Active-absorbing transition in the linear perceptron model

Supervisor:
Pierfrancesco Urbani

Author:
Francesco Morri

Co-Supervisor:
Alfredo Braunstein

Master in Physics of Complex Systems

Academic Year 2021/2022

Abstract

Recently, many studies have been conducted on systems of particles suspended in liquid, since their physics is related to glassy systems. These systems showcase a particular type of out of equilibrium phase transition, called *absorbing state phase transition*.
In order to study this phenomenon, many simple models have been proposed to simulate the dynamics, and they have been characterized through their critical exponents at the transition, suggesting that they belong to the *Manna universality class*.
We focused on simulating a mean field model, the spherical perceptron, using finite learning rate gradient descent on the hinge loss, which reproduces a dynamic with an absorbing phase transition. We conducted extensive numerical simulations to extrapolate the critical exponents of the activity (the number of negative gaps) at the transition point, in order to find if also this system belongs to the same universality class as the other algorithms proposed.
We characterize the critical behavior at the transition and find numerically the critical exponents associated to the order parameter, the activity as well as the associated susceptibility.

Contents

List of Figures

Chapter 1

The Absorbing State Phase Transition

Recently there has been much interest in the study of colloidal suspensions, both from the experimental point of view and from the theoretical one. In particular systems of particles suspended in liquids undergoing periodic shearing have shown interesting characteristics.
These kind of systems can be studied using the shear amplitude as a control parameter ([3]), in order to drive the system from an *active state*, where the trajectories of the particles are chaotic and irreversible, to an *absorbing state*, where trajectories are reversible. This out of equilibrium phase transition, that occurs at a critical value of the shear amplitude, is called *absorbing phase transition*.
This kind of transition is not something new, as it is common in systems belonging to the Manna universality class and described by the directed percolation field theory ([22], [23], [24]), which are used to model a vast range of natural phenomena.
The absorbing phase transition point is characterized by the self-organized criticality ([27]) of the system: one can define a diverging correlation length as well as diverging susceptibility. At the same time the model displays hyperuniformity, meaning a suppression of density fluctuations. All these phenomena appear out of equilibrium, since the phase transition point separates two non-equilibrium phases.
In order to study this kind of systems it is important to define the order parameter characterizing the transition: since in the active region the system is 'moving' (with different meanings, specific for each models), while, once it enters the absorbing state it 'stops' (again with the meaning changing from model to model), it is reasonable to quantify the amount of movement, or **activity**. This parameter should be different from 0 in the active region, and go to 0 at the critical point, remaining then 0 after the transition.

In order to simulate these kind of systems, simple algorithms have been introduced ([15], [14], [13], [25]), based on *qualitative* rules more than potential based ones. The idea of all algorithms is to construct an out of equilibrium dynamics that, as a function of a control parameter, can display an asymptotic active state where the system keeps moving in phase space, and an absorbing state where, after some transient time, the dynamics stops at a point in phase space.
We will focus and describe more in depth [15] and [13].
The algorithm presented in [15] is called *random organization* (RO), and it aims at recreating the dynamics of sheared particles experiments using a completely random

protocol. The idea is the following: a set of spheres are placed inside a box, which is periodically sheared with a constant amplitude; after every shearing the overlapping particles are displaced of a small random amount, while non overlapping spheres are left unchanged. The results obtained from the first simulation of this algorithm ([15]) showcased that it does not belong to the directed percolation universality class.
Building on these results, another algorithm, is proposed and studied in [13]. The different protocol adds a deterministic kick to the spheres, together with the random one, and is called *biased random organization* (BRO). This is done introducing a control parameter (δ) that controls the ratio between the two; when the algorithm is driven only by the deterministic kick, we will refer to it as *fully biased random organization* (fBRO).
The precise definition of the BRO algorithm is not entirely clear from [13]. However, we believe that the fBRO algorithm coincides with what we are going to describe and use in the following sections[1]. Anyway, it may be possible that there is a discrepancy between the algorithms, but qualitatively, they both display an absorbing phase transition as a function of the packing fraction.
As we mentioned, the two contributions in BRO are weighted by a parameter called δ, and have a magnitude dependent on another parameter, ϵ. An important difference between [13] and [15] is that in the former it is no longer used the shearing amplitude as a control parameter, but the density of the spheres; in fact there is actually no shearing at all.
The update rule can be written as:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \epsilon \left[\sqrt{1-\delta}\nu_i \Theta\left(n_i\right) + \sqrt{\delta} \sum_{j \neq i} \Theta(-h_{ij}) \frac{r_i - r_j}{|r_i - r_j|} \right] \tag{1.1}$$

where we are using $h_{ij} = (r_i - r_j) - (R_i + R_j)$ (r is the position of the sphere, R the radius), and $n_i = \sum_{j \neq i} \Theta(-h_{ij})$ is the number of overlaps. In the first term, the random displacement, ν_i is a random vector of typical magnitude 1[2], with $\Theta(n_i)$ enforcing the presence of this term only if there are overlaps. In the second term we have the deterministic displacement, where again we use the Heaviside function to sum only over overlapping particles. In the limit $\delta = 0$ we return to the RO algorithm, while for $\delta = 1$ we obtain the fBRO algorithm.
The aim of [13] is to show that BRO is in the *Manna universality class*. Since now we have defined the model and protocol, we can also identify more precisely the order parameter (activity) that we have previously introduced in a very general way: it will be the number of particles overlapping (or active particles), since those are the ones that will contribute to the dynamics at each step.
Then, in order to extrapolate the universality class, the authors study the critical exponents of the activity and relaxation times. The results presented mostly refer to fBRO and RO, so we will focus on these two algorithms. The critical exponents found seems to be compatible with the Manna universality class, both for RO and BRO. It is worth mentioning though, that the exponents computed in [15] were not compatible with the Manna exponents, but they simulated only 1 and 2 dimensions, while in [13] the results refer to simulations in 3 dimensions.

[1] More precisely, we know the algorithms coincide for pairwise overlaps, it is unclear if this is true for three or more overlaps

[2] In [13] they firstly introduce a random displacement with a random magnitude and later the magnitude seems to be fixed

Another line of research, always related to the simulation of systems of spheres, focused on a different problem, namely reproducing and studying system of spheres interacting with a repulsive potential.
We may start by introducing standard results regarding the protocols with which these system are simulated, since the dynamic depends on them.
Starting from an important work in the field, in [12] the authors introduce three different local potentials to describe the interaction between pairs of spheres, of the form:

$$V(r_{ij}) = \begin{cases} \frac{\epsilon}{\alpha}(1 - \frac{r_{ij}}{\sigma_{ij}})^{\alpha} & r_{ij} < \sigma_{ij} \\ 0 & r_{ij} \geq \sigma_{ij} \end{cases} \tag{1.2}$$

with $\alpha = 2, 3/2, 5/2$. In Eq. 1.2 we call r_{ij} the distance between the centers of spheres i, j and σ_{ij} the sum of the radii.
Let us notice that the case of a *linear potential*, meaning $\alpha = 1$, is not considered, this peculiar situation (the force is independent from the amount of overlap of the spheres), will be explored later.
The evolution of the system is driven by gradient descent towards the nearest energy minimum. The result of this dynamics depends on the value of the density of the spheres:

$$\begin{cases} \phi < \phi_c & \text{zero energy configuration: no overlaps} \\ \phi > \phi_c & \text{local minimum at positive energy: spheres overlap} \end{cases} \tag{1.3}$$

and it is important to notice that in *both* situations we reach, at the end of the simulations, for long times, configurations where the spheres are still, since in both cases the gradient of the potential will be zero. The critical point separating the two phases is called the *jamming transition*: the particles either reach mechanical stability (above the critical density) or they find a zero energy state (below the critical density); this transition has been studied a lot in the past 20 years, and a systematic approach using tools of disordered and glassy systems has been developed only recently ([18]).

A different and interesting result is obtained if we use a linear potential, as shown in [10]. The main difference is that for the class of potentials shown in Eq. 1.2, $\alpha > 1$ means that the local potential is convex, while $\alpha = 1$ is exactly at the boundary between convex and concave. This type of potential is not very common in physics simulation, while is well known in the machine learning community with the name of **hinge loss**. This creates a situation where in local energy minima for the full systems, the configurations will have both overlapping spheres and spheres only touching (*kissing spheres*). This is possible since these kissing spheres may sustain a finite amount of force. The result is that a minimum in the phase space is now an angular point[3].
Using a protocol with finite time step (or finite learning rate) to drive the systems will make it constantly moving in the denser ($\phi > \phi_c$) phase, since, even though local minima exists, it is impossible to find one with this type of dynamics (Eq. 1.4).

$$\begin{cases} \phi < \phi_c & \text{arrested state} \\ \phi > \phi_c & \text{active state} \end{cases} \tag{1.4}$$

[3]The simplest way to think about a low dimensional version of local minima on the hinge loss is to think about the landscape of $|x|$. This function has a minimum in $x = 0$, yet it is singular

Therefore, finite learning rate gradient descent on the hinge loss gives rise to a phenomenology in which at high density the system is in an active state and at low density the system finds an absorbing state. The dynamical phase transition between these two phases is purely out of equilibrium and it has the same qualitative features of an absorbing phase transition, as the one obtained from simulations done with fBRO and RO.

As for the jamming transition, where it is possible to introduce simple mean field models in order to study the critical point, we will follow the same strategy to study the absorbing phase transition.
In the next section we will introduce the mean field model we are going to use, the spherical perceptron, and a protocol to simulate its dynamics, gradient descent over the hinge loss, which can be directly mapped to the biased random organization algorithm for hard spheres.

