# Dynamics of Model Battles:

# Markovian and strategic cases

**Linda Shanahan, Surajit Sen**

*Physics Department, State University of New York, Buffalo, New York, 14260-1500, USA*

## Abstract

We study the time evolution and the eventual outcome of processes that can be generically described as "battles." Here by battle we mean processes such as the battle for survival between two competing species trying to dominate one another for a common resource such as food as well as the simplified abstraction of a real battle between two armies. We study two types of battles. These are (i) battles where each side attacks the opponent at random with no pre-defined strategy, in which case the battle can be analyzed within the context of Markovian processes, and (ii) battles that are similar in spirit to modern, strategic land battles fought in enemy territory between "Attackers" and "Insurgents" with the assumption that the Attackers attack strategically while the Insurgents attack randomly. These latter battles can be analyzed via event-driven many body simulations which are often employed in statistical mechanics based studies of complex systems. For (i) we develop a theoretical description based upon Markov processes and show how strength in numbers and ways to kill opponents can help establish territorial dominance. For (ii) we use simulations to construct phase diagrams to establish that risk exposure and local intelligence influence an intelligent army's ability to win against a well matched insurgency in enemy territory. We close with observations on how this work can be extended to understand ways in which an intelligent army can minimize losses and maximize the chances of dominating the adversary in particularly contentious battles.

PACS Nos: 87.10.Hk;89.20.Dd,89.75.Fb,89.90.+n

## 1. Introduction

Many systems in the biological and social sciences are characterized by mobile interacting entities that exist in some confined territory and evolve in time based upon mutual, typically short ranged interactions, and a constantly changing environment [1, 2]. Modern statistical physics based approaches with attention towards understanding of macroscopic behavior of such systems have typically been invoked to analyze many such systems [3]. The discrete time evolution process in these systems is usually event-driven and dictated by strategy and response protocols to the changes in the environment. Further, these systems are often "open" – by this we mean that new entities can enter and/or leave the system. There is no theoretical way to describe these so called *complex adaptive systems*. A rapidly growing body of literature on work along these lines, such as those of consensus formation [4], studies of extinction-rate and species abundance in nature, on the evolution of viruses in an adaptive immune system [5] and more can now be found [6].

While the event driven evolution of these types of systems are adaptive in that the dynamics is affected by the unpredictability of the environment, it is instructive to first examine the simplest systems of interest. We start with two-party systems. Each party's troops are regarded as *completely non-interacting* except that they can attack to kill or partially negate each other when they share the same location with an adversary. Fixed numbers of entities are introduced into the system at randomly located sites. Thus, we only let a series of random events dictate the *Markovian* battle between two adversaries. We allow the sides to have the same or different killing power, which we refer to as the *force level* in this work. This is the study we address in Sec. 3.1 below.

Next, to develop an understanding of the simplest of *strategic* battles, we introduce some specifics to keep the treatment manageable. We restrict ourselves to strategic land battles fought in enemy territory. We assume that this territory is adequately described by a $2D$ square lattice. Though assuming such a simple lattice may seem restrictive at first, let us recall that most urban areas where contentious battles may be fought have some form of characteristic geometry. In this context, a 2D square net is not an unreasonable picture. We assume that the battle transpires between an *intelligent attacker* and an opposing insurgency that strikes unpredictably – hence randomly [2,7-9]. To focus on the interesting cases of battles between well matched adversaries, we allow each side to have identical force levels. In spite of considerable work on battle studies [10, 11], it appears that much remains to be learnt about the fundamental characteristics of such simple *well-matched* battles [12]. We model battles that are completely mission oriented as opposed to those desiring annexation of new territories. Thus, which party occupies more territory plays no role in determining which side wins the battle in our studies. Winning is based upon the ability of a side to decimate the opponent's ability to fight [13, 14]. The evolution of these battles is discussed in Sec. 3.2.

