw the first memory into the second frequency, nor will it inhibit the realization of the second memory pattern in the latter frequency. Such processes may be at work in several units, and some of them may counteract each other. Then, the network enters an unstable phase that stimulates the occurrence of more fluctuations, until a fluctuation occurs that favours one of both memory patterns to a sufficient extent in one of both frequencies. This effect, that has been observed in many simulations of QNET, would be impossible if step (i) of the updating rule contained a linear principle instead of a square one.
3. The pattern of activation for which this assertion was verified takes value one when the amplitudes of the first three frequencies is $(1/3)^{(1/2)}$, and it takes value zero when the amplitudes in the first frequencies are zero.

4. We notice that the quantities that are plotted and that concern the first four frequencies cannot decay below 1.17 if sentence 4 is retrieved in two frequencies. Every clamped unit gives a contribution $(1-(1/4)^{(1/2)})^2 = 0.25$ to these quantities. Once the sentence has been retrieved, a non-clamped unit of sentence 4 gives a contribution $(1-(1/2)^{(1/2)})^2 = 0.086$, so that the minimal value for the distance measure in every frequency is $4 \times 0.25 + 2 \times 0.086 = 1.17$.
5. This time, the distance measure has a lowest value of $5 \cdot (1-(1/3)^{(1/2)})^2 = 0.893$.

REFERENCES

- Amit D., Gutfreund H., Sompolinsky H. (1985), Spin-glass models of neural networks, *Physical Review A* **32**, pp. 1007-1018
- Cariani P. (1995), As if time really mattered. Temporal strategies for neural coding of sensory information, *CCAI: Journal for the Integrated Study of Artificial Intelligence, Cognitive Science and Applied Epistemology* **12**, pp. 157-219
- Deutsch, D. (1985), Quantum theory, the Church-Turing principle and the universal quantum computer, *Proceedings of the Royal Society A* **400**, pp. 97-117
- Di Lorenzo P., Hecht G. (1993), Perceptual consequences of electrical stimulation in the gustatory system, *Behavioral Neuroscience* **107**, pp. 130-138
- Eckborn R., Reitboeck H., Andt M., Dicke P. (1990), Feature linking via synchronization among distributed assemblies: simulations of results from cat visual cortex, *Neural Computation* **2**, pp. 293-307
- Edelman G. (1987), *Neural Darwinism*, New York: Basic Books
- Emmers R. (1981), *Pain, a spike-interval coded message in the brain*, New York: Raven Press
- Engel A., Koenig P., Kreiter A. Gray C., Singer W. (1991), Temporal coding by coherent oscillations: a potential solution to the binding problem, in Schuster H., Singer W. (eds.), *Nonlinear dynamics and neural networks*: Weinheim
- Fahlman S. (1979), *NETL: A system for representing real-world knowledge*: MIT Press
- Ghitza O. (1992), Auditory nerve representation as a basis for speech processing, in Evans E., Wilson J. (eds.), *Psychophysics and physiology of hearing*, London: Academic Press
- Hopfield J. (1982), Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy*

- of Sciences: USA* **79**, pp. 2554-2558
- Kanter I., Sompolinsky H. (1987), Associative recall of memory without errors, *Physical Review A* **33**, pp. 380-392
- Lockwood M. (1989), *Mind, brain and the quantum*, Oxford: Blackwell
- Mc Clurkin J., Zarbock J., Optican L. (1993), Neurons in primate visual cortex multiplex information about red/green, blue/yellow, and black/white opponencies using temporal codes, *Neuroscience Abstracts* **19**, 1576
- Penrose O. (1989), *The emperor's new mind*, Oxford: Oxford University Press
- Penrose O. (1994), *Shadows of the mind*, Oxford: Oxford University Press
- Personnaz L., Guyon I., Dreyfus G. (1986), Collective computational properties of neural networks: new learning mechanisms, *Physical Review A* **34**, pp. 4217-4228
- Richmond B., Optican L., Gawne T. (1989), Neurons use multiple messages encoded in temporally modulated spike trains to represent pictures, in Kulikowski J., Dickenson C. (eds.), *Seeing contour and colour* pp 705-713, New York: Pergamon Press
- Rumelhart D., Smolensky P., Mc Clelland J., Hinton G. (1986), Schemata and sequential thought processes in PDP models, in: Mc Clelland, J., Rumelhart, D. (eds.), *Parallel distributed processing: explorations in the microstructure of cognition*, vol 2: MIT Press
- Shastri L., Ajjanagadde V. (1993), From simple associations to systematic reasoning: a connectionist representation of rules, variables and dynamic bindings using temporal synchrony, *Behavioral and Brain Sciences* **16**, pp. 417-494
- Van Loocke Ph. (1991), Study of a neural network with a meta-layer, *Connection Science* **4**, pp. 367-379
- Van Loocke Ph. (1994a), *The dynamics of concepts: a connectionist model*, Heidelberg/New York/Berlin: Springer Verlag
- Van Loocke Ph. (1994b), *Connectionisme en natuurfilosofie. Een onderzoek van enkele concrete mogelijkheden tot interactie*, Ghent: Communication and Cognition (higher aggregation dissertation; published in Dutch)
- Van Loocke Ph. (1994c), QNET: A new connectionist architecture and its relevance for variable binding and constraint satisfaction problems, *Cybernetica* **38**, pp. 85-106
- Van Loocke Ph. (1996), Constraint satisfaction in different frequencies of a single network and the binding problem, paper submitted to the NIPS-1996 conference
- Van Esch A., Van 't Veld J., Koenderinck J. (1988), A temporal opponent

red-green mechanism, *Biological Cybernetics* 58, pp. 329-355