Chapter 2

A Mean Field model: the Spherical Perceptron

The perceptron is one of the first examples of a machine learning architecture ([16]). It consists in a set of weight, represented as a vector, and the optimization will update these weights based on some constraints. It is mostly applied to classification problems, with the limitation that the data points must be *linearly separable*, since the solutions it can find are only hyperplanes dividing the points.
We are going to study a more constrained version of this model, the so-called **random spherical perceptron**. In this model the weights' vector has fixed magnitude, hence it is spherically constrained.
More specifically, we consider N variables $x_i \in \mathbb{R}$ with the condition:

$$\sum_{i=1}^{N} x_i^2 = N$$

which also implies that each variable is typically of order 1.
We now introduce the set of constraints or, to keep the analogy with machine learning problems, *patterns*: $\boldsymbol{\xi}^\mu$ with $\mu = 1, \dots, M$. Each constraint is a random N-dimensional vector $\boldsymbol{\xi}^\mu = \{\xi_1^\mu, \dots, \xi_N^\mu\}$, with each component being a Gaussian random variable with zero mean and unit variance (all *iid* variable). The number of constraints is linked to the dimensionality by $M = \alpha N$, and α is a control parameter for the system. We also notice that the parameter M is not needed, being dependent on N.
Let us now define the gap variable:

$$h_\mu = \frac{\boldsymbol{\xi}^\mu \cdot \mathbf{x}}{\sqrt{N}} - \sigma \tag{2.1}$$

where we divided the scalar product by $\sqrt{N}$ in order to obtain a quantity of order 1. σ is another control parameter of the system, and continuing with the machine learning parallelism, it would be the *bias*, used to enforce that the hyperplane is far enough from each point: large positive values of σ make the problem harder, negative values make it easier. The optimization problem is formulated in terms of these gap variables:

$$h_\mu > 0 \quad \forall\, \mu = 1, \dots, \alpha N \tag{2.2}$$

where with this set of inequalities we are enforcing the satisfaction of all constraints. Fig. 2.1 shows a simple example of a spherical perceptron in 2D with 3 constraints, where the region satisfying all constraints is the one with all three colors overlapping.

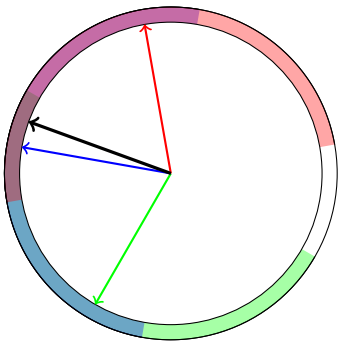


Figure 2.1: Graphical representation of three constraints in 2D: the colored part of the circle represent the region where each constraint would be satisfied, so the solution of this problem would be anywhere in the part of the circle where the colors overlap. In this case we are considering $\sigma = 0$.

In order to quantify the accuracy of the perceptron we use a loss function:

$$V_{HL} = \sum_{\mu=1}^{\alpha N} |h_\mu| \Theta(-h_\mu) \tag{2.3}$$

This function is known as *hinge loss*, and we want to stress that is a linear loss.
The optimization of this algorithm is carried out using gradient descent on this loss function, which gives:

$$\begin{aligned} x_i(t+1) =& x_i(t) - \eta \frac{\partial V_{HL}}{\partial x_i} \\ =& x_i(t) + \eta \sum_\mu \frac{\xi_i^\mu}{\sqrt{N}} \Theta(-h_\mu) \end{aligned} \tag{2.4}$$

where we also introduced the third control parameter of the system η, called *learning rate*. We can quickly generalize the algorithm and add a random part, obtaining:

$$x_i(t+1) = \frac{\sqrt{N}}{|\mathbf{x}|} \cdot \left[x_i(t) + \eta \left(\epsilon \nu_i \Theta(n_i) + (1-\epsilon) \sum_\mu \frac{\xi_i^\mu}{\sqrt{N}} \Theta(-h_\mu) \right) \right] \tag{2.5}$$

where ν_i is a vector composed of random *iid* variables extracted from a Gaussian distribution, and as in Eq. 1.1 $n_i = \sum_\mu \Theta(-h_\mu)$. Also we underline that even when the random kick is present, it is not a Langevin type equation. In front of everything we add a projector to keep the vector always on the sphere.
From Eq. 2.5 and Eq. 1.1 we can easily see how the BRO algorithm for spheres can be mapped on the gradient descent on the hinge loss for the perceptron:

- σ is the sum of the radii of the spheres (which are all of the same size)
- $\epsilon \to \eta$
- $\delta \to \epsilon$
- $\frac{\boldsymbol{\xi}_i^\mu}{\sqrt{N}}$ has the role of the direction of overlap

This model is of particular interest in relation to the hard spheres simulations not only for the direct mapping between the two. Using the control parameters of the perceptron we can drive the system between two phases: one where all constraints can be satisfied (SAT phase) and one where there is no possible solution (UNSAT phase). We can summarize the dynamics in the same manner as Eq. 1.4:

$$\begin{cases} \sigma < \sigma_{crit} & \text{SAT} \\ \sigma > \sigma_{crit} & \text{UNSAT} \end{cases} \tag{2.6}$$

The transition between these two regions can be thought as a jamming transition ([19], [26], [17]) and it can be studied in the exact same way as the spheres model. Still, a good parallelism using linear potential or cost function is missing.
This is a fundamental difference, as the form of the cost function influences the UNSAT region, leaving the SAT one unchanged, since it is an absorbing state and the systems just stops in a flat region without 'feeling' the effects of the form of function.
The landscape for the perceptron with a linear cost function has been extensively studied ([9], [11]), but in our simulations we are going to use gradient descent with a finite learning rate, meaning that the system can not actually set in a minimum, it will continue to move (hence resulting in an *active* state).

There is a secondary, but still important, reason to study the perceptron in depth. Being a rather simple a small model, it is possible to simulate many parameters configurations quite cheaply. Even though is not the focus of this work, this may allows us in the future to use these results to understand better the learning dynamics underlying more complex algorithms. In particular the role of noise, which is a fundamental part of many learning algorithms, can be explored here in this simple settings. This kind of analysis would fit in the very recent and rich field of studying machine learning algorithm using statistical physics tools. Many important results have been already obtained in this direction ([1], [2], [4], [5], [6], [7], [8]), but there are still many unknowns.

Chapter 3

The dynamics: implementation of the algorithm

We wrote a C++ program for running the simulations in parallel using OpenMPI. In order to optimize the system we implemented the algorithm described in Eq. 2.5, where we can notice that the gradient is nothing more than a vector-matrix product:

$$\begin{aligned}\frac{\partial V_{HL}}{\partial x_i} =& \frac{1}{\sqrt{N}} \sum_{\mu} \xi_i^{\mu} \Theta(-h_{\mu}) \\ \nabla V_{HL} =& \frac{1}{\sqrt{N}} \hat{\boldsymbol{\xi}} \cdot \mathbf{O} = \frac{1}{\sqrt{N}} \begin{bmatrix} \xi_0^0 & \dots & \xi_0^{\alpha N} \\ \vdots & \ddots & \vdots \\ \xi_N^0 & \dots & \xi_N^{\alpha N} \end{bmatrix} \begin{bmatrix} \Theta(-h_0) \\ \vdots \\ \Theta(-h_{\alpha N}) \end{bmatrix}\end{aligned} \tag{3.1}$$

This way we obtain a vector and we can simply subtract it to the weights' vector $\mathbf{x}$ (considering of course the learning rate η). This is an important simplification, since the computing of the gradient is usually the most expensive part of the simulation, and using scientific libraries (specifically we used GSL [20]) this product is as optimized as possible. The matrix composed of the constraints vectors is always the same and does not need to be computed each iteration, so we only need to compute the gaps vector $\mathbf{O}$, which simply consists in computing all gaps and verify which ones are not satisfied. This computation is also important to compute the activity[1] for the system, defined as:

$$A(t) = \frac{1}{\alpha N} \sum_{\mu} \Theta(-h_{\mu}) \tag{3.2}$$

which is nothing else than the normalized number of unsatisfied constraints.
In Alg. 1 we show how we compute the vector $\mathbf{O}$ and simultaneously also the activation. Here we also check if all the constraints are satisfied, in which case the simulation is ended.

After this first step, we can compute the gradient. In Alg. 2 we show the implementation of Eq. 3.1, with the generalization to include a random displacement (in order to work with the algorithm BRO). The last step is to set the magnitude of $\mathbf{x}$ to N in order to keep it on the hypersphere. Notice that, with respect to Eq. 2.5, we do not multiply the random part with the product of unsatisfied constraints. This is because we already have a control on whether the problem is solved or not, so if the simulation arrives at the step of computing the gradient it means that there still are unsatisfied constraints.

[1] In the spheres model, the activity is defined as the amount of spheres that have at least one overlap. Here instead we define it as the amount of negative gaps

Algorithm 1 Pseudo-Code to compute the unsatisfied constrain (gaps)

```
function COMPUTE-GAPS(...)
    h ← ξ̂ · x − σ                          ▷ Compute gaps
    O ← 0                                   ▷ Prepare overlaps vector
    act ← 0
    for i = 0 to i = len(h) do
        if h_i < 0 then
            O_i = 1
            act+ = 1
        end if
    end for
    save-activity(act)
    if act == 0 then
        stop-simulation
    end if
end function
```

Algorithm 2 Pseudo-Code to compute the gradient and update the vector $\vec{x}$

```
function GRAD-STEP(...)
    ∇ ← N(0, 1)          ▷ Use grad vector to hold the random part of the step
    ∇ ← η(ε∇ + (1 − ε)ξ̂ · O)                ▷ Compute gradient
    x ← x + ∇
    x ← x · √len(x) / ||x||
end function
```

Eventually we can put everything together, as shown in Alg. 3: in every simulation we run $N_{samples}$ samples for each thread, running also multiple threads. For each sample we defined a *max-time*, after which we stopped the simulation. The value of *max-time* and $N_{samples}$ depend on the size of the vector $\mathbf{x}$ and on how close to the jamming transition we are.
In the next section we will discuss the parameters used for the simulations and the results.