## 2. Results and Discussion

Markovian battles are devoid of strategy and memory. The outcome of a step depends only on what happened in the previous step. We use one step as our time unit. Hence not surprisingly, the size of each troop, both in the reserves where they are being drawn from and on the lattice where the battle is being waged, turns out to be a crucial determinant of the outcome of the battle. We assume that for both Markovian and strategic battle studies, the troop size in the reserves is the same for both parties. However, the percentage of lattice sites being occupied by any given side was treated as a variable and was used to control what we called the “exposure” of each side to enemy attacks. We found that in Markovian battles, risk exposure played a weak role. The party with more troops in the lattice fared better and this was irrespective of the force levels used. In other words, presence of more troops did not lead to greater risk of exposure of the troops to enemy attacks. As we shall see, this is not the case in strategic battles, where risk exposure is a crucial factor in determining whether or not a side wins a battle. Given that both sides attack randomly and no one has any particular strategic advantage over the other, we label one side Red and the other Blue when describing the Markovian system studies. The force levels of Reds and Blues played a role in determining the time rate at which the opponent was eliminated in the course of the battle. Eqs. (3), (4), (7), (9), (14)-(16), and (18)-(22) describe the distribution of troops on the lattice, which turns out to be a normal distribution for systems with various force levels and troop distributions in the battlefield. The presented treatment develops formulae for the mean value and the variance of this normal distribution for arbitrary force levels irrespective of whether the force levels on either side are the equal or unequal.

Dynamics of simple strategic land battles between two well matched parties, in this case between “intelligent” Attackers and Insurgents, provide an example of a complex adaptive system with history-dependent evolution. Here we have borrowed ideas from three standard approaches often used to study various dynamical many body systems. These are (i) a variant of a lattice-gas-like model [15], (ii) concept of height differences at a given site often used in surface roughness studies [16], and (iii) neighbor search algorithms commonly invoked in Molecular Dynamics simulations to calculate local forces on moving atoms and molecules [17]. In our work, the value or weight of each lattice site at a time step reflects the net sum of forces deployed by the Attackers and the Insurgents at that time.  The Insurgents attack randomly whereas the Attackers act by strengthening near Insurgent strongholds to locally decimate the strengths of the Insurgents. Our studies for well matched battles show that when the force levels used in the combat are sufficiently low, the Attackers cannot win unless the numbers of Attackers and Insurgents in the battle field are at least comparable and there is *adequate short-range intelligence* to strongly influence the Attacker's moves. When the battle involves higher force levels, the Attackers can potentially dominate the Insurgents with significantly smaller troop levels. However, for such success, knowledge of the local troop distribution (or local intelligence that dictates the strategy of the Attackers) must be sufficiently short ranged. As we shall see, local deployments made on the basis of information across large enough distance scales may hinder the ability of the Attackers to neutralize the local attacks of the insurgency and may in turn negatively affect the Attacker's ability to win a battle. In all of our studies we observed that too much exposure of any party to attacks adversely affected its ability to win a battle.

In what follows, we present the details of our model, followed by the results. The Marovian battle is presented in Sec. 3.1. The strategic battle is presented in Sec. 3.2.

## 3. Materials and Methods

In Sec. 3 we shall consider the evolution of both Markovian battles in which each party is constrained to strike in random locations with pre-set probabilities and Strategic battles where elements of one side are able to make intelligent battle moves based upon the threats posed in their neighborhoods by the presence of an insurgent army that can attack randomly. Sec. 3.1 describes our work on Markovian battles. Sec. 3.2 presents the details of the study on strategic battles. The sides are equally matched in terms of fire power and troop sizes in both cases.

However, each side will typically attack differing percentages of the field sites, thus exposing itself to differing risk levels. In Markovian battles, the winner gains territorial dominance. In strategic battles the winner manages to decimate the opponent's ability to continue the battle whether or not territorial dominance is established. Thus, the two battles proceed very differently and we treat them here as separate problems.

### *3.1. Markovian Battle: The Model*

In Sec. 3.1.1 we introduce the Markovian approach. Details of calculations for the case of lowest possible force levels (Case 1) used in the battles are presented via four different methods in Sec 3.1.2. The more challenging cases of multiple force levels are presented under Case 2 and Case 3 in Sections 3.1.3. and 3.1.4.

