

MAXIMAL AVALANCHES IN THE BAK–SNEPPEN MODEL

ALEXIS GILLET^{*, **} AND

RONALD MEESTER,^{*, ***} *Vrije Universiteit Amsterdam*

PETER VAN DER WAL,^{****} *EURANDOM*

Abstract

We study the durations of the avalanches in the maximal avalanche decomposition of the Bak–Sneppen evolution model. We show that all the avalanches in this maximal decomposition have infinite expectation, but only ‘barely’, in the sense that if we made the appropriate threshold a tiny amount smaller (in a certain sense), then the avalanches would have finite expectation. The first of these results is somewhat surprising, since simulations suggest finite expectations.

Keywords: Bak–Sneppen evolution model; avalanche; self-organised criticality

2000 Mathematics Subject Classification: Primary 60K35; 82B20; 82C22; 60J99

1. Introduction and main results

The Bak–Sneppen model was originally introduced as a simple model of evolution by Per Bak and Kim Sneppen [3]. Their model can be defined as follows. There are N species arranged on a circle, with a random *fitness*, independent and uniformly distributed on $(0, 1)$, assigned to each species. At each discrete time step the system is updated by locating the lowest fitness and replacing this fitness and those of its two neighbours by independent uniform $(0, 1)$ random variables. We think of the N species as the vertices of a circular graph, Λ_N , with N vertices.

This model can be defined on any finite connected graph, with the neighbours of each vertex being determined by the structure of the graph. For us, only the number of vertices of such a graph will be important (and the fact that it is connected), and we define \mathcal{G}_N as the collection of connected graphs with N vertices.

One of the ways to analyse the Bak–Sneppen model is to break it down into a series of *avalanches*. An avalanche at a *threshold* b (also called a *b-avalanche*) is said to occur between times s and $s + t$ if at time s all the fitnesses are equal to or greater than b , with equality at at most one vertex, and time $s + t$ is the next time after s at which this occurs. In the literature a number of different types of avalanche have been proposed. Our definition is consistent with a number of other papers [3], [9], [10], [12], although b is not always used to denote the threshold in precisely the same way; other types of avalanche have for instance been defined in [7] and [8]. Note that if we have a minimum fitness value of b , then we can choose any value up to (and including) b to be our avalanche threshold. Furthermore, it is the threshold (and not the exact

Received 14 June 2005; revision received 31 May 2006.

* Postal address: Department of Mathematics, Vrije Universiteit Amsterdam, De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands.

** Email address: ajg@cs.vu.nl

*** Email address: rmeester@cs.vu.nl

**** Postal address: EURANDOM, PO Box 513, 5600 MB Eindhoven, The Netherlands.

Email address: vanderwal@eurandom.tue.nl

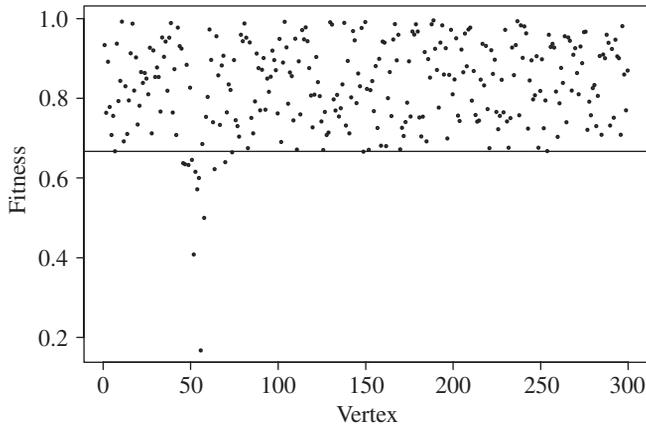


FIGURE 1: A snapshot of the fitnesses in the Bak–Sneppen model in stationarity.

initial values of the model) that determines the behaviour of an avalanche. Once we have used the initial fitnesses to find out the minimal fitness and its location, all other information can be discarded for the purposes of analysing individual avalanches.

One of the main reasons for interest in the Bak–Sneppen model is that it gives an example of *self-organised criticality*. Spatial probabilistic models (such as percolation) typically have a parameter that needs to be set before running the model, for example the probability of an edge being open in bond percolation. Often this parameter can be set to a particular value, the critical value, where unusual behaviour is observed. Models at criticality tend to exhibit scale-free or fractal behaviour, that is, the model ‘looks the same’ when viewed at different levels of magnification. Their behaviour often obeys power laws. However, this behaviour is very unstable, as the parameter has to be set exactly to the critical value.

Many observable phenomena seem to obey power laws; the relationship between the frequency of earthquakes and their sizes is perhaps the most commonly cited example. However, models that require a finely tuned parameter lack the robustness to explain such phenomena. This led Bak (among others) to consider models that tune themselves automatically to this critical state, coining the phrase ‘self-organised criticality’ [4]. Interested readers should also see Bak’s book on the subject [1].