Algorithm 3 Pseudo-Code of the complete algorithm (here we skip all the parts regarding saving the data)

```
for i = 0 to i = N_threads do
    for j = 0 to j = N_samples do
        for time = 0 to maxtime do
            COMPUTE-GAPS()
            GRAD-STEP()
        end for
    end for
end for
```

Chapter 4

Simulations and Results

We ran many simulations testing different configurations of parameters. In all the testing we always kept fixed the value of α, the ratio of number of weights vs. constraints, to 3. We then have three other parameters to vary: σ, η and ϵ. The first one is the one we explored the most, since it is the parameter that allow us to sample the critical region between jammed and unjammed, and has a similar effect as the spheres density in RCP. The other two are important in order to understand properly the universality of the system at criticality. The value of η will let us compare these results with the analysis done in [13]. On the other hand varying ϵ can give us information about the similarities and differences of fBRO and BRO, which may even help understand better the role of randomness in learning algorithms. We mostly focused on running numerical simulations of the fully biased algorithm, firstly to have a good amount of results to study the algorithm and secondly because of time constraints. Nevertheless we also run some tests using BRO, obtaining some interesting behaviours.
All the considerations about the results should be done in the limit of infinite dimensions ($N \to \infty$), in order to understand and extrapolate the behaviour in such limit we simulate different sizes of the weights' vector. We started from the relatively low dimension of 64 and reached 2048. There are clear limitations: the higher the dimension, the slower the simulation. In order to partially solve this problem, we adapted the number of samples and maximum time of each simulation based on the value of N, always trying to keep an amount of data that would results in good statistic.

In practice we simulated a large range of σ for fixed values of $(\alpha, \eta, \epsilon, N)$, in order to obtain the value of σ_{crit}, i.e. the point of transition between SAT and UNSAT. The first run is usually done with short values for *max-time*, just to obtain the correct range. Once this is found, the simulations close to σ_{crit} are ran again for longer times, so that the system can reach a stationary state. This way we obtain a cleaner value for σ_{crit} and for the critical exponents.
Due to the large amount of plots, in the following sections we will report only a selection. The rest will be left in appendices with some brief comments.
We show in Fig. 4.1 how the dynamics looks like: the activity decreases and reaches a stationary state or zero, depending on the value of σ.

4.1 Probability of Satisfiability

For each value of σ we simulate many samples of the system. Doing so we can compute the probability of satifiability as a function of σ, simply counting how many samples can

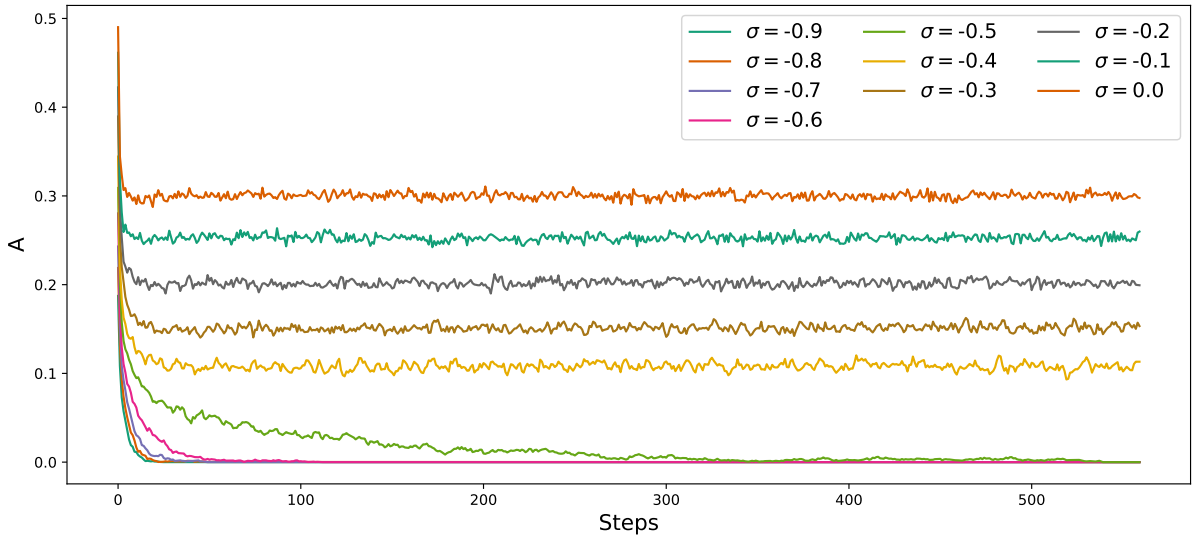


Figure 4.1: Dynamics of the system for $N = 2000$ for various value of σ

satisfy all constrains (Eq. 4.1).

$$P_{SAT} = \frac{N \text{ samples with all constrains satisfied}}{\text{total number of samples}} \tag{4.1}$$

The data obtained in such way is showed in Fig. 4.3a-Fig. 4.3c for $\eta = 1, 0.1$.
In all of these plots we can see that P_{SAT} follows a *sigmoid-like* behaviour, with the steepness increasing with N. This is because in the limit $N \to \infty$ we would obtain a perfect step function, with a jump corresponding to σ_{crit}. The effect of finite N is very clear in the curve $N = 64$, while we already obtain an almost perfect step function with $N = 512$. We also notice that the value of σ^N_{crit} decreases with N, approaching the theoretical value for infinite dimensions.
The estimates for the critical value of σ obtained by fitting P_{SAT} with a sigmoid are reported in Tab. 4.1. It is interesting to notice that for decreasing values of η, the difference between the σ_{crit} of increasing sizes gets smaller and smaller, as is clearly shown in Fig. 4.2, where $\Delta\sigma_{crit}(N_1 - N_2) = \sigma^{N_1}_{crit} - \sigma^{N_2}_{crit}$. This may be exploited to speed up the process, since we can simulate just lower dimensions for small η (faster simulations) and still obtain results very close to the infinite dimension one.

N	$\sigma_{crit}(\eta = 1)$	$\sigma_{crit}(\eta = 0.5)$	$\sigma_{crit}(\eta = 0.1)$	$\sigma_{crit}(\eta = 0.05)$
64	-0.64807	-0.42815	-0.28381	-0.26786
128	-0.68536	-0.45497	-0.28913	-0.26861
256	-0.70429	-0.46817	-0.29094	-0.26965
512	-0.71492	-0.47476	-0.29179	-0.26926
1024	-0.71776	-0.47817	-0.29289	-0.26975
2048	-0.71953	-0.47951	-	-
∞	-0.723	-0.481	-0.293	-0.270

Table 4.1: σ_{crit} obtained with the sigmoid fit of P_{SAT} for all η.

We also report the infinite dimension estimate, obtained from a linear fit of σ^N_{crit} as a function of $1/N$, shown in Fig. 4.3b and Fig. 4.3d.

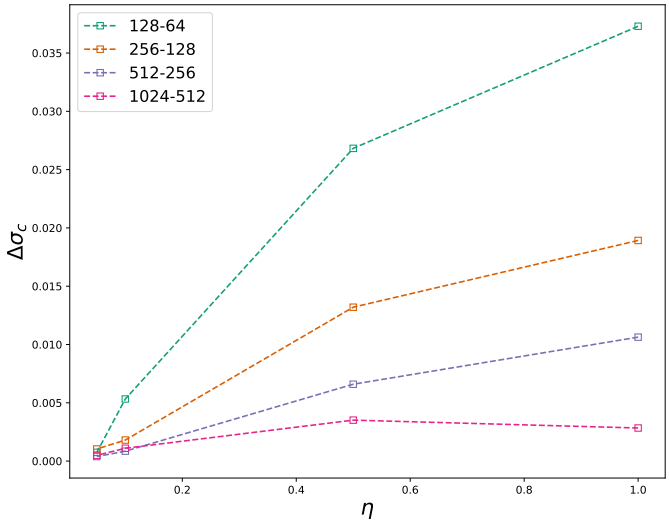


Figure 4.2: Plot of the difference between σ_{crit} for increasing N and different values of η. We can notice the convergence toward $\Delta\sigma_{crit} \approx 0$.

The points obtained for different N can be rescaled and collapsed in a single curve. This is obtained plotting them as a function of $(\sigma - \sigma^N_{crit}) \cdot N^\beta$, where β is a critical exponent that has to be found, and also corresponds to the critical exponent of the activity, which we will study in the next section. The optimal value for β is the one that achieve the best possible collapse between all curves. In Fig. 4.4, we show the curve collapse for $\eta = 0.5$ with different choices of the exponent (other examples of the collapsing of the curves are in Appendix B). We can see that $\beta = 0.4$ does not keep the curves close enough, while $\beta = 0.6$ starts to spreading too much the points closer to 1. Then we can estimate $\beta \approx 0.5$.

