arXiv:cond-mat/0102490 27 Feb 2001

Training a perceptron in a discrete weight space

Michal Rosen–Zvi and Ido Kanter
Minerva Center and the Department of Physics, Bar–Ilan University
Ramat–Gan, 52900, Israel

On-line and batch learning of a perceptron in a discrete weight space, where each weight can take $2L+1$ different values, are examined analytically and numerically. The learning algorithm is based on the training of the continuous perceptron and prediction following the clipped weights. The learning is described by a new set of order parameters, composed of the overlaps between the teacher and the continuous/clipped students. Different scenarios are examined among them on-line learning with discrete/continuous transfer functions and off-line Hebb learning. The generalization error of the clipped weights decays asymptotically as $exp(-K\alpha^2)/exp(-e^{|\lambda|\alpha})$ in the case of on-line learning with binary/continuous activation functions, respectively, where α is the number of examples divided by N, the size of the input vector and K is a positive constant that decays linearly with $1/L$. For finite N and L, a perfect agreement between the discrete student and the teacher is obtained for $\alpha \propto \sqrt{L\ln(NL)}$. A crossover to the generalization error $\propto 1/\alpha$, characterized continuous weights with binary output, is obtained for synaptic depth $L > O(\sqrt{N})$.

I. INTRODUCTION

Neural networks and the perceptron as the simplest prototype have become most popular as a tool for understanding human learning and as a basis for many various applications [1,2]. We are interested in the perceptron learning ability as an archetype for machines that are able to learn. Most of the perceptrons that have been studied until now are under two totally different constraints, two extremes. Either the teacher weight vector is restricted to a binary space, (the Ising teacher), or it is continuous, confined to a hypersphere. Only a few aspects of the learning ability of weights which are confined to have a finite number of values have been studied, although the realistic case on digital computers where numbers have a finite depth representation is described by this model. Furthermore, the applicability of neural networks to biology and to the construction of real devices requires the understanding of the interplay between the weights depth and the network ability of learning. Those systems are the intermediate case, in which the weights are confined to finite space, $(2L+1)^N$, when L is an integer and N stands for the input size. [3–5].

The generalization ability of such networks, in which the synapse has a finite depth has been analyzed by using replica calculations and has been found to have interesting nontrivial behavior of phase transition. The learning procedure composed of two phases; one in which the learning ability is very limited, the generalization error is finite, another phase is when the generalization error is exactly zero, perfect learning is gained and it happens in a finite α, where α is the number of patterns divided by the size of the input N, [5]. Nevertheless, replica calculations do not involve practical algorithms that one may use in order to obtain that learning behavior. In the Ising case, for instance, although a phase transition was predicted, no practical algorithm reproduces this discontinuous behavior [6,7].

In contrast to the batch learning, when all the examples are used together to achieve perfect learning, on-line learning is a procedure in which an update rule is used and learning in each step utilizes only the last of a sequence of examples. Such an algorithm drastically reduces the computational effort compared with batch learning and no explicit storage of a training set is required [8]. It was shown that there is no updating rule that uses only the discrete vector for updating and results in perfect learning [9].

In this paper we address the issue of practical learning from a finite depth teacher. The method we introduce is based on the clipping of a continuous perceptron. Having an artificial continuous weight vector enables smooth learning; clipping it results in a discrete student $\vec{W}^S$, whose components are close to those of the teacher. This method has been used successfully in the Ising perceptron [6,7,10,11]. The questions that arise from the procedure above are; whether learning is possible at all and if it is possible, does it give better results then the learning in a continuous space. It seems very natural that if the weights" depth is very large, i.e. there are many possible values to each weight, the learning behavior of the discrete weights will be exactly the same as those of a continuous weight. However, in the following we examine if and what are the scaling relations between both properties, L and N.

Our main results are: (a) Learning in the case of finite depth is possible by using a continuous precursor. This result was confirmed both analytically and numerically. (b) In the on-line learning scenario: Having a binary output results in a fast decay of the generalization error and at the large α regime it decays super-exponentially with α, $\epsilon_g \propto \exp(-K\alpha^2)$ where K is some constant. Having a continuous output results in a much fast decay of the generalization error,$\exp(-K_1 \exp(K_2\alpha))$, where K_i

are positive constants. (c) In batch Hebbian learning, having a binary activation function, the generalization error falls exponentially with α. (d) Perfect learning is obtained when N is very large but finite, unlike the continuous perceptrons performance. Quantitatively, for a given N and L the perfect learning is achieved for $\alpha_f \propto O(\sqrt{L \ln(LN)})$. (e) A crossover to the behavior of the generalization error in the presence of continuous weights occurs for $L > o(\sqrt{N})$.

The paper is organized as follows: In section II the architectures and the dynamical rules are defined as well as the continuous and discrete students. In section III the order parameters are defined and the relations between the overlaps of the continuous teacher with the discrete/continuous students are derived analytically. In section IV, the dynamical evolution of the order parameters in the case of binary output is derived analytically and confirmed by simulations. Both, on-line scenario and Hebbian learning are examined. In section V the case of large synaptic depth and the crossover to the continuous weights is studied. In section VI, the perfect learning in finite N systems is examined both analytically and numerically. Section VII is devoted to analyze results in the case of continuous output. Finally, in section VIII results are concluded and open questions are addressed.

II. THE MODEL

A. The Architecture

We investigate a teacher-student scenario where both nets are single-layer feed-forward. The examples are generated by the so-called teacher, which is known to be restricted to a well-defined discrete set of values. We define a synaptic depth L and a set of digital values to be as follows [3,5],

$$W_i^T = \pm\frac{1}{L}, \pm\frac{2}{L} ... \pm 1. \tag{1}$$

In case that the zero value is part of the game, the possible values of the weights are

$$W_i^T = 0, \pm\frac{1}{L}, \pm\frac{2}{L} ... \pm 1. \tag{2}$$

For the sake of simplicity, we present results in this paper only for the including zero case (Eq. 2). It is easy to generalize our results for the other case, (Eq. 1).

The input patterns $\vec{\xi}^\mu$ are chosen at random and independent of each other. In the following they are drawn from a Gaussian distribution with zero mean and unit variance. The size of the teacher, the student and the input is N. For any input $\vec{\xi}$ the so-called teacher generates an output, S, according to some rule

$$S = F(\frac{\vec{W}^T \cdot \vec{\xi}}{\sqrt{N}}). \tag{3}$$

In the following we will discuss both binary and continuous rules. The student has in mind the rule F and the discrete set of values that the teacher is confined to. In addition, in an on-line learning scenario, the student is given in each time step, μ, the input $\vec{\xi}^\mu$ and the teacher's output S^μ, whereas in batch learning the set $(\vec{\xi}^\mu, S^\mu)$ $\mu = 1 ... \alpha N$ is given altogether.