In the original Bak–Sneppen model (on the circle), the notion of an avalanche helps to explain the criticality of the model. For large N , after a while the dynamics appear to consist of consecutive avalanches at a threshold, b_c , close to $\frac{2}{3}$. In addition, these avalanches seem to exhibit power law behaviour in the sense that both duration and range can be described by power laws [3]. The threshold b_c is not set beforehand; the model seems to organise *itself* into this state. See Figure 1 for a typical snapshot of the Bak–Sneppen model in stationarity, with $N = 300$. On the horizontal axis we have the 300 vertices, with the dots representing the fitnesses of the vertices.

The Bak–Sneppen model can be thought of as a sequence of consecutive avalanches. Since different avalanche thresholds can be chosen, there are a number of ways to perform such an *avalanche decomposition*. One common approach is to consider the model as a series of avalanches at some *fixed* threshold level b ; see, for instance, [10].

There has also been considerable attention given to the *maximal avalanche decomposition*. Here the first avalanche threshold is defined to be the minimum initial fitness value. After this

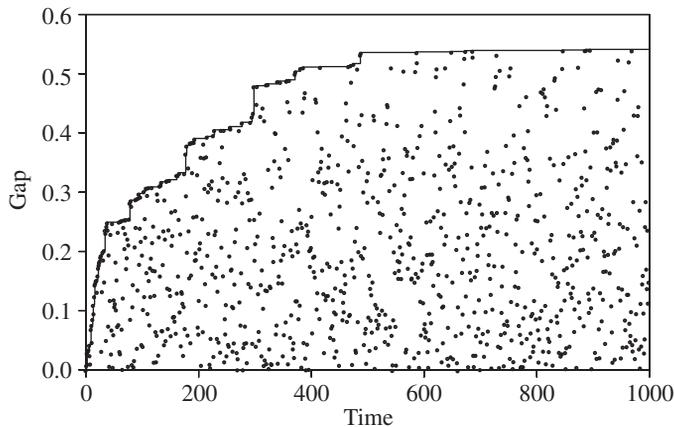


FIGURE 2: A realisation of the gap function for $N = 100$.

and every subsequent avalanche, another avalanche begins with the threshold chosen to be the new minimal value of the model; this is the maximal threshold choice. It is clear that this leads to a picture of the Bak–Sneppen model as a series of avalanches at strictly increasing thresholds. The *gap function* at time s , $G(s)$, is defined to be the avalanche threshold at time s [2]. The gap function is a stepwise-increasing function which jumps to a new value each time an avalanche ends. Note that, for all finite systems, the gap function tends to 1 almost surely. Figure 2 shows a realisation of the gap function, represented by the line, with the dots being the minimum fitness values at each time step. The initial fitnesses were independent and uniform(0, 1)-distributed.

In this report we investigate the expected durations of avalanches in the maximal avalanche decomposition. One reason for looking at the maximal avalanche decomposition is to gain insight into how the Bak–Sneppen model tends towards criticality. An alternative approach to the same question is via the *gap equation*. The gap equation is a conjectured differential equation that describes the behaviour of the gap function. It has been commonly referred to in the physics literature [12], [13] and is only informally defined. It refers to a different form of the gap function and is not studied in this paper.

Previous mathematical literature on the model concentrated on the expected duration of an avalanche at a fixed, nonrandom threshold b [11]. The results of [11] include a number of useful monotonicity results, as well as an explicit differential equation relating the expected duration of avalanches to their expected range. In this paper we study the avalanches at random thresholds which either appear in the maximal avalanche decomposition or are strongly related to the thresholds therein.

On Λ_N (or any transitive graph), the threshold is the only variable needed to determine the distribution of an avalanche’s duration. By this we mean that the durations of two avalanches on a transitive graph are identically distributed if their thresholds are the same. On a nontransitive graph, the origin of the avalanche (the vertex which has the minimal initial fitness) also plays a role. However, we can still discuss the distribution of the initial avalanche with a random origin.

Consider the Bak–Sneppen model on $G_N \in \mathcal{G}_N$. Concentrating first on the *initial* avalanche in the maximal decomposition, we see that the initial threshold is the minimum of N independent uniform(0, 1) random variables. To be more explicit, we have an avalanche with random

threshold B whose density, $h_N(b)$, is given by

$$h_N(b) = N(1 - b)^{N-1}, \quad 0 \leq b \leq 1.$$

Letting $D(G_N)$ denote the random duration of the initial avalanche on G_N , we have the following theorem.

Theorem 1.1. *The expected duration of the first avalanche on G_N is infinite, i.e. $E(D(G_N)) = \infty$, for any $G_N \in \mathcal{G}_N$.*

One consequence of this result (for transitive graphs) is that any subsequent avalanche also has infinite expected duration, as its threshold is stochastically larger than that of the first avalanche. Hence, the gap function consists of a sequence of steps, each of which has infinite expected length.