4.2 Activity

We will study here the observable introduced in Eq. 3.2, the activity of the system. This quantity can give us information about the number of constraints unsatisfied and whether the system has reached a stationary state or not. It is particularly important close to σ_{crit}, since there small differences can lead to very different outcomes (namely we may end up in the SAT or UNSAT region). For values of σ larger than the critical one, the activity reaches a stationary state rather fast, since the number of unsatisfied constraints will be high and not much can change in the system. Similarly for smaller values of σ, the activity will simply reach zero and stop changing after some time.
In the following, when we refer to the time evolution of the activity, what we actually mean is that each point is a mean value, averaged over $N_{samples} \times N_{threads}$. The values of these two parameters will be specified in each simulations, since they change for each N. We used the time evolution to qualitatively check whether the system reached a stationary state or not. In Fig. 4.5a, Fig. 4.5b, Fig. 4.5c, Fig. 4.5d we plot the dynamic for different values of N and η, with an inset in each graph highlighting the ending part of the evolution. We can observe that for smaller Ns the activity is less stable even at long times (notice how the fluctuations are more pronounced in Fig. 4.5a than in the other graphs), but the stationarity strongly improves already at $N = 512$. These graphs also show how η influences the dynamics: smaller values will produce a less noisy and

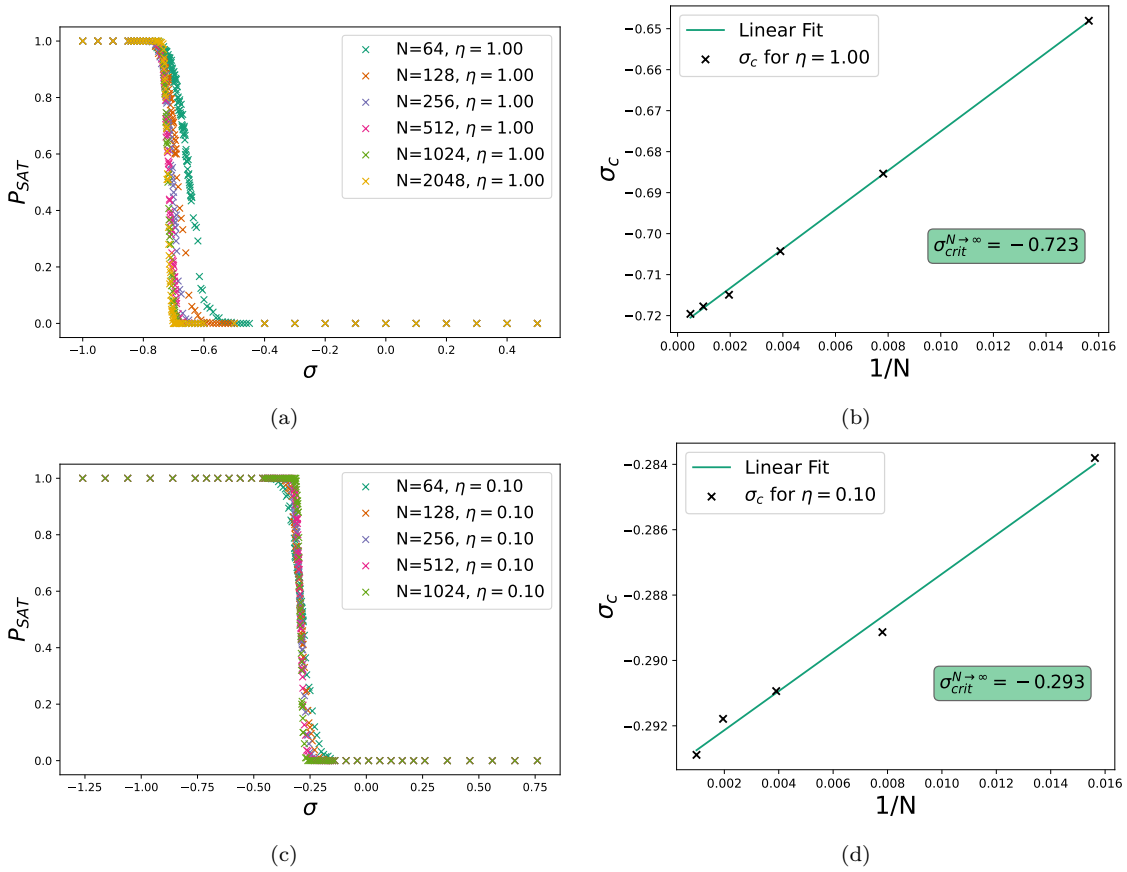


Figure 4.3: Plot of P_{SAT} vs σ for different values of N and η. In Fig. 4.3a and Fig. 4.3c we can clearly see the transition between the SAT region ($P_{SAT} = 1$) and the UNSAT region ($P_{SAT} = 0$).
In Fig. 4.3b and Fig. 4.3d we plot a linear fit of the values of σ_{crit} obtained from fitting the P_{SAT} data with a sigmoid. In the text-box we show the intercept of the line for $\frac{1}{N} = 0$ ($N \to \infty$).

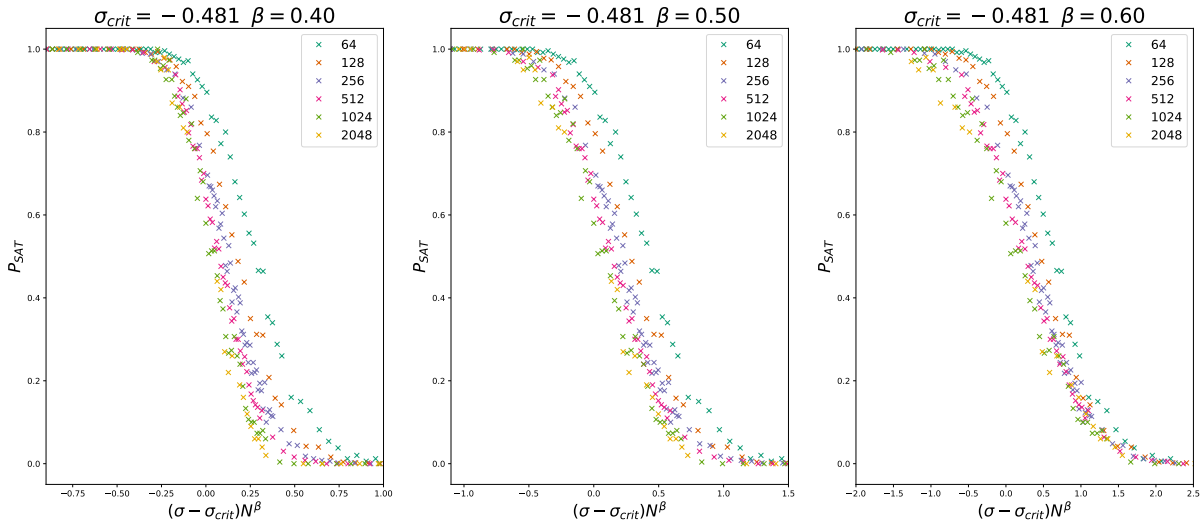
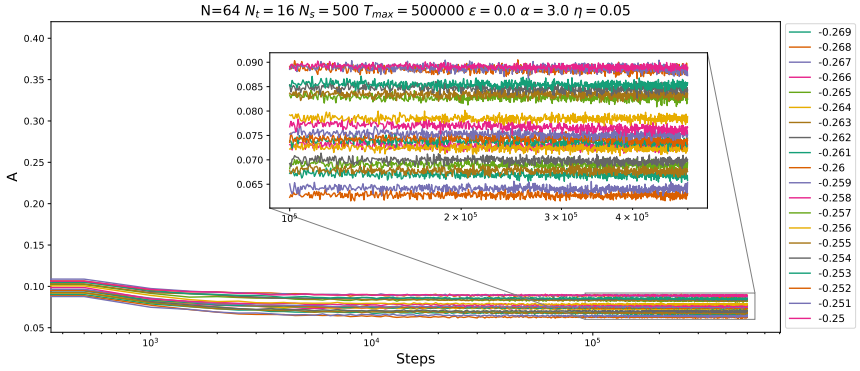
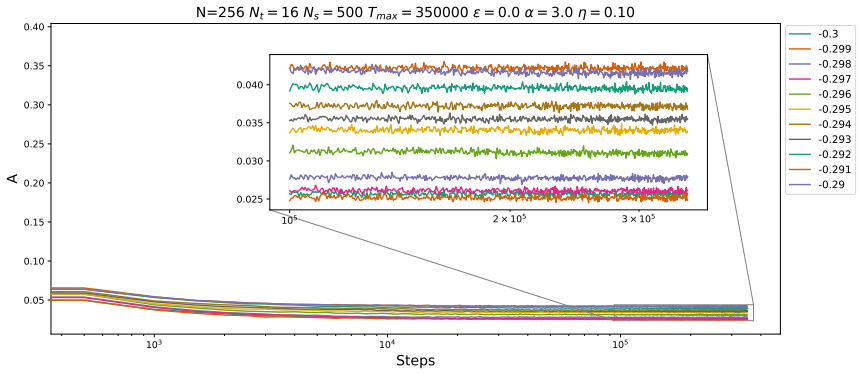


Figure 4.4: Plot of P_{SAT} for $\eta = 0.5$ using different exponents for the collapse of the curves. The best result is obtained with $\beta = 0.5$.

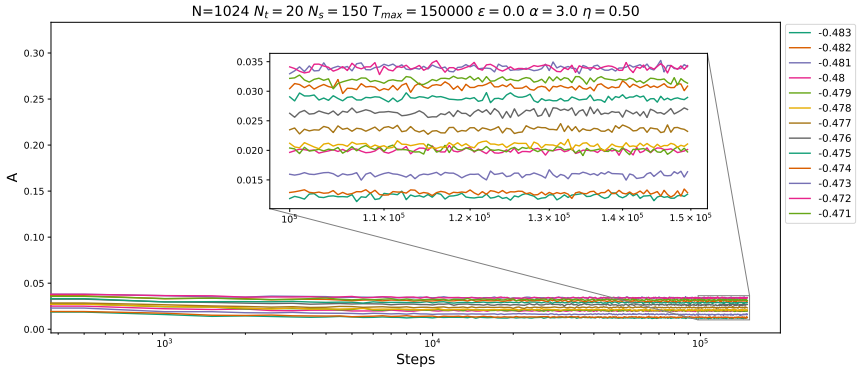
smoother evolution, which has the double effect of making the activity stationary earlier and of obtaining a value for σ_{crit} larger (the system can sustain a stronger bias), as shown in the previous images of P_{SAT}.