B. Dynamics of the Weights

A continuous precursor for the student, $\vec{J}$ is needed for learning from a discrete teacher. The learning procedure, having a continuous student, is well known. In an on-line scenario at each step the continuous student updates its weight vector according to some learning algorithm (f). The generic form of the learning algorithm is

$$\vec{J}^{\mu+1} = \vec{J}^\mu + \frac{\eta}{\sqrt{N}} f(S^\mu, x_J^\mu) \vec{\xi}^\mu S^\mu, \tag{4}$$

where η is the learning rate and x_J is the student's local field, $x_J \equiv \frac{1}{\sqrt{N}} \vec{J} \cdot \vec{\xi}$. Such a learning algorithm means that at each learning step μ, the current weight vector $\vec{J}^\mu$ is updated according to the new example, $\vec{\xi}^\mu$ and each example is presented only once.

In an off-line scenario, there is a set of examples $\vec{\xi}^\mu$ $\mu = 1 ... \alpha N$ and they are used altogether to gain perfect learning. There are methods in which the off-line leaning is made according to a rule that defines an additive quantity of all the examples. The Hebb learning is an archetype of those methods,

$$\vec{J}^{Hebb} = \sum_{\mu=1}^{\alpha N} \vec{\xi}^\mu S^\mu. \tag{5}$$

Such procedures were shown to end up in perfect learning [14,15]. Since having a discrete teacher is merely a special case, not using the knowledge that the teacher is confined to a discrete set of values gives the well-known results; an exponential decay in the case of continuous rule (on-line learning [12,13]) and a power law decay in the case of binary rule (on-line and off-line learning [14–18]).

The way to gain from the knowledge of the discrete nature of the weights is in the center of our work, and it is based on having in addition a discrete student $\vec{W}^S$ derived from the continuous one using the following clipping procedure. A continuous weight is clipped to the nearest discrete value, among the $2L + 1$ possibilities. Such a clipping procedure is the optimal one with the lack of any prior knowledge about the weights accept that each value appears with the same probability. We define limit values, λ_l, which are arranged in increasing order. The limit values divide the continuous region of the precursor weight vector components to $2L+1$ intervals, according to the number of the available values as in Eq. 2. The clipping process is such that J_i is mapped onto $\frac{l}{L}$ for $J_i \in (\lambda_l,\ \lambda_l+1)$. The set of limits includes $\{\lambda_{-l}, \lambda_{-l-1}, ...\lambda_{-1}, \lambda_0, \lambda_1...\lambda_{l+1}\}$. It is given by the following mathematical rule:

$$W_i^S = \sum_{l=-L}^{L} \frac{l}{L}[\theta(\lambda_{l+1} - J_i) - \theta(\lambda_l - J_i)] \tag{6}$$

where θ is the Heavyside function.

Since the value of those limits, λ_l, is somewhat unclear, we would like to exemplify it with some specific cases. In the case of $L=1$,Eq. 1, for instance, due to symmetry it is obvious that the limit between -1 and 1 should be 0. Hence, one introduces the following limits, $\lambda_{-1} = -\infty$, $\lambda_0 = 0$, $\lambda_1 = \infty$. Evaluating the mapping equation results in the well known clipping rule, $W_i^S = sign(J_i)$, [6,10]. Finding the appropriate value for all other cases but the Ising perceptron becomes more complicated, the continuous space is no longer divided into two clear regions and hence one has to consider carefully the value of the limits.

In this paper we chose to nail down the general results by focusing in the including zero case, $L=1$, i.e., $W_i = 0, \pm 1$. This case is known as the diluted Ising case and some other aspects of it have been studied in references [19–21]. It contains the simplicity of the Ising case on the one hand and introduces a more generality concerning digital values on the other hand. In this case, there is only one unknown parameter, λ_1, since $\lambda_2 = -\lambda_{-1} = \infty$, and $\lambda_0 = -\lambda_1$.

While choosing the value of the limits, (in the last case that means only choosing the value of λ_1) one should take into consideration the a priori knowledge about the weights of teacher. It is clear that the limits should scale with the student norm, since the exact set of values that the continuous student end up with is irrelevant. The mapping rule ensures that the digital student ends up with the same values as those of the teacher. This will be shown only after analyzing the new order parameters and their dependent on the former one, as is presented in the next chapter.

III. THE ORDER PARAMETERS

Evaluating the agreement between teacher and student is made by calculating either the generalization error or the order parameters. The generalization error, ϵ_g, is calculated by taking the average of the student/teacher disagreement over the distribution of input vectors. The generalization error is given, in principle, by the overlaps between the vectors, (the so-called order parameters). However, in order to get into details one has first to define the rule, (F in Eq. 3). This will be done in the next sections. In the following we concentrate in introducing the complete set of order parameters and their inter-relations.

In our case there are three vectors and hence two interdependent sets of order parameters; one set concerns the continuous overlaps,

$$\begin{aligned} R_J &\equiv \frac{1}{N}\vec{J}\cdot\vec{W}^T, \\ Q_J &\equiv \frac{1}{N}\vec{J}\cdot\vec{J}, \end{aligned} \tag{7}$$

and one set concerns the digital vector's overlaps,

$$\begin{aligned} R_W &\equiv \frac{1}{N}\vec{W}^S\cdot\vec{W}^T, \\ Q_W &\equiv \frac{1}{N}\vec{W}^S\cdot\vec{W}^S. \end{aligned} \tag{8}$$

Note that the dynamical evolution of the continuous set of order parameters, Eq. 7, is independent of the clipped order parameters, since the *training* is done only following the continuous weights. In contrary to the training process the *prediction* and the generalization error is made following the clipped student. Hence, finding the quantitative interplay between the continuous set of order parameters, Eq. 7, and the discrete set of order parameters, Eq. 8, is the cornerstone for the analytical description of the generalization ability of the student.

In this section we examine the relation between the clipped set and the continuous one. The development of R_J, Q_J is not influenced by the clipping method. Hence, finding out the above relation enables finding the development of the clipped order parameters and results in a description that gives the whole picture of the learning process.