#### *3.1.1. Introductory Statements*

We establish the parameters for a simple random battle below and illustrate how the model lends itself to modification for more complicated systems. The battlefield sites may be taken as a set of randomly located points. *No spatial structure* or distance metric is needed. For the simple model, we allow both sides the same fire power and troop reserves. Individual combatants are assigned numeric strengths. Deployment to the battlefield sites is made randomly, but according to constant, preset, percentages assigned to each side. Casualties result when opposing forces occupy the same site. The battle proceeds for some predetermined time or until some desired outcome is achieved.

For simplicity, we consider a set of coplanar battlefield points, identified by their locations, $K(i, j)$, using a $50 \times 50$ lattice for illustration only. Since the statistics will be intrinsic, the results can be applied to any field size or configuration, with larger sets of points, of course, achieving better correlation with the expected probability distribution values. Insurgents and

Attackers are designated as Red and Blue, respectively. Blue forces will be represented by negative numerical values and Red forces by positive values. Our simplest battle will consist of totally random striking of the battlefield, $K$. Blue attack forces having strength values of $B = \{(-1, w_{-1}), (-2, w_{-2}), (-3, w_{-3}), ...\}$ (where $w_m$ is a multiplicative weight factor to $m$ having a value of $0 \le w_m \le 1$ as desired and applies to *all the randomly distributed sites* with a deployment value $m$) land at each attack site, and Red Insurgent forces with strength values of $R = \{(+1, w_{+1}), (+2, w_{+2}), (+3, w_{+3}), ...\}$ counter at the randomly chosen defense sites. The battle will be symmetric if the same absolute maximum values of $B$ and $R$ are used, and asymmetric if not. Calculations will be presented for $B = \{-1\}$, $R = \{+1\}$ and $B = \{-1,-2\}$, $R = \{+1,+2\}$. The general ideas can, however, be applied to any magnitudes or $B$ and $R$. Extension to an asymmetric type battle, where $B_{\max} \ne R_{\max}$ will be considered later in Sec. 3.1.3. A one-to-one annihilation ratio will be used, so that equal Blue and Red forces occupying the same site will sum to zero.

At each iteration or time step, forces will be deployed to some (constant) percentages of randomly chosen sites, with the percentages specified by $\varepsilon_R$ and $\varepsilon_B$. Forces can accumulate at each site, thus the value of $K(i, j)$ after $n$ steps will be an integer, $v_{ij}^{(n)}$, such that $\{-nB_{\max} \le v_{ij}^{(n)} \le nR_{\max}\}$. The time ordered sequence of states $\{v_{ij}^{(1)}, v_{ij}^{(2)}, ...v_{ij}^{(n)}\}$ defining the value of each site $K_n(i, j)$ at a given time step forms a discrete parameter, Markov dependent sequence, or process, defined as a process where the conditional probability at step $n$ depends only on the values of $v_{ij}^{(n)}$ and $v_{ij}^{(n-1)}$, i.e.

$$p(v_{ij}^{(n)} \mid v_{ij}^{(1)}, \ldots, v_{ij}^{(n-1)}) = p(v_{ij}^{(n)} \mid v_{ij}^{(n-1)}). \tag{1}$$

The Markov chain process for $n$ steps is irreducible (each state is accessible from every other state) and aperiodic. We will make use of the Markov chain concept to find the associated probability distributions, arrive at some useful approximations, and illustrate the extension to more complex models.