The usual way to analyse the Bak–Sneppen model has been to run computer simulations. Compared with these simulations, our result seems somewhat surprising, since divergent behaviour is not typically noticeable in numerical simulations of the model, especially when N is large. This is because the long avalanches that are responsible for this behaviour occur when the (random) threshold B is high, which is exponentially unlikely in N . If we were to run computer simulations of the initial avalanche in order to estimate its expected duration, it would still be possible to detect this, but only from the dramatic variability of these estimations (even when a very large number of simulations are used). Theorem 1.1 is, therefore, an example of the value of analytic methods, as only very careful interpretation of computer simulations would lead one to suspect this result.

We decided to perturb the avalanche threshold by making it stochastically smaller and see whether this would lead to convergence. It turns out that $E(D(G_N))$ is ‘barely infinite’, in that making the threshold stochastically smaller by a tiny amount (where this reduction tends to 0 as N tends to ∞) yields finite expected durations. To be precise, we denote by $D_n(G_N)$ the expected duration of an avalanche at a threshold which is set by the minimum of n uniform(0, 1) random variables on G_N . In this notation, the previously used $E(D(G_N))$ can now be written as $D_N(G_N)$, with Theorem 1.1 now stating that $D_N(G_N) = \infty$. We have the following result.

Theorem 1.2. *For all $G_N \in \mathcal{G}_N$, an avalanche at a threshold chosen as the minimum of $n > N$ independent uniform(0, 1) random variables has finite expectation, i.e. $D_n(G_N) < \infty$ for all $n > N$.*

Just adding one uniform random variable when setting the threshold is thus enough to obtain a finite expected duration.

However, it is possible to show that, under certain conditions, all *further* avalanches have infinite expected duration. Recall that, on G_N , setting the threshold as the minimum of N independent uniform(0, 1) random variables gives infinite expected duration. If all the fitnesses (except the minimum) are independent and uniformly distributed above the threshold at the start of the avalanche, then at the end of the avalanche all the vertices will again be independent and uniformly distributed above the threshold. So even if b is fixed and the fitnesses are chosen to be uniform and greater than b , it follows from Theorem 1.1 that the next avalanche will have infinite expected duration.

A more general, but weaker, form of this result applies when we drop the condition that the fitnesses be uniformly distributed at the start of the avalanche. All the vertices updated by the avalanche will be independent and uniformly distributed above the threshold at the end of the avalanche. So once we have had a spanning avalanche (one that updates every vertex in

the system during its duration), all subsequent avalanches (from maximal thresholds) will have infinite expected duration, no matter what initial fitness values are chosen.

2. Proof of Theorem 1.1

We prove that the initial maximal avalanche has infinite expected duration. This is done by first proving the result for the special case in which $G_N = \Lambda_N$ and then explaining how to modify the proof to generalise the result to all graphs $G_N \in \mathcal{G}_N$.

It is essentially a trivial result that $D_3(\Lambda_3) = \infty$. Indeed, at each time step the fitnesses are independently and identically distributed, so this is just a record values result [5, ‘The persistence of bad luck’, p. 15]. Alternatively, an explicit expression for the duration of an avalanche at a fixed threshold can be found and then integrated out over the distribution of B , the random avalanche threshold. For $N > 3$, a more subtle approach is required since we have neither independence nor identically distributed minimal fitnesses. In the proof for general N , we use the observation that if, at any time, all the fitnesses are below the (random) threshold B , then the avalanche cannot stop before at least N update values greater than B have been observed. We calculate the expected time taken for N such updates to be observed between times when all fitnesses are below the threshold.

Before we start on the main body of the proof we need some definitions. Let $f_i(n)$ denote the fitness of vertex i at time step n , where the initial fitnesses, corresponding to $n = 0$, are independent and uniform on $(0, 1)$. We generate our updates from a sequence of independent uniform $(0, 1)$ random variables, $U_k, k = 0, 1, 2, \dots$, as follows: if vertex i has the minimal fitness at time n , then the fitnesses at positions $(i - 1, i, i + 1)$ are replaced by $(U_{3n}, U_{3n+1}, U_{3n+2})$, giving the fitnesses at time $n + 1$.

We divide the avalanche into time *blocks* of length $2N$, such that block m contains the time steps starting with $2mN$ and ending with $2(m + 1)N - 1$, for $m = 0, 1, 2, \dots$. We say that a block is *bad* if all the updates in the block are below the threshold B and are in decreasing order. Hence, block m is bad if

$$B > U_{6mN} > U_{6mN+1} > U_{6mN+2} > \dots > U_{6(m+1)N-1}.$$

We let s_r denote the (random) time just after the r th bad block, for $r = 1, 2, \dots$. The next lemma tells us that after a bad block all fitnesses are below B .