The teacher's norm is determined according to the a-priori probabilities for each discrete value. Having equal probability and taking the thermodynamic limit results in the norm,

$$T \equiv \frac{1}{N}\vec{W}^T\cdot\vec{W}^T = \frac{1}{L^2 n_L}\sum_{l=1}^{L} l^2 = \frac{1}{3} + \frac{1}{3L}, \tag{9}$$

where n_L defined to be the number of optional values, $n_L = 2L+1$. The order parameters in the clipped machines, R_W and Q_W, as a function of those of the continuous machine, R_J and Q_J, are evaluated as follow:

$$\begin{aligned} R_W &=< \frac{1}{N}\sum W_i^T \frac{l}{L}[\theta(\lambda_{l+1} - J_i) - \theta(\lambda_l - J_i)] > \\ Q_W &=< \frac{1}{N}\sum \frac{l^2}{L^2}[\theta(\lambda_{l+1} - J_i) - \theta(\lambda_l - J_i)]^2 > \end{aligned} \tag{10}$$

where $< A >$ is an average over the known constraints and the known overlaps,

$$< A >\equiv \frac{Tr_{W^T}\int dJ_i \delta(J_i^2-NQ_J)\delta(J_iW_i^T-NR_J)A}{Tr_{W^T}\int dJ_i \delta(J_i^2-NQ_J)\delta(J_iW_i^T-NR_J)}. \tag{11}$$

The validity of this average is based on the assumption that all vectors $\vec{J}$ which are consistent with the constraints are taken with equal probability. This assumption is violated in case that the updating of the continuous vector itself is made according to the clipped one, see [6,11].

The results are:

$$\begin{aligned} R_W &= \frac{1}{2L^2n_L}\sum ll'[erf(\Phi_{l+1,l'})-erf(\Phi_{l,l'})], \\ Q_W &= \frac{1}{2L^2n_L}\sum l^2[erf(\Phi_{l+1,l'})-erf(\Phi_{l,l'})], \end{aligned} \tag{12}$$

where the summation is over all the possible values, starting from $l, l' = -L, \ -L+1, ..., L$ and we defined

$$\Phi_{l,l'} \equiv \frac{\frac{\lambda_l}{\sqrt{Q_J}} - \frac{\rho_J}{\sqrt{T}}\frac{l'}{L}}{\sqrt{2(1-\rho_J^2)}}), \tag{13}$$

where $\rho_J \equiv \frac{R_J}{\sqrt{T}\sqrt{Q_J}}$, $\rho_W \equiv \frac{R_W}{\sqrt{T}\sqrt{Q_W}}$ are the geometrical order parameters.

In the limit $L \to \infty$ the summation in Eq. 12 can be replaced by an integral. Calculating the integrals in this limit results in the obvious identities, $R_W = R_J$ and $Q_W = Q_J$. Note that taking integrals instead of summation imposes an inequality. The difference $\Phi_{l,l'} - \Phi_{l+1,l'}$ tends to zero as long as $L >> 1/\sqrt{1-\rho_J^2}$, (see Eq. 13). Hence, in the event that L is very large, learning with the continuous student or learning with the clipped version performs the same result as long as ρ_J is smaller then $2/L$. This limit is discussed in section VI.

We exemplify the general results in the case of the diluted Ising perceptron. In that case we used the following limits,

$$\begin{aligned} \lambda_2 = -\lambda_{-1} = \infty \\ \lambda_1 = -\lambda_0 \end{aligned} \tag{14}$$

and the teacher's norm is $T = 2/3$. The mapping above gives

$$\begin{aligned} R_W &= \frac{1}{3}[erf(A_+)+erf(A_-)] \\ Q_W &= 1-\frac{1}{3}erf(A_0)+\frac{1}{3}erf(A_-)-\frac{1}{3}erf(A_+) \end{aligned} \tag{15}$$

were $A_\pm = \frac{\rho_J/\sqrt{T}\pm\lambda_1/\sqrt{Q_J}}{\sqrt{2(1-\rho_J^2)}}$ and $A_0 = \frac{\lambda_1}{\sqrt{2Q_J(1-\rho_J^2)}}$.

From Eq. 15 one can verify that at the limit $\alpha \to \infty$ when the continuous order parameters achieve a perfect learning, $\rho_J \to 1$, the discrete order parameters achieve perfect learning as well, $R_W \to 2/3$, $Q_W \to 2/3$ and $\rho_W \to 1$ given that the positive quantity, λ_1, is smaller than $\lambda_1 < \sqrt{Q_J/T}$.

In general, in order that the digital student will gain perfect learning it is necessary that the relation $\sqrt{Q_J/T}(l-1) < \lambda_l < \sqrt{Q_J/T}l$ holds for any positive l. Note that the interpretation of the above constraint is that in the vicinity of perfect learning the precursor might be focused around any set of discrete symmetric values, but not necessarily the ones that the clipped student has.

One of the conclusions concerning λ_l is that the law according which ϵ_g decays is independent of the exact value of the limit value, λ_l. It depends only on the ruler (binary/continuous), the specific strategy of learning (on-line/off-line) and the learning algorithm one uses. In the following we analyze all these variations.

IV. BINARY OUTPUT

A. On-line Learning

In an on-line learning scenario one can write equations of motion that determine the development of the order parameters as a function of α. The rate of convergence depends on the rule, F (Eq. 3) and the learning algorithm that one uses f (Eq. 4). Fine tunes are made by choosing the learning rate, η.

We analyze learning procedure in the case of binary rule,

$$S = sign(x), \tag{16}$$

where x is the local field and the generalization error as a function of ρ is known to be

$$\epsilon_g = \frac{1}{\pi}\cos^{-1}(\rho) \tag{17}$$

Although it was shown that using the "expected stability" algorithm that maximizes the generalization gain per example leads to an upper bound for the generalization ability, [17], we choose to concentrate on the so-called AdaTron or relaxation learning algorithm. This latter algorithm for zero stability, $\kappa = 0$, performs comparably well and unlike the "expected stability" algorithm does not require additional computations in the student network besides the updating of its weights, and the analysis is simpler as well [8].

The convergence to perfect learning depends on the learning rate, if it is too large perfect generalization becomes impossible. The transition from learnable situation to unlearnable occurs at η_c. In the following, in order to simplify the analysis; we choose a fixed learning rate, $\eta = 1$, which is below η_c in all scenarios.