***3.1.2. Details of the Calculations, Case 1: Strength values*** $R = +1, B = -1$

At $n = 0$, we let all sites have initial values $K_0(i, j) = v_{ij}^{(0)} = 0$. Let us now allow the Red force to attack with rate $\varepsilon_R$ $(0 < \varepsilon_R \le 1)$, and the Blue force with rate $\varepsilon_B$ $(0 < \varepsilon_B \le 1)$. If, for example, $\varepsilon_R = 0.5$, then approximately 1250 randomly located sites of a 2500 site lattice will receive deployments equal to $R = +1$. Conversely, a given site does *not* receive Red or Blue deployment with probabilities $(1 - \varepsilon_R) = \bar{\varepsilon}_R$, and $(1 - \varepsilon_B) = \bar{\varepsilon}_B$, respectively. When $n = 1$, each site has value $K_1(i, j) = v_{ij}^{(1)} = -1, 0, or +1$. The notation $\beta_n^{(v)}$ will be used to signify the probability of finding a value of $v$ after $n$ steps. The probability of each $v_{ij}^{(1)}$ for the first step of Case 1 is found from considering the four possible permutations of $R$ and $B$:

$$p(v_{ij}^{(1)} = -1) = p(R \to 0) \times p(B \to -1)$$

$$= (1 - \varepsilon_R)\varepsilon_B = \bar{\varepsilon}_R \varepsilon_B,$$

$$= \beta_1^{(-1)},$$

$$p(v_{ij}^{(1)} = 0) = p(R \to 0) \times p(B \to 0) + p(R \to 1) \times p(B \to -1)$$

$$= (1 - \varepsilon_R)(1 - \varepsilon_B) + \varepsilon_R \varepsilon_B = \bar{\varepsilon}_R \bar{\varepsilon}_B + \varepsilon_R \varepsilon_B$$

$$= \beta_1^{(0)},$$

$$p(v_{ij}^{(1)} = +1) = p(R \to 1) \times p(B \to 0)$$

$$= \varepsilon_R (1 - \varepsilon_B) = \varepsilon_R \bar{\varepsilon}_B$$

$$= \beta_1^{(+1)}.$$

The completeness relation is given by $\sum_{v=-n}^{+n} \beta_n^{(v)} = 1$, thus:

$$\sum_{v=-1}^{+1} \beta_1^{(v)} = \beta_1^{(-1)} + \beta_1^{(0)} + \beta_1^{(+1)}$$

$$= (1-\varepsilon_R)\varepsilon_B + (1-\varepsilon_R)(1-\varepsilon_B) + \varepsilon_R\varepsilon_B + \varepsilon_R(1-\varepsilon_B) = 1. \quad (2)$$

We can conveniently express the probability density for one step as a vector:

$$D_1 = [\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}]. \quad (3)$$

The discrete probability distribution for an arbitrary number of steps, $n$, can be found in several ways. We discuss them in the following sections.

***Method 1: Probability based approach***

We can find the discrete probability distribution for $n$ steps using the recursive probability approach used to find the step one probabilities. For $n = 2$, we utilize the values from $n = 1$, and the same set of possible permutations of $R$ and $B$, as follows:

$$p(v_{ij}^{(2)} = -2) = p[v_{ij}^{(1)} = -1] \times p\,(R \to 0) \times p\,(B \to -1) = \beta_1^{(-1)^2},$$

$$p(v_{ij}^{(2)} = -1) = p[v_{ij}^{(1)} = -1] \times p(R \to 0) \times p(B \to 0)$$

$$+\ p[v_{ij}^{(1)} = 0] \times p(R \to 0) \times p(B \to -1)$$

$$= 2\beta_1^{(-1)}\beta_1^{(0)},$$

$$p(v_{ij}^{(2)} = 0) = p[v_{ij}^{(1)} = -1] \times p\,(R \to 1) \times p\,(B \to 0)$$

$$+\ p[v_{ij}^{(1)} = 0] \times p(R \to 0) \times p(B \to 0) + p[v_{ij}^{(1)} = 0] \times p(R \to 1) \times p(B \to -1)$$

$$+\ p[v_{ij}^{(1)} = 1] \times p(R \to 0) \times p(B \to -1)$$

$$= 2\beta_1^{(-1)}\beta_1^{(+1)} + \beta_1^{(0)^2},$$

$$p(v_{ij}^{(2)}=1)=p[v_{ij}^{(1)}=1]\times p(R\to 0)\times p(B\to 0)$$

$$+\,p(v_{ij}^{(1)}=0)\times p(R\to 1)\times p(B\to 0)$$

$$=2\beta_1^{(+1)}\beta_1^{(0)},$$

$$p(v_{ij}^{(2)}=2)=p[v_{ij}^{(1)}=1]\times p(R\to 1)\times p(B\to 0)=\beta_1^{(+1)^2}.$$