Lemma 2.1. *For all $i, 0 \leq i \leq N - 1$, and $r = 1, 2, \dots$, we have $f_i(s_r) < B$.*

Proof. Consider the state of the system at the beginning of the r th bad block. We update the vertex with the minimal fitness together with its two neighbours. Since the block is bad, the three updated vertices now have fitnesses below the threshold. We now have two possible cases: either one of the new fitnesses is the minimum, or another vertex has the minimal fitness. If the minimum is drawn from a newly updated vertex then the avalanche will continue (for the duration of the bad block) around the circle of vertices, since after another update the minimum among the three updated vertices will be lower than the minimum at the previous step and, so, will in fact be the global minimum. The second case cannot occur more than N times before the first case occurs. Once the minimum is among the new fitnesses, it cannot take more than N vertices for the first case to cover the entire vertex set. Hence, after $2N$ updates all the vertices will have fitnesses generated by new updates. Thus, all fitnesses will be below the threshold, as required.

We call a period between the endpoints of two bad blocks an *epoch*; hence, the m th epoch corresponds to the time interval $[s_m, s_{m+1} - 1]$, $m = 1, 2, \dots$. We say that an epoch is *good* if, among the updating random variables corresponding to that epoch, there are at least N updating random variables greater than B . So, the m th epoch is good if

$$\sum_{k=3s_m}^{3s_{m+1}-1} \mathbf{1}_{\{U_k \geq B\}} \geq N,$$

where $\mathbf{1}$ denotes the indicator function. We let S be the index of the first good epoch. Now observe that, given that block 0 is bad, the first avalanche can only end during a good epoch. This indicates that the expectation of S plays an important role in our analysis.

Lemma 2.2. $E(S) = \infty$.

Proof. The key to the proof is to compute the expectation of S given that $B = b$. Indeed, conditioned on $B = b$, S has a geometric distribution, with a successful trial being a good epoch. This is because, conditional on the threshold, the lengths of the epochs are independent and identically distributed. We let $p(b)$ be the probability that the first epoch (or any other, for that matter) is good, given that $B = b$. Integrating over the range of b , we then have

$$E(S) = \int_0^1 \frac{N(1 - b)^{N-1}}{p(b)} db,$$

since B is distributed as the minimum of N independent uniform(0, 1) random variables. It remains to estimate $p(b)$. Since we want to bound the integral from below, we have to bound $p(b)$ from above. We let F_n be the event that the first epoch does not end before or at time $s_1 + n$, and let G_n be the event that

$$\min \left\{ \ell : \sum_{k=3s_1}^{\ell} \mathbf{1}_{\{U_k \geq B\}} \geq N \right\} = n,$$

that is, the event that the n th update random variable since the start of the first epoch – naturally ordered $U_{3s_1}, U_{3s_1+1}, \dots$ – is the N th above b . We can now write

$$p(b) = \sum_{n=N}^{\infty} P(G_n \mid B = b) P(F_{\lfloor n/3 \rfloor} \mid G_n, B = b),$$

where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x . Now observe that

$$P(G_n \mid B = b) = \binom{n-1}{N-1} (1-b)^N b^{n-N},$$

which we bound from above by $\binom{n-1}{N-1} (1-b)^N$. Furthermore, conditioning on $B = b$ and G_n means that we randomly select $N - 1$ updates among the first $n - 1$ updates after s_1 , which are declared to be the updates above b ; the only information about all the other updates is that they are below b . No matter the precise realisation, this implies that (for $n > 6N^2$) there are at least $\lfloor (n - 1)/(6N) \rfloor - (N - 1)$ blocks so far for which our only information is that their updates are all below b . Hence, the conditional distribution of the updates in these blocks is

i.i.d. uniform on $(0, b)$ and any of these blocks is bad with a (conditional) probability which is uniformly bounded below by a constant, $c_1(N)$ say. This then gives

$$\begin{aligned}
 p(b) &\leq \sum_{n=N}^{6N^2} \binom{n-1}{N-1} (1-b)^N \\
 &\quad + \sum_{n=6N^2+1}^{\infty} \binom{n-1}{N-1} (1-b)^N (1-c_1(N))^{\lfloor (n-1)/(6N) \rfloor - (N-1)} \\
 &\leq (1-b)^N \left(\sum_{n=N}^{6N^2} \binom{n-1}{N-1} + \sum_{n=6N^2+1}^{\infty} n^N (1-c_1(N))^{\lfloor (n-1)/(6N) \rfloor - (N-1)} \right) \\
 &= c(N)(1-b)^N,
 \end{aligned}$$

for a suitable constant $c(N) < \infty$.