We update the artificial continuous weight vector, $\vec{J}$. The updating is made as in Eq. 4 according to the following learning rule:

$$J_i^{\mu+1} = J_i^{\mu} - \frac{\eta}{\sqrt{N}}(\frac{\vec{J}^{\mu}\cdot\vec{\xi}^{\mu}}{\sqrt{N}})\xi_i^{\mu}\theta(-\frac{\vec{J}^{\mu}\cdot\vec{\xi}^{\mu}}{\sqrt{N}}S^{\mu}) \tag{18}$$

The equations for the order parameters with $\eta = 1$ are,

$$\begin{aligned}\frac{d\rho_J}{d\alpha} &= -\frac{\rho_J}{2\pi}\cos^{-1}(\rho_J) + \frac{1}{\pi}(1-\frac{\rho_J^2}{2})\sqrt{1-\rho_J^2},\\ \frac{dQ_J}{d\alpha} &= \frac{Q_J}{\pi}[\rho_J\sqrt{1-\rho_J^2} - \cos^{-1}(\rho_J)].\end{aligned} \tag{19}$$

In the limit $\alpha \to \infty$, one obtains a power law that describes the convergence of ρ_J and Q_J,

$$\begin{aligned}\rho_J &\sim 1 - 2(\frac{3\pi}{4})^2\frac{1}{\alpha^2}\\ Q_J &\sim Q_0(1-\pi^2(\frac{3}{4})^3\frac{1}{\alpha^2})\end{aligned} \tag{20}$$

Note: since we have a binary output unit, perfect learning is gained as soon as the angle between the vectors goes to zero independent of the student's norm.

The solution of Eq. 19 only describes the development of the continuous perceptron's overlaps. The next step is mapping the continuous precursor to the clipped one. Since in the case of binary ruler the student's norm converges to some unknown value, one way of choosing λ_l is simply "half the way" between the constrained values, i.e. $\lambda_{-L} = \lambda_{L+1} = infty$ and otherwise

$$\lambda_l = \frac{1}{L}(l-\frac{1}{2})\sqrt{\frac{Q_J}{T}}. \tag{21}$$

The development of the order parameter ρ_J, is independent of the norm Q_J. Using a limit set that scales with $\sqrt{Q_J}$, Eq. 21, ends up in ρ_W which depends on ρ_J but does not depend on Q_J. Hence, plugging into it Eq. 20, one can find the the asymptotic behavior of the generalization error, Eq. 17 in the limit $\alpha \to \infty$

$$\epsilon_g \propto \frac{\exp(-K(\lambda)\alpha^2)}{\alpha^{\frac{1}{2}}}, \tag{22}$$

where $K(\lambda) = \min|\lambda_l - \frac{l}{L}\sqrt{Q_J/T}|$.

We exemplify the aforementioned discussion in the diluted Ising perceptron. We use the limits as in 14 and assume $\lambda_1 = c\sqrt{Q_J/T}$. In that case

$$\rho_W = \frac{erf(a_+) + erf(a_-)}{\sqrt{T}\sqrt{9 - 3erf(a_0) - 3erf(a_+) + 3erf(a_-)}}, \tag{23}$$

where $a_\pm = \frac{\rho_J \pm c}{\sqrt{2T(1-\rho_J^2)}}$ and $a_0 = \frac{c}{\sqrt{2T(1-\rho_J^2)}}$. In the limit of large α one finds

$$\epsilon_g \propto \frac{\exp(-b_c\alpha^2)}{\alpha^{\frac{1}{2}}}, \tag{24}$$

where for $c \geq 1/2$ $b_c = \frac{c^2}{3\pi^2}$ and otherwise $b_c = \frac{(1-c)^2}{3\pi^2}$. One can see that choosing $c = 1/2$ results in a fastest decay of the generalization error.

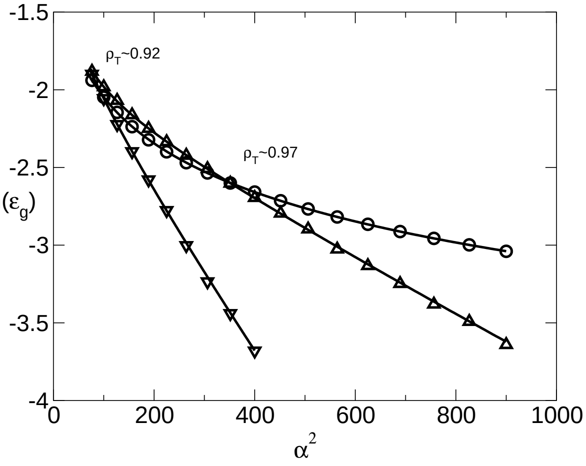

FIG. 1. Simulation results of $ln(\epsilon_g)$ of the continuous precursor ($\circ$) and of the clipped vector vs. α^2. The clipping is made according to the mapping in 14, where the results are for $\lambda_1 = 0.5\sqrt{Q_J/T}$ ($\bigtriangledown$) and $\lambda_1 = 0.3\sqrt{Q_J/T}$ ($\bigtriangleup$). error bars are smaller then symbols. Solid lines are the numerical integrals (Eq. 19). ρ_T refers to the point at which a transition occures between a superior performance by continuous/clipped perceptron, (see text).

The analytical results are compared with simulations on a teacher of the type of the diluted Ising perceptron with the following parameters; $\lambda = 0.5\sqrt{Q_J/T}$ and $\lambda = 0.3\sqrt{Q_J/T}$, see Figure 1. The initial conditions for the continuous student weight vector are $Q_J(\alpha = 0) = T = 2/3$ and $R_J(\alpha = 0) = 0$. The weight components were drawn out of a Gaussian distribution. We used $\eta = 1$, $N = 3000$ and each point was averaged over 50 samples. One can see in Figure 1 that the analytical results give by Eq. 23 and Eq. 24 are in agreement with simulations.

One can see that the super-exponentially decay is independent of the accurate value of λ. However, two important parameters do depend on the exact choice of λ. One is the decay rate, the factor $K(\lambda)$ in the large α limit. One can see, for instance, that the optimal limit, $\lambda = 0.5$ results in a faster decay than the limit $\lambda = 0.3$. The second is the exact α or the exact value of ρ_J at which the clipped version gives a better result than the continuous one. We named this value as ρ_T. For $\rho_J < \rho_T$ the clipping lowers the overlap ρ_J since the learning solution does not contain enough information about the real direction of the teacher, $\vec{W}^T$, so that clipping only leads the solution to forget a little about the learned pattern without bringing it closer to the exact solution. In the other region, when $\rho_J > \rho_T$, clipping becomes efficient because the learning solution is near the exact one. The numerical results of ρ_T according to the mapping, (Eq. 23), are $\rho_T \sim 0.92$ for $\lambda_1 = 0.5\sqrt{Q_J/T}$ and $\rho_T \sim 0.97$ for $\lambda_1 = 0.3\sqrt{Q_J/T}$, see Figure 1.

B. Clipped -Hebbian Learning

Ising perceptron, diluted Ising perceptron and all the binary units that are confined to discrete values exhibit a phase transition [5,22,23]. This known result was hard to achieved by a practical algorithm. One way to gain a perfect learning is to include the information of all the patterns simultaneously in the weights by using the Hebb learning procedure, Eq. 5. Such a learning will end up in a discrete student only in the limit $\alpha \to \infty$. The decay of the generalization error in that case is known, since it is exactly the same as having a continuous teacher, $\epsilon_g \propto 1/\sqrt{\alpha}$ [14,15]. In such a way the knowledge of having digital values is not used, one has a continuous student that happens to realize, after learning, that the values are constrained to a finite depth.