We again express the discrete distribution values as a vector:

$$D_2=[\beta_1^{(-1)^2}\quad 2\beta_1^{(-1)}\beta_1^{(0)}\quad 2\beta_1^{(-1)}\beta_1^{(+1)}+\beta_1^{(0)^2}\quad 2\beta_1^{(0)}\beta_1^{(+1)}\quad \beta_1^{(+1)^2}]. \tag{4}$$

This type of recursion approach is obviously cumbersome for large $n$.

***Method 2: Markov chain approach***

To construct a Markov chain which will allow for calculation of the discrete probability distribution for arbitrary $n$, one must define probability matrices for each step in the process, from step 1 to step $n$ (see Clarke [18], pp. 217-237, for a detailed discussion of this approach). The one-step transition matrices $P_{(n-1,n)}$ define the one-step transition probabilities from state $n-1$ to state $n$. In general, the matrix $P_{(n-1,n)}$ is a $(2n-1)\times(2n+1)$ matrix where each entry represents the probability of going from the row state to the corresponding column state. We can define the first-step probability vector for Case 1 as $P_{(0,1)}=D_1=[\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}]$, recalling from the completeness relation (2), that the entries sum to 1. In this case, one iteration can only increase or decrease a given site value by one unit. Thus for subsequent iterations each site value $K(i,j)$ increases, decreases, or remains the same according to the same probability vector, $P_{(0,1)}$, and the sequence is said to have stationary transition probabilities. For the $n^{th}$ iteration the corresponding matrix will be:

$$
P_{(n-1,n)} = \begin{array}{c|ccccccc|}
 & -n & (-n+1) & (-n+2)\ldots & & \ldots & \ldots & +n \\
\hline
-n+1 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & 0 & 0 & 0 & 0 \\
-n+2 & 0 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & 0 & 0 & 0 \\
\ddots & 0 & 0 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & 0 & 0 \\
\ddots & . & . & . & . & . & . & . \\
n-1 & 0 & 0 & 0 & 0 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)}
\end{array} . \tag{5}
$$

From the completeness relation (1), the entries of each row sum to 1, hence each transition matrix is also stochastic [18]. For a system with three possible transitions, the matrices are *tridiagonal*, and straightforward to construct. The probability density function of site values for the $n^{th}$ iteration can be found from matrix multiplication, or the inner product, of the single-step matrices:

$$
D_n = P_{(0,1)} \cdot P_{(1,2)} \cdot P_{(2,3)} \ldots \cdot P_{(n-1,n)} , \tag{6}
$$

yielding the desired $1 \times (2n+1)$ vector containing the probabilities:

$$
D_n = [\beta_n^{(-n)} \quad \beta_n^{(-n+1)} \quad \ldots \quad \ldots \quad \beta_n^{(n-1)} \quad \beta_n^{(n)}] . \tag{7}
$$

For example, the Case one 2-step distribution matrix is (again):

$$
D_2 = P_{(0,1)} \cdot P_{(1,2)} = [\beta_1^{(-1)} \quad \beta_1^{(0)} \quad \beta_1^{(+1)}] \cdot \begin{vmatrix} \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & 0 & 0 \\ 0 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & 0 \\ 0 & 0 & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} \end{vmatrix}
$$

$$
= [\beta_1^{(-1)^2} \quad 2\beta_1^{(-1)}\beta_1^{(0)} \quad 2\beta_1^{(-1)}\beta_1^{(+1)} + \beta_1^{(0)^2} \quad 2\beta_1^{(0)}\beta_1^{(+1)} \quad \beta_1^{(+1)^2}] . \tag{4}
$$

The Markov chain method has the advantage of being simple to construct and calculate computationally, and allows for easy modification of the transition probabilities.