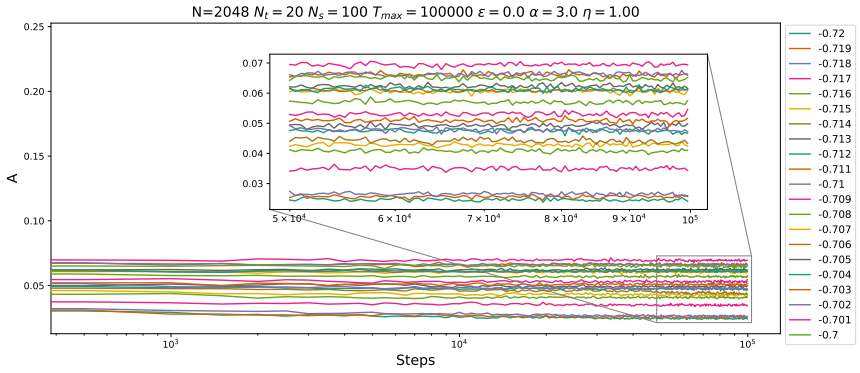
(a)



(b)



(c)



(d)

Figure 4.5: Evolution of the activity for different values of η.

For a given set of parameters we need a single value for the activity which characterizes

the system. In order to obtain such value we averaged over the time evolution, starting from a time t^* chosen arbitrarily. We also fitted these evolutions to test if the chosen t^* was a good value to obtain an average over a the stationary part of the evolution. The fitting function is of the form:

$$f(t) = A_\infty + Be^{-\frac{t}{\tau}} \tag{4.2}$$

with fitting parameters A_∞, B, τ. In Fig. 4.6a and Fig. 4.6c ($\eta = 1, 0.5$) we show the time-averaged activity as a function of σ (colored crosses) and we plotted on top the fitted values of A_∞ in black. We show in the inset of each graph an enlarged portion of the data, where we can notice that the fitted parameter and the averaged value are almost identical. Again we see that the larger differences are for smaller values of N, as a consequence of the difficulty in reaching an actual stationary state.
We are interested in characterizing the behaviour of the activity as it reaches 0 coming from the UNSAT phase. In order to do so we rescale all the points, plotting them as a function of $\sigma - \sigma_{crit}$ and in log-log scale. This way we obtain a graph that shows in a clearer way how the activity approaches 0. In Fig. 4.6b and Fig. 4.6d we show this behaviour, together with an estimate of the critical exponent for a power law fit. We can notice again the effect of the finite size, with $N = 64$, in both graphs, being the farthest from the theoretical power law. The exponent obtained from this analysis is $\beta \approx 0.5$, which is consistent with the value obtained from the collapsing of the P_{SAT} curves. This analysis still hold also for the other values of *eta*, which we report in Appendix A.

We can conclude the study of the activity with a comparison of the critical behaviour for the different values of η.

4.3 χ

We may study another quantity showing interesting behaviours at criticality: the variance of the activity, or χ, defined in Eq. 4.3. When we talk about this variance we once again refer to the averaged value over time (represented by $< \cdots >$) of mean computed for $N_{samples} \times N_{threads}$ (represented by the over-line).

$$\chi = < \overline{A_t^2} > - < \overline{A_t} >^2 \tag{4.3}$$

This is similar to a fluctuation-dissipation relation, showing how much the activity changes close to the critical σ. We then expect a curve that goes to 0 for $\sigma \ll \sigma_{crit}$ and is constant for $\sigma \gg \sigma_{crit}$, while for $\sigma \approx \sigma_{crit}$ it should diverge. The points will actually diverge only in the limit $N \to \infty$, while reaching a maximum value for finite N. In order to plot the data we need to rescale it to account for the effects of finite dimensionality, this is obtained simply multiplying the points by their respective N.
Notice that since the curves will have a maximum at σ_{crit}^N, the position of this point will depend on N itself. We estimated the position and value of the maximum taking the 5 largest points in the curve and using the mean. We can then plot the value of the maxima as a function of how far is their position from the real σ_{crit}, in order to study the approach to the theoretical divergence at $N \to \infty$. We obtain once again a power law behaviour with a deviation term, shown in log-log scale in Fig. 4.8b and Fig. 4.8d, controlled by an exponent that would seem to be dependent on η (each numerical value is reported in the corresponding graph). This result is consistent with the behaviour found in Fig. 4.2, since we saw that for smaller η the values of σ_{crit}^N of increasing N, and consequently the

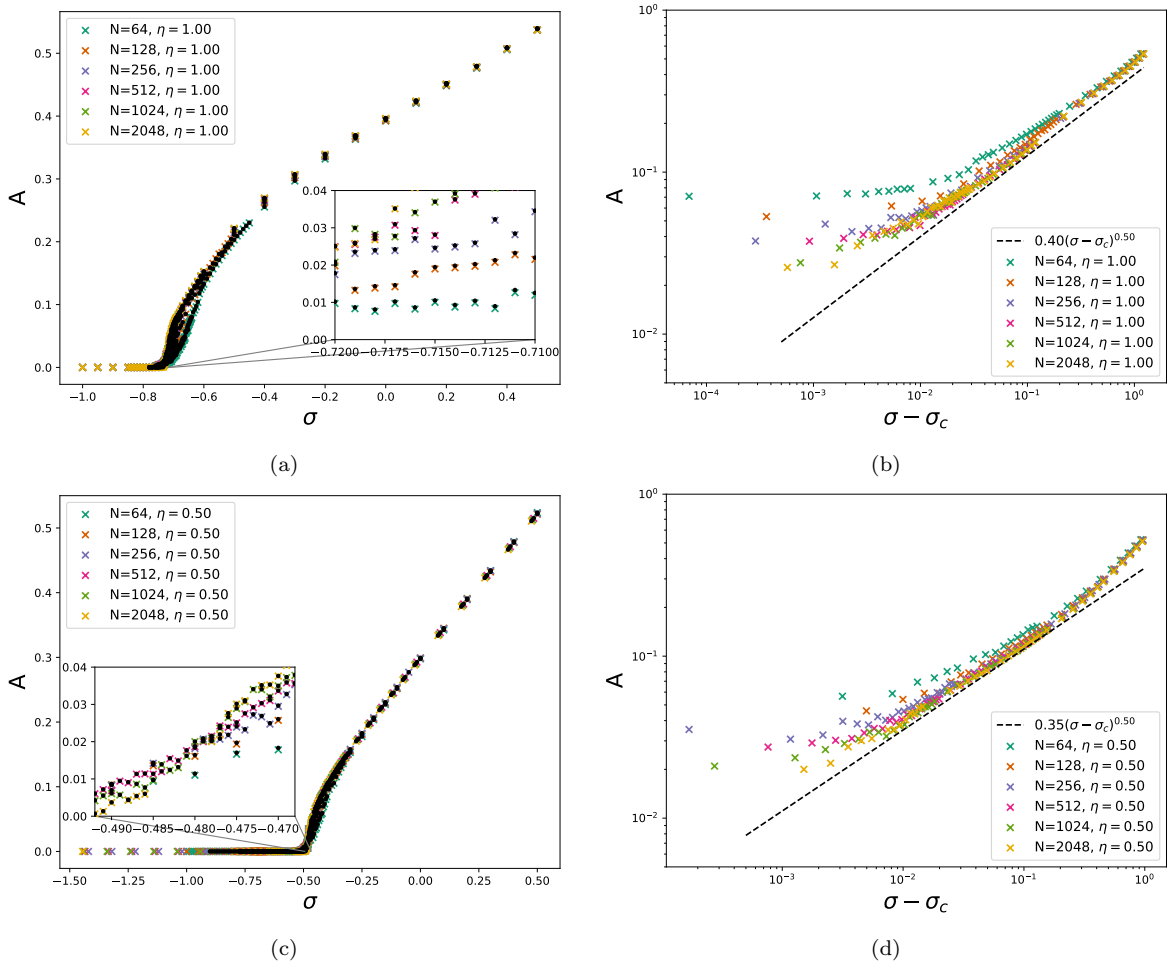


Figure 4.6: Fig. 4.6a and Fig. 4.6c shows activity vs. σ for the different N and η. The black dots represent the fitted parameter A_∞, and it is clear that the difference between the parameter and the average over the data is minimal.
In Fig. 4.6b and Fig. 4.6d we plot of the average value of the activity in the stationary regime vs. $|\sigma - \sigma_{crit}$ in a log-log scale. Each set of points has been rescaled using its own σ_{crit}, while for the black line we used the value obtained for $N \to \infty$. The critical behaviour is well captured by a power law with an exponent close to 0.50.

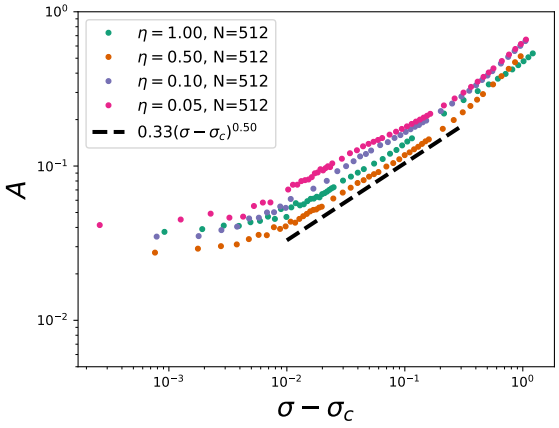


Figure 4.7: Plot of the activity at different values of η for the same N. In black we plotted a power law highlighting the behaviour close to criticality, as in the previous activity's graphs.