We can now substitute our upper bound for $p(b)$ back into our integral for $E(S)$, giving

$$E(S) = \int_0^1 \frac{N(1-b)^{N-1}}{p(b)} db \geq \frac{N}{c(N)} \int_0^1 \frac{(1-b)^{N-1}}{(1-b)^N} db = \frac{N}{c(N)} \int_0^1 \frac{1}{1-b} db = \infty.$$

Let A be the event that block 0 is bad. We have already noted that if block 0 is bad then the first avalanche can only end during a good epoch. This gives

$$E(D(\Lambda_N)) \geq E(D(\Lambda_N) \mid A) P(A) \geq E(S \mid A) P(A),$$

and since $P(A) > 0$ it suffices to prove that $E(S \mid A) = \infty$.

Lemma 2.3. $E(S \mid A) = \infty$.

Proof. Writing $F_{B \mid A}$ for the conditional distribution function of B given the event A , we have

$$E(S \mid A) = \int_0^1 E(S \mid A, B = b) dF_{B \mid A}(b). \tag{2.1}$$

We now claim that

$$E(S \mid A, B = b) = E(S \mid B = b). \tag{2.2}$$

To see this, observe that when we condition on $B = b$ further knowledge about updates in block 0 is irrelevant for the distribution of S ; indeed, the distribution of S depends only on B , with $E(S \mid B = b)$ nondecreasing in b .

Furthermore, we claim that

$$F_{B \mid A}(b) \leq F_B(b), \tag{2.3}$$

where F_B denotes the distribution function of B . To see this, we let $U = \max\{U_0, \dots, U_{6N-1}\}$ and argue as follows:

$$\begin{aligned}
 F_{B \mid A}(b) &= P(B \leq b \mid A) \\
 &= P(B \leq b \mid B > U_0 > U_1 > \dots > U_{6N-1}) \\
 &= P(B \leq b \mid B > U, U_0 > U_1 > \dots > U_{6N-1}).
 \end{aligned}$$

Since the event $\{U_0 > U_1 > \dots > U_{6N-1}\}$ is independent of (U, B) , we can remove it from the conditioning event, and find that

$$F_{B|A}(b) = P(B \leq b \mid B > U) \leq P(B \leq b),$$

since B and U are independent and it is well known that, in general, conditioning a random variable B to be larger than another, independent, random variable U makes B stochastically larger.

Taking (2.1), (2.2), and (2.3) together with the fact that $E(S \mid B = b)$ is nondecreasing in b , we finally obtain

$$E(S \mid A) \geq \int_0^1 E(S \mid B = b) dF_B(b) = E(S),$$

which by Lemma 2.2 is equal to infinity, proving the result.

The proof above only gives the result for Λ_N . However, the only part of the proof that really utilises the circular graph structure is Lemma 2.1. If, for any graph G_N , a result equivalent to Lemma 2.1 can be proven, then the rest of the proof will follow as above (up to a few changes of constants). Therefore, the goal is to come up with a similar notion of a bad block that again ends with all fitnesses below the threshold. Below is a recipe for doing this for any graph G_N .

A *closed walk* in a graph is defined to be an ordered collection of vertices (v_1, v_2, \dots, v_n) such that v_i and v_{i+1} are neighbours for all i , $1 \leq i \leq n - 1$, and v_n and v_1 are neighbours. Note that the walk is allowed to visit the same vertex or traverse the same edge on more than one occasion. For any finite connected graph, it is possible to find a closed walk that visits every vertex of the graph at least once.

Consider a minimal such closed walk on G_N . This walk is used to update the Bak–Sneppen model. Recall that the updates are drawn from a sequence of independent uniform(0, 1) random variables U_0, U_1, U_2, \dots . Let the vertex with the initial minimal fitness be v . Locate v in the walk. It may occur more than once, in which case just pick one of these occurrences. Now update this vertex and its neighbours in such a way that the last random variable from the sequence is used to update the neighbour of v that occurs next in the walk. For example, if we use U_0, U_1 , and U_2 , then U_2 will be the new fitness of the vertex that comes next in the walk. Now we continue along the walk until we reach the vertex with the new minimal fitness, and update as above.

Now consider what happens if we have a bad block. Recall that a bad block is a block in our sequence of random variables where the random variables are decreasing and below the threshold. If, at any point during the bad block, the minimal fitness comes from the last random variable of the bad block, then the location of the minimal fitness will follow the walk updating all the vertices.

For any finite connected graph G_N , we can bound from above the number of random variables needed in a bad block to ensure that the minimal fitness comes from the last random variable of the bad block. Note that here we are measuring the length of a bad block in terms of random variables rather than update steps (as before). Denote this bound by l_1 (in the case of $G_N = \Lambda_N$, we used $l_1 = 3N$ in the proof of Lemma 2.1). Let l_2 be the number of random variables used to travel around our walk (in the case of $G_N = \Lambda_N$, we used $l_2 = 3N$ in the proof of Lemma 2.1). Then a bad block of length $l = l_1 + l_2$ will ensure that all fitnesses are below the threshold, as desired.