***Method 3: Graphical approach***

A graphical approach can be used to check or visually illustrate the matrix and algebraic methods [19]. The order or number of possible values, of a graph for $n$ steps is $(2n+1)$ . The set of directed edges, $A$, is a set or ordered pairs representing the start and endpoints of each arrow.

The graph size (number of edges in $A$), can be determined by subtracting from the total number of *possible* connections, i.e. $(2n+1)^2$, those connections coming *from* $-n$ or $+n$ (equal to $2(2n+1)$), and those connections jumping by more than one integer in value (equal to $(2(n-1))\times(2n-1)$). Following these machinations one finds the size to be $(6n-3)$.

The weighted, directed graphs shown in Figs. 1 and 2 illustrate the possibilities for one step and two steps of Case 1, respectively. To determine, for example, the probability of reaching $v_{ij}^{(2)}=-1$, there are two valid paths $[0\rightarrow -1\rightarrow -1]$ and $[0\rightarrow 0\rightarrow -1]$ each having respective probabilities $\beta_1^{(-1)}\beta_1^{(0)}$, summing to $2\beta_1^{(-1)}\beta_1^{(0)}$ as found in the vector in Eq. (4). Due to the symmetry of the transition probabilities, it is useful to recognize that the $n^{th}$ distribution can also be found algebraically using the completeness relation in Eq. (2). Taking $(\beta_1^{(-1)}+\beta_1^{(0)}+\beta_1^{(+1)})^n$ and collecting the terms appropriately, one obtains a summation of the distribution values. In this instance,

$$(\beta_1^{(-1)}+\beta_1^{(0)}+\beta_1^{(+1)})^2=\beta_1^{(-1)^2}+2\beta_1^{(-1)}\beta_1^{(0)}+2\beta_1^{(-1)}\beta_1^{(+1)}+\beta_1^{(0)^2}+2\beta_1^{(0)}\beta_1^{(+1)}+\beta_1^{(+1)^2}. \quad (8)$$

***Method 4: Gaussian distribution approximation***

The probability distribution or distribution of magnitudes on the field sites, for Case 1 can be well-approximated for the $n^{th}$ iteration by a normal distribution of the form:

$$f_n(x)=\frac{1}{\sqrt{2\pi\sigma_n}}e^{(-(x-\mu_n)^2/2\sigma_n)}, \quad (9)$$

using an average or mean value $\mu_n=n(\varepsilon_R-\varepsilon_B)$ and variance $\sigma_n=[n\varepsilon_B(1-\varepsilon_B)+n\varepsilon_R(1-\varepsilon_R)]=[n(\varepsilon_B\bar{\varepsilon}_B+\varepsilon_R\bar{\varepsilon}_R)]$ as derived below, and illustrated in Fig. 3. The average $\mu_n=n(\varepsilon_R-\varepsilon_B)$ can be found by induction. Let us define the matrix of possible numerical site values as $M_{-n,n}=[-n \quad -n+1 \quad \dots \quad -1 \quad 0 \quad 1 \quad \dots \quad n-1 \quad n]$.

Since the average value is $\sum_i x_i \cdot p(x_i)$, we can find the average site value from $D_n \cdot M_{-n,n}$. For

$n = 1$,

$$\mu_1 = P_{(0,1)} \cdot M_{-1,1} = [\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}] \cdot \begin{bmatrix} -1 \\ 0 \\ +1 \end{bmatrix} = -\beta_1^{(-1)} + \beta_1^{(+1)}$$

$$= -(1-\varepsilon_R)\varepsilon_B + (1-\varepsilon_B)\varepsilon_R = 1(\varepsilon_R - \varepsilon_B). \tag{10}$$

Assume that for $n = m$:

$$\mu_m = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m-1,m)} \cdot M_{-m,m} = m(\varepsilon_R - \varepsilon_B).$$

For $n = m+1$:

$$\mu_{m+1} = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m-1,m)} \cdot P_{(m,m+1)} \cdot M_{-(m+1),(m+1)}.$$