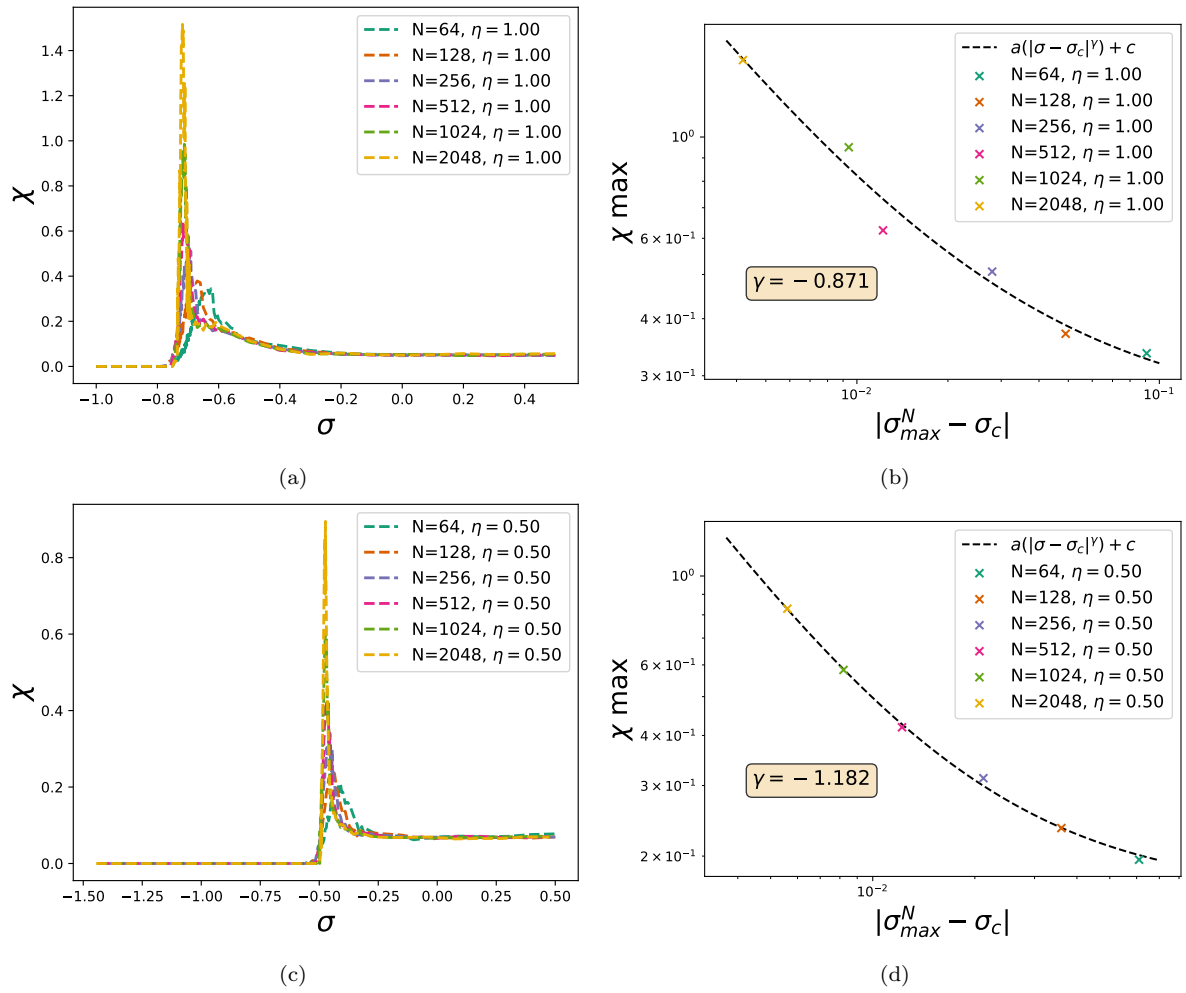


Figure 4.8: In Fig. 4.8a and Fig. 4.8c we plot the values of χ for all N and the expected behaviour is clear. In Fig. 4.8b and Fig. 4.8d we plot the value of the max as a function of its distance from σ_{crit}, showing also the power law fit and printing the exponent.

position of the maximum of χ, were closer to the theoretical value in infinite dimension. This means that the maxima will start to diverge faster as η gets smaller.

4.4 BRO with $\epsilon = 0.1$

We ran simulations for the same range of N, but with $\epsilon = 0.1$. We expect multiple effects from this change. Firstly, the time needed by the activity to reach a stationary state will increase considerably, since now at every step the vector is also moved in a random direction, that in general may not be helpful. Still, the main drive of the dynamics is the gradient descent, so we just need to wait more.
Secondly the optimization problem is now harder, meaning that the value of σ_{crit} will decrease with respect to the case of $\epsilon = 0$.
In Fig. 4.9 we show the data obtained by these simulations. We only reached $N = 512$, since, as mentioned, the convergence to a stationary state was very slow.

This affects the data needed for P_{SAT} because, for $\sigma \leq \sigma_{crit}$, the activity kept decreasing very slowly and very close to 0. This means that a longer simulation may let the activity reach 0 (the system eventually satisfy all constraints), and as a consequence we

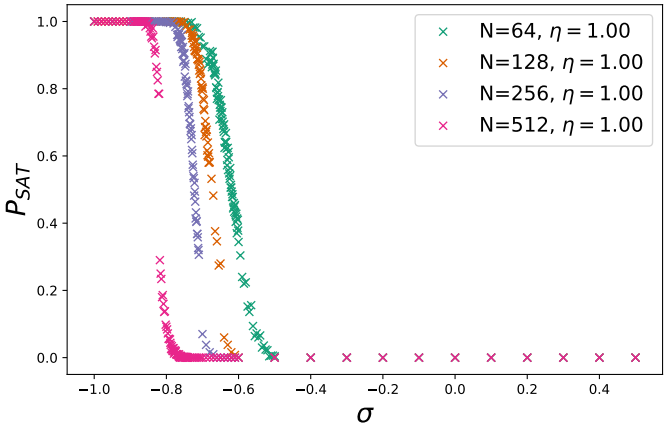


Figure 4.9: P_{SAT} for $\epsilon = 0.1$. The behaviour is as similar to the previous case, except for a larger drift related to increasing values of N.

would obtain a 1 for P_{SAT}, moving the sigmoid rightwards. This behaviour can be clearly seen in Fig. 4.10, where the curves with $\sigma \leq -0.75$ are extremely close to reaching 0 (one of them is actually 0), but they stay slightly above, oscillating and sometimes decreasing to a lower level. The problem in this is a wrong estimate of σ_{crit}, which would then make the whole study of the critical point not precise.

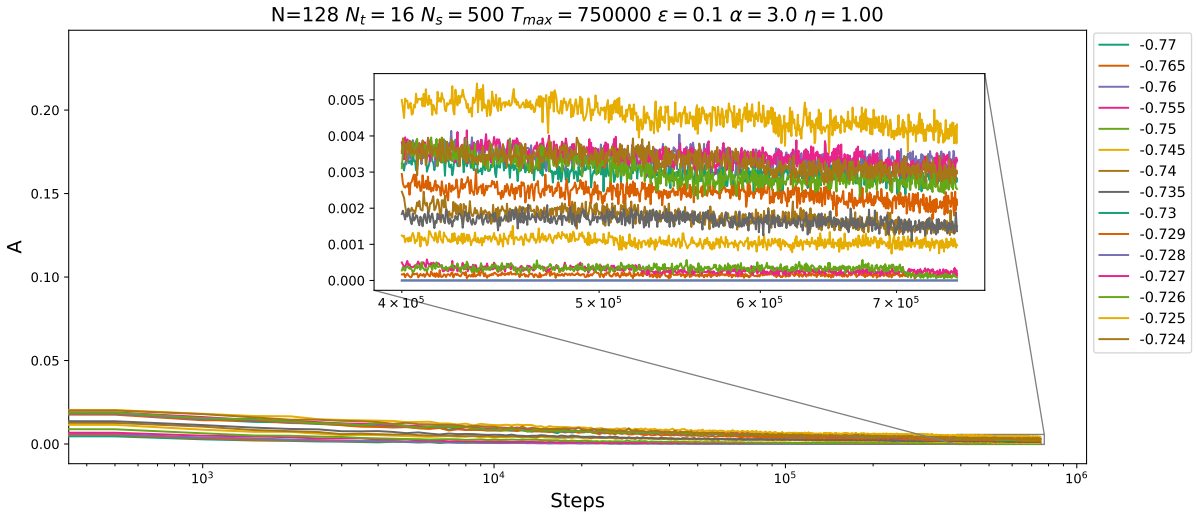


Figure 4.10: Activity dynamic for $\epsilon = 0.1$, $N = 128$, $\eta = 1$. In the inset we show the discrete jumps close to 0.

We also noticed an interesting difference between this evolution and the ones with $\epsilon = 0$. Especially when the value of the activity is very small, we can clearly see *jumps* between discrete levels. This may be a sign of the dynamics being stuck close to a minimum due to the random kick, which, if it is large enough, will continue to push the system in random directions around the minimum. It is then possible that after some time a *lucky* random move is extracted and the system actually descent closer to the minimum, in another level closer to $A(t) = 0$.
We did not explored further the behaviour, but it may be interesting carry on this study and possibly relate it to why some random learning algorithms work.