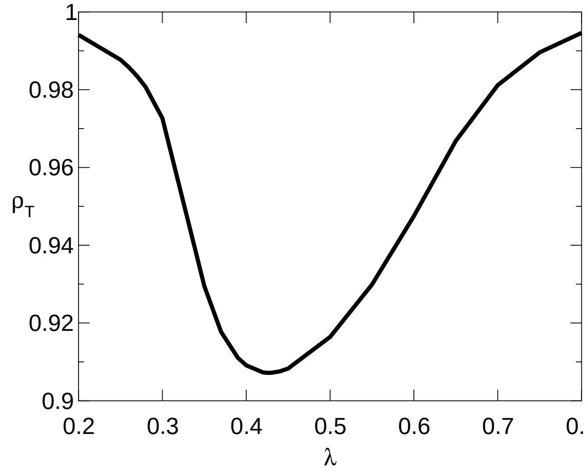

FIG. 2. Analytical results of ρ_T, as a function of the limit (λ_1) in the diluted Ising case. ρ_T stands for the continuous overlap value at which below/above it, a better generalization is achieved by the continuous/clipped perceptron.

The above mentioned procedure describes a way of using two vectors. A continuous one, which is evaluated according to the Hebb rule and a discrete student, obtained by clipping the continuous precursor according to Eq. 6. The latter mapping results in a better generalization error for large enough α.

We take for example the diluted Ising case. Given that the continuous student is normalized to be the same as the teacher one, $Q_J = T = 2/3$, one can find the exact point, ρ_T at which the clipped method results in a better generalization. This value depends on the limit one chooses. One can see that the limit λ_1, that results in a better generalization of the clipped version in smaller ρ, is $\lambda_1 \sim 0.43\sqrt{Q_J/T}$, see Figure 2. One might anticipate to get as a result $\lambda_1 \sim 0.5$, that was found to optimize the decay as $\alpha \to \infty$. However, the above value is determined by the distribution of the continuous weights in the beginning of the learning process, small α. In this regime the distribution of the weights is close to a Gaussian, and its tail influences the value of λ_1. The analysis above indicates that choosing α–dependent limits, $\lambda(\alpha)$ in this specific case might perform an even better generalization curve.

To conclude, the benefit from the clipping is evident only after the Hebb solution is near the exact one, after gaining large ρ. For optimizing the learning time, choosing the limits should be done cautiously. If the aim of the learning is to minimize the generalization error at the very end of the procedure, after a long learning process, than the best choice for the limit will be the “half the way” method, Eq. 21. However, to minimize the generalization error for a given finite α, the best value might be around $\lambda_1 \sim 0.425\sqrt{Q_J/T}$. These results suggest that it is possible to optimize the generalization error of the clipped perceptron by the choice of a dynamical $\lambda_1 = \lambda_1(\alpha)$.

V. LARGE SYNAPTIC DEPTH

In this section we examine the crossover of the generalization error in the presence of continuous weights as we increase the synaptic depth. As long as the synaptic depth $L < O(\sqrt{N})$, the generalization error still vanishes super-exponentially, Eq. 22, where the pre-factor decreases with L. For $L \geq O(\sqrt{N})$ the learning is characterized by the features of spherical constrained learning.

A first step towards the continuous case limit is to find out the change of the decay of the generalization error as a function of L. We focus on the binary unit in the on-line scenario. The analytic tractability of this model enables a profound study of the influence of the synaptic depth over the learning features.

In the last model the generalization decays super-exponentionall, $\epsilon_g \sim \exp(-K\alpha^2)$, (see Eq. 22). The factor K depends on the limits one chooses, λ_l. Hence, in order to keep on consistency, we use the abovementioned limits, (Eq. 21), in the different depths cases. We should emphasize at this stage that only one out of many super-exponentionall terms that arise from the asymptotic expansion of all the error functions (Eq. 10), was kept (Eq. 22). As soon as the deviations between different factors in the exponent are too small, one has to integrate all the terms together instead of neglecting all but one. Such a procedure results in a different type of decay, a power law instead of a super-exponentionall decay.

Analytical and simulations results of the generalization error in varieties of synaptic depths are presented in Figure 3. Simulations were carried out with $N = 630$ and each point is averaged over 100 samples. The insert shows the estimated slope K, (Eq. 22), as a function of the depth L. One can see that K decreases linearly with $1/L$. The deviation from the analytically predicted interplay for large α, $K \propto 1/L$, is probably due to finite N effects.

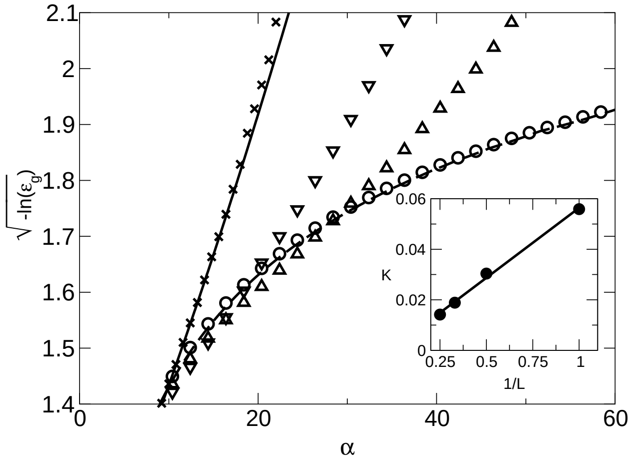

FIG. 3. Simulation results of $\sqrt{-ln(\epsilon_g)}$ in the case of $L = 1$, (diluted Ising) (×), $L = 2$ ($\bigtriangledown$), $L = 3$ ($\bigtriangleup$) and $L = 157$ ($\bigcirc$) versus α. The analytical results obtained by the numerical integration of Eq. 19 and Eq. 23 is presented for the ISing case (solid line). The dashed line is the analytical curve for $\sqrt{-ln(\epsilon_g^J)}$, were ϵ_g^J is the generalization error of the *continuous* student. Inset: The dependece of the prefactor $K(L)$ on the depth L, in Eq. 22. Solid line is the least squered fit, $K = 0.06/L$.