Considering the product $P_{(m,m+1)} \cdot M_{-(m+1),(m+1)}$, the first line will be

$$[P_{(m,m+1)} \cdot M_{-(m+1),(m+1)}]_1 = [\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}] \cdot \begin{bmatrix} -(m+1) \\ -m \\ -m+1 \end{bmatrix} = -(\beta_1^{(-1)} + \beta_1^{(0)} + \beta_1^{(+1)})m + (-\beta_1^{(-1)} + \beta_1^{(+1)})$$

$$= -m + \mu_1,$$

where we have made use of the completion relation in Eq. (2). The full product, correspondingly, is equal to $M_{-m,m} + \mu_1[1]_{2m-1,1}$ where, for example, $[1]_{3,1}$ is equal to $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$, hence

$\mu_{m+1} = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m-1,m)} \cdot (M_{-m,m} + \mu_1[1]_{2m-1,1})$. From

$P_{(0,1)} \cdot \mu_1[1]_{3,1} = \mu_1\, [\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}] \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \mu_1(\beta_1^{(-1)} + \beta_1^{(0)} + \beta_1^{(+1)}) = \mu_1[1]_{1,1}$. Or,

$$P_{(m-1,m)} \cdot \mu_1[1]_{2m-1,1} = \mu_1 \sum_j P_j(1) = \mu_1[1]_{2m-3,1}\,.$$

With the previous assumption on $\mu_m$, we can extrapolate to find that

$$\mu_{m+1} = \mu_m + \mu_1 = m(\varepsilon_R - \varepsilon_B) + (\varepsilon_R - \varepsilon_B)$$

$$= (m+1)(\varepsilon_R - \varepsilon_B)\,. \qquad (11)$$

The variance $\sigma_n = n(\varepsilon_B \bar{\varepsilon}_B + \varepsilon_R \bar{\varepsilon}_R)$ can be found in a similar manner.

Defining the variance as $\sum_i p(x_i)(x_i - \mu)^2$, we can find the $\sigma_n$ from $D_n \cdot [(M_{-n,n} - \mu_n)^2]$.

For $n = 1$,

$$\sigma_1 = [\beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}] \cdot \begin{bmatrix} (-1-\mu_1)^2 \\ (0-\mu_1)^2 \\ (+1-\mu_1)^2 \end{bmatrix} = (\beta_1^{(-1)} + \beta_1^{(+1)}) + (2\beta_1^{(-1)} - 2\beta_1^{(+1)})\mu_1 + \mu_1^2$$

$$= (\beta_1^{(-1)} + \beta_1^{(+1)}) - 2\mu_1^2 + \mu_1^2 = -\mu_1^2 + (\beta_1^{(-1)} + \beta_1^{(-+1)})$$

$$= -(\varepsilon_R - \varepsilon_B)^2 + (1-\varepsilon_R)\varepsilon_B + (1-\varepsilon_B)\varepsilon_R$$

$$= (1-\varepsilon_R)\varepsilon_R + (1-\varepsilon_B)\varepsilon_B\,. \qquad (12)$$

Assume that for $n = m$:

$$\sigma_m = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m-1,m)} \cdot [(M_{-m,m} - \mu_m)^2] = m(1-\varepsilon_R)\varepsilon_R + m(1-\varepsilon_B)\varepsilon_B\,.$$

For $n = m+1$:

$$\sigma_{m+1} = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m,m+1)} \cdot [(M_{-(m+1,m+1)} - \mu_{m+1})^2]\,.$$

Considering the product $P_{(m,m+1)} \cdot [(M_{-(m+1),(m+1)} - \mu_{m+1})^2]$, the first line will be

$$[P_{(m,m+1)} \cdot [(M_{-(m+1),(m+1)} - \mu_{m+1})^2]_1 = [\beta_1^{(-1)} \ \beta_1^{(0)} \ \beta_1^{(+1)}] \cdot \begin{bmatrix} (-(m+1)-(m+1)\mu_1)^2 \\ (-m-(m+1)\mu_1)^2 \\ (-m+1-(m+1)\mu_1)^2 \end{bmatrix}$$