3. Proof of Theorem 1.2

Let $G_N \in \mathcal{G}_N$. Recall that Theorem 1.2 stated that an avalanche on G_N at a threshold distributed as the minimum of $N + 1$ independent, uniform random variables has finite expected duration. Our approach is to couple the Bak–Sneppen model with a similar, simpler model whose avalanche durations are more easily calculated. In the Bak–Sneppen model we have the vertices arranged on a circle, so each vertex has only two neighbours. For the Bak–Sneppen model on the *complete graph* K_N , updating the minimal vertex and its neighbours will update the entire graph. This yields a fairly uninteresting model, with no time dependence or self-organised critical behaviour in the infinite limit. However, in this model the expected duration of an avalanche at a threshold b is easy to calculate and is equal to $(1 - b)^{-N} < \infty$.

Intuitively one might think that an avalanche on K_N at a threshold b would be longer than an avalanche on G_N at the same threshold. This is because the original model eliminates the smallest value and its neighbours, leaving some old fitnesses that are certainly greater than the minimum. It seems plausible that the distributions of these other vertices are stochastically larger than uniform(0, 1) random variables. This intuition is correct, as we will now demonstrate.

Since, as in the Bak–Sneppen model on Λ_3 , we have an explicit expression for the expected duration of an avalanche on K_N at a fixed threshold, we can calculate $D_n(K_N)$ by integration. We find that

$$D_n(K_N) = \int_0^1 \frac{n(1 - b)^{n-1}}{(1 - b)^N} db < \infty \quad \text{for all } n > N,$$

where the expression $n(1 - b)^{n-1}$ is the density of the threshold, which is set by the minimum of n uniform random variables. Thus, we need to provide a coupling of the Bak–Sneppen models on K_N and G_N to show that the expected duration of a b -avalanche is shorter for G_N and, therefore, that the initial avalanche has finite expectation when $n > N$.

The main difficulty to overcome when coupling these two models is that the time dependencies are completely different. The Bak–Sneppen model on G_N depends heavily upon the values at the previous step, whereas on K_N the fitnesses are completely independent of previous fitness values. Our approach is as follows. We consider an alternative way of constructing a Bak–Sneppen avalanche that does not fix the fitness values and has the same duration distribution. We then demonstrate how this construction can be realised using a sequence of independent uniform(0, 1) random variables and in this way coupled with the model on K_N . Readers should note that an approach similar to this has also been used to compare Bak–Sneppen avalanches to site percolation [6]. Before proving that it has the desired properties, we give an explicit example of this coupling.

Consider a Bak–Sneppen b -avalanche on G_N . Pick a vertex uniformly at random to be the origin of the avalanche, and give it fitness b . Let all the other fitnesses be uniformly distributed on $(b, 1)$. At the first update step we generate new fitnesses that are associated with (but not assigned to) the origin and its neighbours. Furthermore, we generate new fitnesses for the remaining vertices, but this time they are not uniform(0, 1), but uniform($b, 1$). We find the minimum of these values and assign it to the corresponding vertex.

Now, in contrast to the usual treatment, we discard the remaining fitness values and merely record the conditional distributions of the fitnesses given that the fitnesses must be greater than the given minimal value. We iterate this process by again generating new fitnesses corresponding to the minimal vertex and its neighbours, and also generate new fitness values corresponding to the other vertices contained in G_N , according to their conditional distributions. Again we locate

and fix the minimal value before discarding the other fitnesses and recording their conditional distributions (i.e. given both their previous distribution and the fact that they are greater than the new minimal value). Note here that at each time step only one vertex has a fixed value, namely the minimal one.

If at any point we wish to have exact fitnesses for the Bak–Sneppen model, the associated fitness values can be fixed to all vertices. This is similar to the induction step above, except that after locating and recording the minimum value the remaining fitnesses are fixed rather than discarded. Fixing all the fitness values ends this construction, since it requires the minimum fitness to be the only fixed value. To continue the avalanche after the fixing, we must therefore return to the original method for realising the Bak–Sneppen model.

Proposition 3.1. *The above construction of a b -avalanche has the same range and duration distributions as a Bak–Sneppen b -avalanche.*

Proof. The construction above follows the rules for a Bak–Sneppen model in that, at each time step, the vertex with the minimal fitness is updated along with the fitnesses of its neighbours. The only difference is in the use of information. Normally, the fitnesses of all the vertices are known at all time steps, but here we only keep track of distributions. By construction, these distributions would be the same as the fitness distributions in a Bak–Sneppen avalanche if the precise fitness values were unknown. In particular, the location and magnitude of the minimal vertex has the correct distribution. Since the location and magnitude is at all times distributed in the same way as in a normal Bak–Sneppen avalanche, the required equivalence is now clear. The duration is determined by the magnitudes of the sequence of minima, and the range by their respective locations.