In the following we present argument supporting the statement that the generalization performance of finite depth machines coincide with the performance of continuous machines as soon as $L \sim \sqrt{N}$. This scaling is found by taking into account that: (a) The difference between two available values is of order of $1/L$. (b) The distribution of the continuous student values around the teacher's one is a Gaussian with a variance of $\sqrt{1-\rho_J^2} = 1/\epsilon_g^J$, where ϵ_g^J is the generalization error of the continuous student. Having a learning procedure (in the continuous space) in a finite dimension results in a generalization error, ϵ_g^J, which is different then the analytical predictions. The variance is of order of $\sqrt{1/N}$ [26]. Hence, an estimation to the order of the lower value that ϵ_g^J gets in a specific run will be $\sqrt{1/N}$. As a consequence, having a discrete machine of depth L when

$$\frac{1}{L} << \sqrt{1-\rho_J^2} \sim \epsilon_g^J \sim \sqrt{\frac{1}{N}} \qquad (25)$$

or $L >> \sqrt{N}$, gives the same results as those of continuous learning. Note that Eq. 25 is consistent with the mathematical constraint that was pointed out in section III when we discussed the continuous limit. The simulations shows indeed that in the case of $L = 157 >> \sqrt{N}$, were $N = 630$ the discrete vector's performance coincide with the analytical learning curve of the *continuous* student.

It is worth pointing out that a similar result was found when analyzing the possibility of learning from a discrete teacher by a discrete student using a general updating rule, [9]. The last analysis uses totally different argument results in the conclusion that only when the teacher's depth is of order $\sqrt{N}$, it is possible to learn the rule using an updating rule that depends on the discrete weights, i.e. only then it behaves as if we have a continuous machine.

VI. FINITE SYSTEMS - PERFECT LEARNING

The theoretical results presented in the previous chapters exhibit the typical behavior of the generalization error and the order parameters. The main result is the fast decay of the generalization error of the clipped perceptron to zero, Eq 22. In the case of teacher and student with continuous weights and finite N, the generalization error is always finite distance from zero, even in the asymptotic stage of the learning process. In contrast to the continuous case, the learning of a perceptron with discrete weights and finite N is characterized by a transition to perfect learning, as was found for the Ising perceptron, [11]. Performing simulations in that case results in a perfect learning in some stage, since in the clipping version the student knows exactly the teacher's optional values. Hence, the overlap becomes exactly one, $\rho_W = 1$, and the generalization error becomes exactly zero as well, $\epsilon_g = 0$.

In order to give an estimation to the number of steps needed for getting perfect learning, α_f, we use the following approximation valid in the $\alpha \to \infty$ regime, where we can give an analytical approximation to the interdependence of ρ_W and α. In addition, the minimal step before perfect learning is well defined: $\rho_W = 1 - 2/(LN)$ or $\epsilon_g \sim \sqrt{1/(LN)}$. Hence, we can find the interplay between α and N.

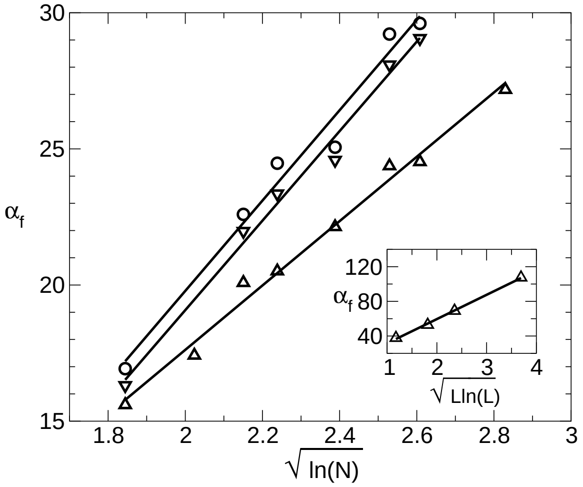

FIG. 4. Simulation results of α_f, the number of rescaled steps necessary to achieve a perfect learning vs. $\sqrt{\ln N}$. Simulations for diluted Ising perceptron, in the case of binary output unit, with $\lambda_1 = 0.4\sqrt{Q_J/T}$ ($\bigtriangledown$), $\lambda_1 = 0.5\sqrt{Q_J/T}$ ($\bigtriangleup$) and $\lambda_1 = 0.6\sqrt{Q_J/T}$ ($\circ$). Solid lines correspond to the linear fit of least square error. Inset: Simulation results of α_f vs. $\sqrt{L\ln L}$ for $N = 630$, $L = 2, 3, 4, 7$ and the limit values are chosen according to Eq. 21. Solid line is least squared fit.

In the binary output perceptron the generalization error falls down super-exponentially, Eq. 22. Hence, the perfect learning is determined by

$$\exp\left(-K(\lambda, L)\alpha^2\right) \sim \sqrt{1/(LN)}, \tag{26}$$

and since we found in the last chapter that K decays linearly with $1/L$ we can derive α_f from the last equation, $\alpha_f \sim \sqrt{L\ln LN}$. This result indicates quantitatively that for any chosen limit, λ_l, the number of learning step necessary to achieve perfect learning is finite as long as N and L are finite.

Figure 4 presents results of α_f obtained in simulations for the diluted Ising perceptron with $c = 0.4$, $c = 0.5$, and $c = 0.6$, (Eq. 23, 24). Results were averaged over $M(N)$ training sets, were values of $M(N)$ ranging from 5000 to 20 in accordance to N which is varied between 30 and 9000. To get results in lower dimension, N, we averaged over a larger number of simulations, M.

One can see from the obtained values of $\alpha_f(N, c)$ in Figure 4, that the last quantity is indeed linear in $\sqrt{\ln N}$. Note that the obtained slope in Figure 4 for $c = 0.4$ and $c = 0.6$ is the same as it is expected since b_c is symmetric around $c = 1/2$. In the inset, one can see that $\alpha_f(L)$ in the case of $N = 630$, indeed increases linearly with $\sqrt{L\ln L}$. As $L \to \infty$ an infinite number of examples are needed for perfect learning, there is a crossover to the spherical case as was discussed in the previous chapter.

Small deviations from a straight line in Figure 4 are expected to be a consequence of the following approximations: (a) We took as an analytical curve (Eq. 26) only the asymptotic function which is an expansion valid in infinite α. (b) We neglected the polynomial corrections in Eq. 26 such as $1/\sqrt{\alpha}$. (c) We derived Eq. 26 from the analytical calculation of $\rho_J(\alpha)$. The latter quantity itself is influenced by finite size effects. Extensive numerical simulations show that the corrections are linear in $1/N$ [24–26] and hence they are negligible after clipping and getting ρ_W (As in Eq. 12).

As was shown in previous chapters, $c = 0.5$ gives the best performance in the asymptotic learning procedure, lower α_f for all N, and it is confirmed in our simulations, Figure 4 . In the thermodynamic limit $N \to \infty$, $\alpha_f \to \infty$ as expected.