Chapter 5

Conclusions

The main purpose of this work was to study the absorbing transition of the spherical perceptron model and fully characterize it. In order to do so we run extensive simulations of the model under many different settings, and analysed the obtained data to extract critical exponents of different parameters of the system.
With the first analysis, P_{SAT}, we checked that the algorithm was producing the correct dynamics and we also extracted the critical values of our control parameter, σ, for the different sizes of the system and for different values of the hyperparameters. The results obtained from this preliminary analysis were positive, confirming that the system obeys a dynamic where there is a transition between an active, unsatisfied state and an absorbing, satisfied state.
The most important analysis we conducted is the one of the order parameter called activity. The latter fully characterize the critical transition, and we were interested in finding the exponent controlling how it reaches zero at the critical point. The results were particularly interesting, since most of the models showing a dynamic with absorbing states belong to the so called Manna universality class, while ours seems to be in a different class. Indeed the expected exponent β for the activity of a mean field model belonging to the Manna class is 1 ([21]), while we obtained $\beta \sim 0.5$[1]. This shows that our model can not be considered part of the same class, and hence further studies have to be conducted.
These results were obtained from the fully deterministic algorithm, since is the one we focused on. Preliminary simulations ran on the partially random algorithm have shown that the dynamic is quite different from the deterministic case, and in particular the time needed to obtain a stationary state that can yield good results is extremely long. A deeper study of the random model may also shed light on the reason why random algorithms still work well when training machine learning models.
We conducted further analysis considering the fluctuations between different samples of the same simulation. With this analysis we observed that, near criticality, small changes in the control parameter produce large changes in the order parameter, in particular we observed a maximum of these fluctuation exactly at the critical point.
It will also be important in future works to develop, alongside these simulations, a theoretical framework, to describe exactly what we observed here.
In conclusion, this work can be considered a step in the study of this model and in the development of a more sound and clear theory that ranges from neural networks to glassy

[1]However we underline again that our definition of the activity is slightly different than in particles systems, since we count the amount of negative overlaps. However close to the transition this quantity is proportional to the number of overlapping particles.

systems. We have shown throughout this work how the tools and techniques used to study the latter can be applied to the characterization of the former. Following on this line, in the future it may become possible to fully understand many of the underlying mechanisms of modern learning techniques, and to be able to then exploit this knowledge for the creation of better and more explainable algorithms.

Bibliography

[1] Lenka Zdeborová and Florent Krzakala. "Statistical physics of inference: thresholds and algorithms". In: *Advances in Physics* 65.5 (2016), pp. 453–552. DOI: 10.1080/00018732.2016.1211393. eprint: https://doi.org/10.1080/00018732.2016.1211393. URL: https://doi.org/10.1080/00018732.2016.1211393.

[2] Sebastian Goldt et al. "Modeling the Influence of Data Structure on Learning in Neural Networks: The Hidden Manifold Model". In: *Phys. Rev. X* 10 (4 Dec. 2020), p. 041044. DOI: 10.1103/PhysRevX.10.041044. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.041044.

[3] Sam Wilken et al. "Hyperuniform Structures Formed by Shearing Colloidal Suspensions". In: *Phys. Rev. Lett.* 125 (14 Sept. 2020), p. 148001. DOI: 10.1103/PhysRevLett.125.148001. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.148001.

[4] Francesca Mignacco, Pierfrancesco Urbani, and Lenka Zdeborová. "Stochasticity helps to navigate rough landscapes: comparing gradient-descent-based algorithms in the phase retrieval problem". In: *Machine Learning: Science and Technology* 2.3 (July 2021), p. 035029. DOI: 10.1088/2632-2153/ac0615. URL: https://doi.org/10.1088/2632-2153/ac0615.

[5] Francesca Mignacco and Pierfrancesco Urbani. *The effective noise of Stochastic Gradient Descent*. 2021. DOI: 10.48550/ARXIV.2112.10852. URL: https://arxiv.org/abs/2112.10852.

[6] Stéphane d'Ascoli, Maria Refinetti, and Giulio Biroli. *Optimal learning rate schedules in high-dimensional non-convex optimization problems*. 2022. DOI: 10.48550/ARXIV.2202.04509. URL: https://arxiv.org/abs/2202.04509.

[7] Sebastian Goldt et al. "The Gaussian equivalence of generative models for learning with shallow neural networks". In: *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*. Ed. by Joan Bruna, Jan Hesthaven, and Lenka Zdeborova. Vol. 145. Proceedings of Machine Learning Research. PMLR, 16–19 Aug 2022, pp. 426–471. URL: https://proceedings.mlr.press/v145/goldt22a.html.

[8] Federica Gerace et al. *Gaussian Universality of Linear Classifiers with Random Labels in High-Dimension*. 2022. DOI: 10.48550/ARXIV.2205.13303. URL: https://arxiv.org/abs/2205.13303.

[9] Silvio Franz, Antonio Sclocchi, and Pierfrancesco Urbani. "Critical Jammed Phase of the Linear Perceptron". In: *Phys. Rev. Lett.* 123 (11 Sept. 2019), p. 115702. DOI: 10.1103/PhysRevLett.123.115702. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.115702.

[10] Silvio Franz, Antonio Sclocchi, and Pierfrancesco Urbani. "Critical energy landscape of linear soft spheres". In: *SciPost Physics* 9.1 (July 2020). DOI: 10.21468/scipostphys.9.1.012. URL: https://doi.org/10.21468%2Fscipostphys.9.1.012.

[11] Silvio Franz, Antonio Sclocchi, and Pierfrancesco Urbani. "Surfing on minima of isostatic landscapes: avalanches and unjamming transition". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.2 (Feb. 2021), p. 023208. DOI: 10.1088/1742-5468/abdc16. URL: https://doi.org/10.1088%2F1742-5468%2Fabdc16.

[12] Corey S. O'Hern et al. "Jamming at zero temperature and zero applied stress: The epitome of disorder". In: *Phys. Rev. E* 68 (1 July 2003), p. 011306. DOI: 10.1103/PhysRevE.68.011306. URL: https://link.aps.org/doi/10.1103/PhysRevE.68.011306.

[13] Sam Wilken et al. "Random Close Packing as a Dynamical Phase Transition". In: *Phys. Rev. Lett.* 127 (3 July 2021), p. 038002. DOI: 10.1103/PhysRevLett.127.038002. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.038002.

[14] Christopher Ness and Michael E. Cates. "Absorbing-State Transitions in Granular Materials Close to Jamming". In: *Phys. Rev. Lett.* 124 (8 Feb. 2020), p. 088004. DOI: 10.1103/PhysRevLett.124.088004. URL: https://link.aps.org/doi/10.1103/PhysRevLett.124.088004.

[15] Laurent Corté et al. "Random organization in periodically driven systems". In: *Nature Physics* 4.5 (May 2008), pp. 420–424. ISSN: 1745-2481. DOI: 10.1038/nphys891. URL: https://doi.org/10.1038/nphys891.

[16] F. Rosenblatt. *The perceptron - A perceiving and recognizing automaton.* Tech. rep. 85-460-1. Ithaca, New York: Cornell Aeronautical Laboratory, Jan. 1957.

[17] Silvio Franz et al. "Universality of the SAT-UNSAT (jamming) threshold in non-convex continuous constraint satisfaction problems". In: *SciPost Phys.* 2 (3 2017), p. 019. DOI: 10.21468/SciPostPhys.2.3.019. URL: https://scipost.org/10.21468/SciPostPhys.2.3.019.

[18] Giorgio Parisi, Pierfrancesco Urbani, and Francesco Zamponi. *Theory of Simple Glasses: Exact Solutions in Infinite Dimensions.* Cambridge University Press, 2020. DOI: 10.1017/9781108120494.

[19] Silvio Franz and Giorgio Parisi. "The simplest model of jamming". In: *Journal of Physics A: Mathematical and Theoretical* 49.14 (Feb. 2016), p. 145001. DOI: 10.1088/1751-8113/49/14/145001. URL: https://doi.org/10.1088/1751-8113/49/14/145001.

[20] M. Galassi et al. *GNU Scientific Library Reference Manual.* URL: http://www.gnu.org/software/gsl/.

[21] Malte Henkel, Haye Hinrichsen, and Sven Lübeck. "Non-Equilibrium Phase Transitions". In: (2016). DOI: https://doi.org/10.1007/978-1-4020-8765-3. URL: https://link.springer.com/book/10.1007/978-1-4020-8765-3.

[22] Ronald Dickman, Alessandro Vespignani, and Stefano Zapperi. "Self-organized criticality as an absorbing-state phase transition". In: *Phys. Rev. E* 57 (5 May 1998), pp. 5095–5105. DOI: 10.1103/PhysRevE.57.5095. URL: https://link.aps.org/doi/10.1103/PhysRevE.57.5095.

[23] Jef Hooyberghs, Ferenc Iglói, and Carlo Vanderzande. "Strong Disorder Fixed Point in Absorbing-State Phase Transitions". In: *Phys. Rev. Lett.* 90 (10 Mar. 2003), p. 100601. DOI: 10.1103/PhysRevLett.90.100601. URL: https://link.aps.org/doi/10.1103/PhysRevLett.90.100601.

[24] Haye Hinrichsen. "Non-equilibrium critical phenomena and phase transitions into absorbing states". In: *Advances in Physics* 49.7 (Nov. 2000), pp. 815–958. DOI: 10.1080/00018730050198152. URL: https://doi.org/10.1080%2F00018730050198152.

[25] Lars Milz and Michael Schmiedeberg. "Connecting the random organization transition and jamming within a unifying model system". In: *Phys. Rev. E* 88 (6 Dec. 2013), p. 062308. DOI: 10.1103/PhysRevE.88.062308. URL: https://link.aps.org/doi/10.1103/PhysRevE.88.062308.