We remark that the idea of constructing the Bak–Sneppen process in this way is closely related to the ‘locking thresholds representation’ introduced in [10]. With this new construction of a b -avalanche we are now ready to describe the coupling.

We start by generating a sequence of uniform(0, 1) random variables, X_1, X_2, \dots , that shall be used to update the fitnesses on K_N . Contrary to earlier notation, for this proof we enumerate the vertices by $1, 2, \dots, N$; the fitnesses on K_N will respectively be X_1, X_2, \dots, X_N after the first time step, $X_{N+1}, X_{N+2}, \dots, X_{2N}$ after the second time step, and so on. We use the same random variables to generate the fitnesses on G_N . However, in the Bak–Sneppen model on G_N , we do not always wish to use uniform(0, 1) random variables: before they can be used, we have to transform them. This is done by simply taking the inverse of the distribution functions (sometimes referred to as probability integral transformation).

It is perhaps worth describing the exact nature of these transformations more precisely. A random variable with any invertible distribution function, F , can be generated by taking a uniform(0, 1)-distributed random variable, X , and taking $F^{-1}(X)$. Thus, our approach here is to use this sort of transformation so that the same sequence, (X_i) , can be used both for the model on K_N and the model on G_N .

We consider two cases. The first possibility is that, on the one hand, the avalanche on K_N stops. In this case we fix the fitness values on G_N to check if the Bak–Sneppen avalanche has also stopped. This is done as above: the random variables are transformed to the correct distributions and then set to be the fitnesses of the vertices. If, on the other hand, the avalanche on K_N is still in progress, then we carry on with the Bak–Sneppen avalanche on G_N according to our new construction.

We are now ready to give a specific example (in this case the Bak–Sneppen model on $G_N = \Lambda_N$), as we know how to update the fitness distributions in the coupled Bak–Sneppen

TABLE 1.

Bak–Sneppen model on K_5					Comments	Bak–Sneppen model on Λ_5				
1	2	3	4	5	Vertex	1	2	3	4	5
$> b$	\underline{b}	$> b$	$> b$	$> b$	Initial setup	$> b$	\underline{b}	$> b$	$> b$	$> b$
X_1	X_2	X_3	X_4	X_5	Update K_5	> 0	> 0	> 0	$> b$	$> b$
X_1	X_2	$\underline{X_3}$	X_4	X_5	Update Λ_5	> 3	> 3	$\underline{X_3}$	$> b$	$> b$
X_6	X_7	X_8	X_9	X_{10}	Update K_5	> 3	> 0	> 0	> 0	$> b$
$\underline{X_6}$	X_7	X_8	X_9	X_{10}	Update Λ_5	> 9	> 9	> 9	$\underline{X_9}$	$> b$
X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	Update K_5	> 9	> 9	> 0	> 0	> 0
X_{11}	$\underline{X_{12}}$	X_{13}	X_{14}	X_{15}	Update Λ_5	$> x$	\underline{x}	$> x$	$> x$	$> x$

model. See Table 1. Each time step is split into two lines in order to illustrate the mechanism of the coupling more clearly. In the first line we update on K_N and write down the marginal distributions for the Bak–Sneppen model at that update. On the next line the fitness values for K_N are used to calculate the exact minimum for the Bak–Sneppen model and to update the remaining marginal distributions. For brevity we use the shorthand ‘ $> b$ ’ for uniform($b, 1$) and ‘ > 3 ’, for example, for uniform($X_3, 1$). We use an underscore to denote the minimum fitness value. It is important to remember that the minimum on K_N does not play an explicit role in determining the behaviour of the Bak–Sneppen model on G_N . The underscore is used here only to highlight the types of behaviour that can be observed from this coupling.

Since vertex 2 has the minimum value initially, the new fitnesses of vertices 1, 2, and 3 will be uniform($0, 1$). We also update the fitnesses on K_N . Note that in this example the minimal fitness on K_N is below b at all times, so the avalanche does not stop during the steps we have detailed. Therefore, we do not fix the exact fitness values in the coupled Bak–Sneppen model, but instead use the random variables X_1, X_2, \dots, X_5 to find the location of the minimal fitness in the Bak–Sneppen model. On K_N , X_3 takes the minimum value, so vertex 3 in the Bak–Sneppen model is minimal with fitness X_3 . This is because, of all the transformed values, X_3 is necessarily the smallest, as the transformations to X_4 and X_5 make them larger and no transformations are required for X_1, X_2 , and X_3 . Since the avalanche is still in progress, $X_3 < b$ and, so, the distributions of the fitnesses of vertices 4 and 5 remain the same, whereas for vertices 1 and 2 they are now uniform($X_3, 1$).