VII. CONTINUOUS UNIT

We now study the case of continuous output perceptrons with finite depth. As long as one uses a continuous activation function, the generalization error decreases exponentially, (see for instance [12,13,18]). In order to learn a rule which is defined by a finite depth vector, we used a spherical vector for the student weight vector, $\vec{J}$, and clipped it in order to have a digital student weight vector $\vec{W^S}$. The updating of the spherical student weight vector is done according to the gradient descent method as usual:

$$\vec{J}^{\mu+1} = \vec{J}^{\mu} - \frac{\eta}{\sqrt{N}}\nabla_{\vec{J}}\,\epsilon(\vec{J}^{\mu}, \vec{\xi}^{\mu}) \tag{27}$$

The error $\epsilon(\vec{J}^{\mu}, \vec{\xi}^{\mu})$ measures the deviation of the student from the teacher's output for a particular input $\vec{\xi}$. The generalization error of a student is defined as the averaged error

$$\epsilon_g =< \frac{1}{2}[S(\vec{J}, \vec{\xi}) - S(\vec{W}^T, \vec{\xi})]^2 >_{\vec{\xi}}\,. \tag{28}$$

Since the learning features of all kinds of the continuous transfer functions are more or less the same, we chose to concentrate in the "sin" activation function

$$S = \sin(kx), \tag{29}$$

The periodic activation function, sin, was found to be learnable given that the period k is small enough [13]. In the following we will simplify our analysis by taking k=1 and the learning rate $\eta = 1$. Since the learning curves of the continuous version are the same as if there was a rule defined by a continuous teacher, (having the finite depth limitation is merely a special case of the spherical constraint), and the learning rate we chose is small enough we find that perfect learning is an attractive fixed point in both scenarios.

Linearizing the equations of motion around those fixed points results in the following form (which holds for all the continuous transfer functions):

$$
\begin{aligned}
R_J &= 1 - \frac{c_1}{\det V} V_{22} \exp(\gamma_1 \alpha) + \frac{c_2}{\det V} V_{12} \exp(\gamma_2 \alpha) \\
Q_J &= 1 + \frac{c_1}{\det V} V_{21} \exp(\gamma_1 \alpha) - \frac{c_2}{\det V} V_{11} \exp(\gamma_2 \alpha)
\end{aligned}
\tag{30}
$$

The two eigenvalues of V, γ_1, γ_2, are both negative. The constants c_1, c_2 are determined from the numerical solution of the equations of motion.

In order to get a description of the discrete learning one has to use the mapping relations as in Eq. 6. The generalization error of the finite depth student directly depends on the order parameters, as can be found by taking the average over the local fields distribution, Eq. 28. The general result of this calculation at the $\alpha \to \infty$ regime is

$$
\epsilon_g \sim \exp\left(-C_0 e^{|K|\alpha}\right), \tag{31}
$$

were K and C_0 depend only on the learning rate ,η, the limits one chose, λ_l and the specific activation function. In the following we examine this result in the diluted Ising case.

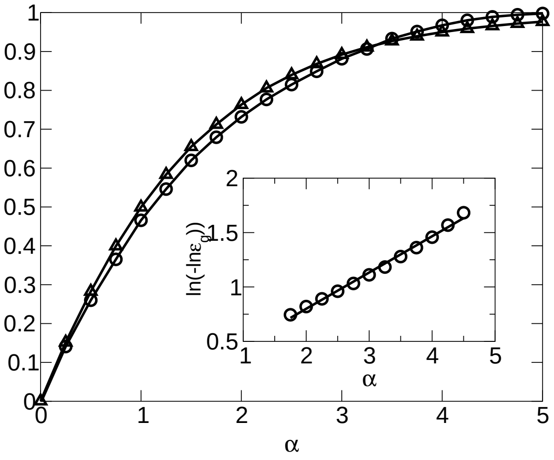

FIG. 5. Simulation results of ρ_J ($\triangle$) and ρ_W ($\circ$) vs. α in the diluted Ising case. Solid lines are the numerical integrals (Eq. 15, 32). Inset: $\ln(-\ln(\epsilon_g))$ vs. α obtained in simulations ($\circ$) with $N = 3000$. Solid line is least squared linear fit, the slope was found to be 0.33.

We performed simulations in the diluted Ising case, when the transfer function is sin. The development of the continuous order parameters in that case is described by the following equations of motion,

$$
\frac{dR_J}{d\alpha} = \frac{1}{2}\left[(R_J{+}1)D_+ {-} 2R_J e^{-2Q_J} {-} (R_J{-}1)D_-\right]
$$

$$
\begin{aligned}
\frac{dQ_J}{d\alpha} &= \left[(R_J{+}Q_J)D_+ {-} 2Q_J e^{-2Q_J} {-} (Q_J{-}R_J)D_-\right] \\
&\quad + \frac{1}{8}\left[2\left(e^{-2Q_J} - e^{-2} - E_- + D_+\right)\right. \\
&\quad \left. + 3 {-} D_-^4 {-} 2D_- + \left(2E_+ - e^{-8Q_J} - D_+^4\right)\right]
\end{aligned}
\tag{32}
$$

with $D_\pm = e^{-(1+Q_J \pm 2R_J)/2}$ and $E_\pm = e^{-(1+9Q_J \pm 6R_J)/2}$. As $\alpha \to \infty$, one gets two eigenvalues, $\gamma_1 \sim -0.30$, $\gamma_2 \sim -0.69$. Using Eq. 15, rescaling R_W and Q_W by the teacher's norm, 2/3, and taking the limit value, λ, to be the one that yields the faster decay at the large α regime, $\lambda = 0.5\sqrt{Q_J/T}$. Collecting everything we have

$$
\begin{aligned}
R_W &\sim 1 - \frac{\exp(-0.15\alpha)}{2\sqrt{\pi}K_1} \exp\left(-K_1^2 e^{0.30\alpha}\right) \\
Q_W &\sim 1 + \frac{\exp(-0.15\alpha)}{\sqrt{\pi}K_1} \exp\left(-K_1^2 e^{0.30\alpha}\right),
\end{aligned}
\tag{33}
$$

where K is determined by the initial conditions. The generalization error as a function of the discrete parameters is

$$
\epsilon_g = \frac{1}{2}\left[1 {-} d_- {+} d_+ {-} \frac{1}{2}\left(e^{-2Q_W} {+} e^{-2}\right)\right] \tag{34}
$$

with $d_\pm = e^{-(1+Q_W \pm 2R_W)/2}$. Expanding the last equation around $R_W \to 1$ and $Q_W \to 1$, we obtain that the generalization error decays very fast, $\epsilon_g \sim \exp\left(-K_1^2 e^{0.30\alpha}\right)$.