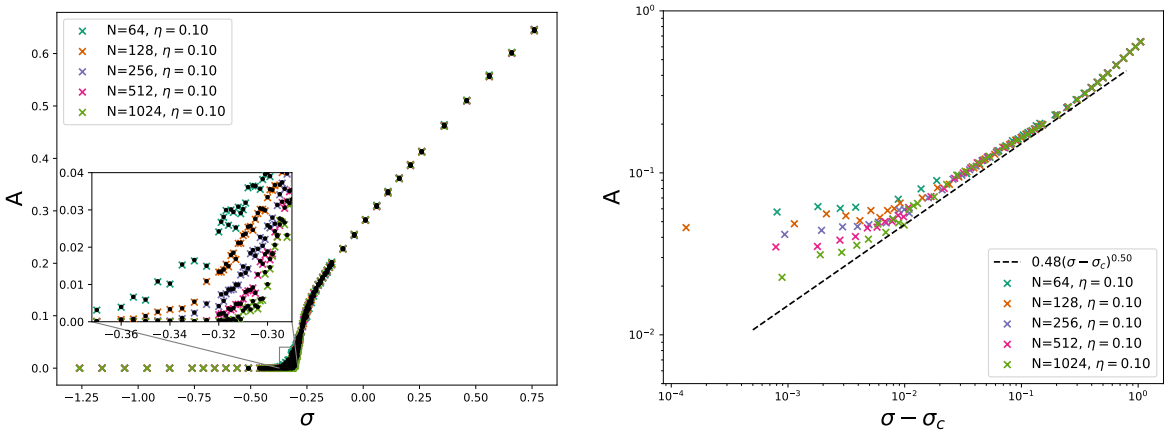
[26] Silvio Franz et al. "Universal spectrum of normal modes in low-temperature glasses". In: *Proceedings of the National Academy of Sciences* 112.47 (2015), pp. 14539–14544. DOI: 10.1073/pnas.1511134112. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1511134112. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1511134112.

[27] Per Bak, Chao Tang, and Kurt Wiesenfeld. "Self-organized criticality: An explanation of the 1/f noise". In: *Phys. Rev. Lett.* 59 (4 July 1987), pp. 381–384. DOI: 10.1103/PhysRevLett.59.381. URL: https://link.aps.org/doi/10.1103/PhysRevLett.59.381.

Appendix A

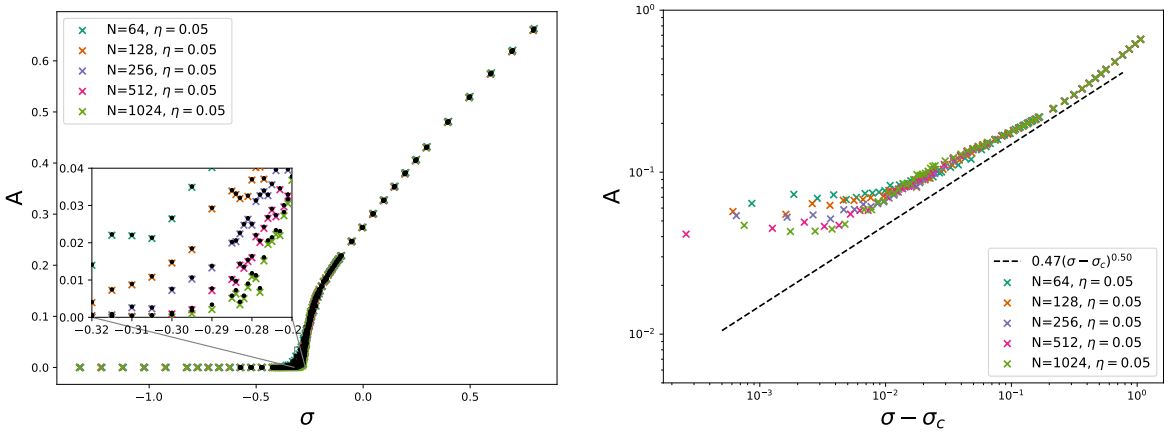
Activity Plots

We report here the other plots for the activity, for $\eta = 0.1, 0.05$. The results are still consistent with our analysis. Notice that for $N = 1024, \eta = 0.05$ some new behaviour seems to appear. It may be interesting to study more this case in the future.



(a) (b)

Figure A.1: Activity for $\eta = 0.1$



(a) (b)

Figure A.2: Activity for $\eta = 0.05$

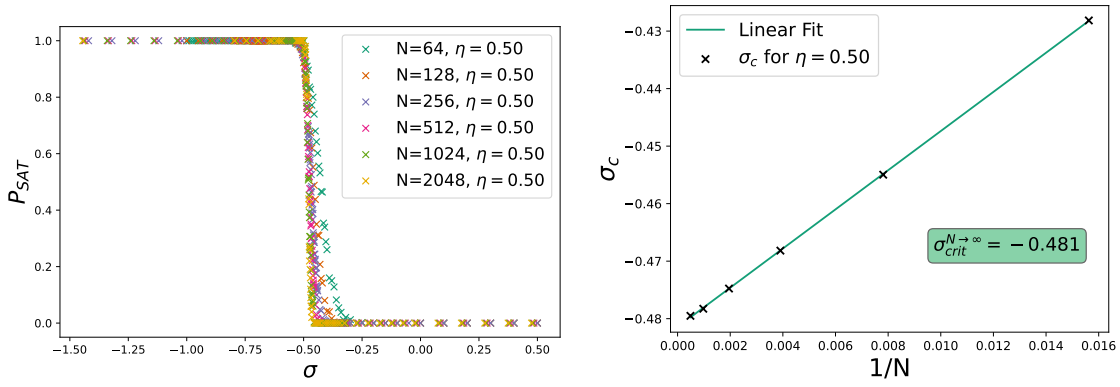
Appendix B

P_{SAT}

We report here the other plots of P_{SAT}.

B.1 P_{SAT} for $\eta = 0.5$

The results shown here are perfectly in line with the ones discussed in the main part of this work.

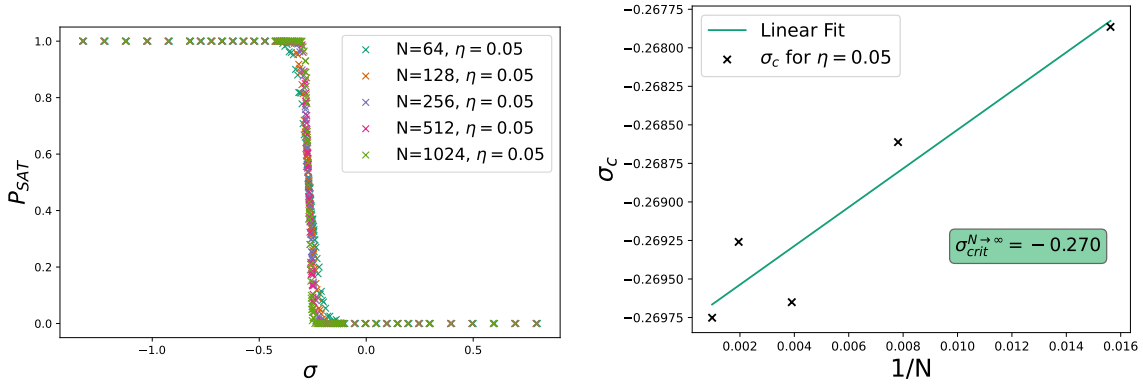


(a) (b)

Figure B.1: P_{SAT} and the linear fit of σ_{crit} for $\eta = 0.5$

B.2 P_{SAT} for $\eta = 0.05$

It is interesting to notice here what we discussed in Fig. 4.2: for $\eta = 0.05$ the difference between σ_{crit} for increasing N are very small, so much that fluctuations become quite important. This is particularly visible in Fig. B.2b: the last three points ($N = 256, 512, 1024$) just oscillates around the fit.



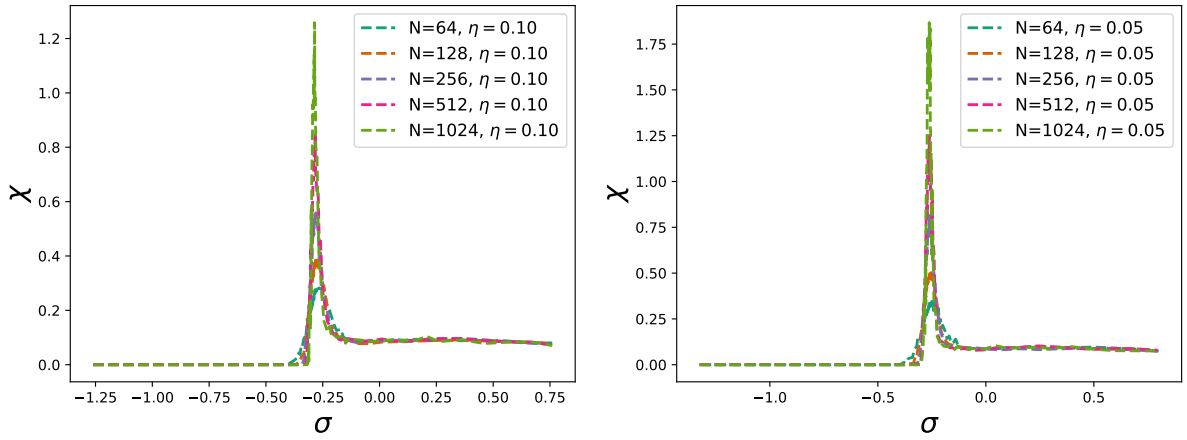
(a) (b)

Figure B.2: P_{SAT} and the linear fit of σ_{crit} for $\eta = 0.05$

Appendix C

χ Plots

Here we present the plot of χ for $\eta = 0.1, 0.05$. The position of the maxima is almost exactly the same now, and looking at the plots of the previous section (Fig. B.2b) it even happened that the maximum value of $N = 512, \eta = 0.05$ appears at a value of σ which is larger than the one of $N = 256$.



(a) (b)

Figure C.1: χ for $\eta = 0.1$ and $\eta = 0.05$.