The next time step follows in a similar vein, except that after the transformations we find that X_9 is the minimal transformed value even though, before transformation, X_6 was smaller. This means that the location of the minimal vertex in the two models is now different. It is also worth noticing that X_9 must be greater than X_3 on account of how the fitness distributions have been updated. The final time step detailed gives an example of the more complicated behaviour that can be observed. Here vertex 2 is minimal in both models, but in the Bak–Sneppen model the fitness value at vertex 2 is x rather than X_{12} , where $x = X_9 + (1 - X_9)X_{12}$. This is because we have had to transform the value of X_{12} so that it has the correct distribution.

To conclude this example, let us consider the case in which the avalanche in the new model stops. This means that $\min\{X_{11}, X_{12}, X_{13}, X_{14}, X_{15}\} > b$. We then fix the fitnesses for the Bak–Sneppen model, giving the fitnesses $X_9 + (1 - X_9)X_{11}, X_9 + (1 - X_9)X_{12}, X_{13}, X_{14}, X_{15}$, respectively.

Having given a description of the coupling, it remains to confirm that this coupling has the desired property, namely that the Bak–Sneppen avalanche on G_N cannot outlive the avalanche on K_N .

Proposition 3.2. *The Bak–Sneppen avalanche finishes no later than the end of the new process.*

Proof. This proposition is proved by showing that the conditional distributions of the Bak–Sneppen model always belong to a certain family of distributions. Observe that initially we have one fixed fitness, with all the others uniformly distributed above the threshold b . When a vertex or one of its neighbours is minimal its fitness distribution is reset to be uniform(0, 1). The only other way the fitness distribution can be altered is by conditioning it on being bigger than a given value. Conditioning a uniform random variable to be bigger than a given value merely restricts the variable to be uniform on the part of its original range that is above the given value. It is obvious that a uniform(0, 1) random variable is stochastically dominated by a uniform(x , 1) random variable when $x \geq 0$. Thus, transforming the fitnesses on K_N to make them applicable to the Bak–Sneppen model on G_N can never make them smaller. Hence, if the fitnesses on K_N are all greater than the threshold b , then so must be the fitnesses of the Bak–Sneppen model on G_N .

This coupling shows that $D_n(K_N) < \infty$ implies that $D_n(G_N) < \infty$. Hence, we have proved Theorem 1.2.

Acknowledgement

We would like to thank an anonymous referee for a number of very good observations which inspired us to present our results in this generality, rather than only on Λ_N .

References

- [1] BAK, P. (1996). *How Nature Works*. Copernicus, New York.
- [2] BAK, P. AND PACZUSKI, M. (1995). Complexity, contingency and criticality. *Proc. Nat. Acad. Sci. USA* **92**, 6689–6696.
- [3] BAK, P. AND SNEPPEN, K. (1993). Punctuated equilibrium and criticality in a simple model of evolution. *Phys. Rev. Lett.* **71**, 4083–4086.
- [4] BAK, P., TANG, C. AND WIESENFELD, K. (1987). Self-organized criticality: an explanation of $1/f$ noise. *Phys. Rev. Lett.* **59**, 381–384.
- [5] FELLER, W. (1966). *An Introduction to Probability Theory and Its Applications*, Vol. 2. John Wiley, New York.
- [6] GILLETT, A., MEESTER, R. AND NUYENS, M. (2006). Bounds for avalanche critical values of the Bak–Sneppen model. To appear in *Markov Process. Relat. Fields*.
- [7] LEE, C., ZHU, X. AND GAO, K. (2003). Avalanche dynamics in the Bak–Sneppen evolution model observed with a standard distribution width of fitness. *Nonlinearity* **16**, 25–33.
- [8] LI, W. AND CAI, X. (2000). Analytic results for scaling function and moments for a different type of avalanche in the Bak–Sneppen evolution model. *Phys. Rev. E* **62**, 7743–7747.
- [9] MASLOV, S. (1996). Infinite series of exact equations in the Bak–Sneppen model of biological evolution. *Phys. Rev. Lett.* **77**, 1182–1185.
- [10] MEESTER, R. AND ZNAMENSKI, D. (2003). Limit behavior of the Bak–Sneppen evolution model. *Ann. Probab.* **31**, 1986–2002.
- [11] MEESTER, R. AND ZNAMENSKI, D. (2004). Critical thresholds and the limit distribution in the Bak–Sneppen model. *Commun. Math. Phys.* **246**, 63–86.
- [12] PACZUSKI, S., MASLOV, M. AND BAK, P. (1996). Avalanche dynamics in evolution, growth, and depinning models. *Phys. Rev. E* **53**, 414–443.
- [13] TABELOW, K. (2001). Gap function in the finite Bak–Sneppen model. *Phys. Rev. E* **63**, 047101.