We ran simulations with $N = 3000$ and averaged over 10 samples. In Figure 5 the development of the discrete as well as the continuous order parameters as a function of α are presented. The solid lines are the analytical numerical integrals of Eq. 32. Note, the transition in this scenario from a poor generalization of the clipped version comparatively to that of the continuous one, to a situation in which the clipped version has a better performance, occurs in the same $\rho_T \sim 0.92$ as in the binary unit. This quantity is related to the clipping rule and it is independent of the specific transfer function one tries to learn.

The inset of Figure 5 shows the unique decay of the generalization error, in order to get linear line we plotted $\ln(-\ln\epsilon_g)$ as a function of α. According to the above analysis the slope should be 0.30 and we obtained in simulations 0.33 ± 0.01. It is in good agreement, considering the fact that we are dealing with an approximation which is valid only in the $\alpha \to \infty$ and simulations results are in finite α. The generalization error of the clipped version for larger α ($\alpha > 7$ in our case) gives better results than those predicted by the analysis, its values are exactly zero due to finite size effects discussed in chapter V.

Following the same arguments used in order to find an estimation to the number of examples needed for gaining perfect learning, one finds that in the case of continuous output $\alpha_f \sim \ln(\ln N)$. It is obvious from the analytical calculations and the simulations above that clipping a continuous vector in order to learn a finite depth teacher results in an extremely fast learning. The learning in finite dimension is characterized by α_f, above which one gets perfect learning of the discrete vector. All those unique characteristics of the discrete learning disappears as soon as the weight depth is of order of $\sqrt{N}$ as was found in chapter VI.

VIII. CONCLUSIONS

In this paper, we presented an analysis of the simplest neural network, the perceptron, that learns from examples given by another perceptron, the teacher, which is confined to a discrete space. In fact, we used two students, a continuous precursor and its clipped version.

We analyzed the new set of order parameters arising from the clipping method. We discussed the issue of how to clip and what set of limits, λ_l, is the best choice. We found that it depends specifically on the kind of optimization one imposes. We showed that after reaching some overlap, ρ_T, a transition occurs and the clipped version results in a better performance then the non-clipped one. If one is interested in optimizing the learning in the sense of getting a better performance as soon as possible, then the minimizing ρ_T limits are the ones needed. However, if by optimizing one tries to get the fastest decrease possible in the $\alpha \to \infty$ regime then the best choice is 'half the way', in-between the values. As we mentioned before, it is possible to have a dynamic set of values that interpolates during the learning process between both values. We left this issue out of the scope of this paper.

As one can see from the definitions in Eq. 2, it is only natural to choose the continuous weight vector not to be the one which is constrained to a hypersphere but a vector which is constrained to a hypercube space. It was shown that in the case of storing random patterns pre-training a continuous student whose weight vectors constrained to the volume of a hypercube results in a better performance [7]. It remains as an open question what is the quantitative benefit that one can gain in a learning procedure by using the cubical constrained and if a learning strategy could be designed which fulfills this constraint.

We studied the case of a very large L and show a scaling relation between L and N arises from the analysis. For $L \sim O(\sqrt{N})$ the learning curve is the one that is typical to the continuous case. However, it should remain clear that learning is the same as having a continuous student unless $\alpha \to \infty$, $\rho_J \to 1$. In that regime the fast decay that characterizes the clipped learning appears. All digital computers actually correspond to such a situation, where all available properties have a finite representation. The machine is using some kind of clipping by rounding the numbers. The differences, as predicted here, can be significant only in the $\alpha \to \infty$ regime or small depth. Visualizing them is usually impossible since they are smaller than the measurements scale.

ACKNOWLEDGMENTS

Discussions with W. Kinzel and M. Biehl are acknowledged. We would like to thank M. Biehl for carefully reading of the manuscript. The research is supported by the German Israel foundation and the Israel Academy of Science.

[1] J. Hertz, A. Krogh and R. G. Palmer, *Introduction to the theory of Neural Computation* (Redwood City, CA: Addison-Wesley, 1991).

[2] T. L. H. Watkin, A. Rau and M. Biehl, Rev. Mod. Phys. **65** 499-556, (1993).

[3] H. Gutfreund and Y. Stein, J. Phys. A **23**, 2613, (1990).

[4] I. Kanter, Europhys. Lett. **17**, 181, (1992).

[5] R. Meir and J. F. Fontanary J. Phys. A **25**, 1149, 1992.

[6] J. Schietse, M. Bouten, and C. Van den Broeck Europhys. Lett. **32**, 279, (1995).

[7] M. Bouten, L. Reimers and B. Van Rompaey, Phys. Rev. E **58**, 2378 (1998).

[8] M. Biehl and P. Riegler, Europhys. Lett. **28**, 525, (1994).

[9] W. Kinzel R. Urbanczik, J. Phys. A **31**, L27-30, (1998).

[10] C. Van den Broeck and M. Bouten Europhys. Lett. **22**, 223, (1993).

[11] M. Rosen-Zvi J. Phys. A **33**, 7277, (2000).

[12] D. Saad and S. A. Solla, Phys. Rev. Lett **74**, 4337, (1995) and Phys. Rev. E **52**, (1995).

[13] M. Rosen-Zvi, M. Biehl and I. Kanter, Phys. Rev. E **58**, 3606, (1998).

[14] F. Vallet, Europhys. Lett. **8**, 747 (1989).

[15] F. Vallet and J-G. Gailton, Phys. Rev. A **41**, 3059 (1990).

[16] M. Opper, Phys. Rev. Lett 77, 4671 (1996)

[17] O. Kinouchi and N. Caticha, J. Phys. A **25**, 6243, (1992).

[18] M. Biehl and H. Schwarze, J. Phys. A **28**, 643, (1995).

[19] M. Bouten, A. Komoda and R. Serneels, J. Phys. A **23**, 2605 (1990).

[20] H. Gutfreund and Y. Stein, J. Phys. A **23**, 2613 (1990).

[21] D. Malzahn Phys. Rev. E **61**, 6261 (2000).

[22] E. Gardner and B. Derrida, J. Phys. A **22**, 1983, (1989).

[23] W. Kinzel Phil. Mag. B **77**, 1455, (1998).

[24] B. Derrida, R. B. Griffiths and A. Prügel-Bennett, J. Phys. A **24**, 4907, (1991).

[25] A. Buhot, J-M. Torres Moreno M B Gordon, Phys. Rev. E **55**, 7434, (1997).

[26] P. Sollich and D. Barber, Europhys. Lett. **38**, 477, (1997).