$$= m^2 - 2m\mu_1 + 2m(m+1)\mu_1 + \beta_1^{-1} + \beta_1^{+1} - 2(m+1)\mu_1^2 + (m+1)^2\mu_1^2$$

$$= (-m-\mu_m)^2 + \sigma_1 ,$$

where we have again made use of the completeness relations. The full product is equal to $(M_{-m,m} - \mu_m)^2 + \sigma_1[1]_{m,1}$, and hence

$$\sigma_{m+1} = P_{(0,1)} \cdot P_{(1,2)} \cdot \ldots \cdot P_{(m-1,m)} \cdot [(M_{-m,m} - \mu_m)^2 + \sigma_1[1]_{m,1})] .$$

From

$$P_{(m-1,m)} \cdot \sigma_1[1]_{m,1} = \sigma_1 \sum_j P_j(1) = \sigma_1[1]_{m-2,1} ,$$

and our previous assumption on $\sigma_m$, we can extrapolate to find:

$$\sigma_{m+1} = \sigma_m + \sigma_1 = (m+1)\varepsilon_R(1-\varepsilon_R) + (m+1)\varepsilon_B(1-\varepsilon_B)$$

$$= (m+1)(\varepsilon_R\bar{\varepsilon}_R + \varepsilon_B\bar{\varepsilon}_B) . \tag{13}$$

A quantity of interest is the percentage of battlefield sites held by the attacking Blue (hence negative) forces after some time (number of iterations). The total negative site probability can be found from the discrete probability distribution function, $F(n) = \sum_{i=-n}^{-1} D_{n,i}$, where the sum of probabilities is over all possible magnitudes of a site held by Blue forces after $n$ iterations. Using the normal approximation,

$$F(n) \sim \begin{cases} \beta_1^{(-1)} & n = 1 \\ \sum_{x=-n}^{-1} \frac{1}{\sqrt{2\pi\sigma_n}} e^{(-(x-\mu_n)^2/2\sigma_n} & n > 1 \end{cases} . \tag{14}$$

A simpler approximation to $F(n)$ can be had by using the Error function:

$$F(n) \sim \begin{cases} \beta_1^{(-1)} & n = 1 \\ \frac{1}{2}(1 - erf\,\frac{(0.5+n\mu_1)}{\sqrt{2n\sigma_1}}) & n > 1 \end{cases}. \tag{15}$$

Taking the derivative of $F$ with respect to $n$, we can find the local maximum:

$$F'(n) = -\tfrac{1}{2}(-1)^0 \tfrac{2}{\sqrt{\pi}} H_0(z)\, e^{-z^2} dz \qquad z = \tfrac{(0.5+n\mu_1)}{\sqrt{2n\sigma_1}},\ H_0(z) = 1;$$

$$= -\tfrac{1}{\sqrt{\pi}} e^{-z^2} \left[ \frac{2n\sigma_1\mu_1 - (0.5 + n\mu_1)\sigma_1}{(2n\sigma_1)^{3/2}} \right]$$

$$= -\tfrac{1}{\sqrt{\pi}} e^{-z^2} \left[ \frac{(n\mu_1 - 0.5)\sigma_1}{(2n\sigma_1)^{3/2}} \right], \tag{16}$$

where $H_n(z)$ is the Hermite polynomial. $F$ thus has a maximum at $n\mu_1 = 0.5$ only if $\mu_1 > 0 \quad (\varepsilon_R > \varepsilon_B)$, as seen in Fig. 4.

Figs. 5 and 6 show the corresponding results of the Case 1 random model simulation found using an actual random generator. Fig. 5(a) and 6(a) illustrate the actual random distributions, and 5(b) and 6(b) show the theoretical and actual values of $F(n)$.

***3.1.3. Case 2:*** $R = +1, +2;\ B = -1, -2$

As with *Case 1,* at $n = 0$ all sites have initial values $K_0(i, j) = v_{ij}^{(0)} = 0$. Again, the Red force attacks with rate $\varepsilon_R$, and the Blue force with rate $\varepsilon_B$. With this model version, however, the deployed forces will be weighted roughly equally between $+1$ *and* $+2$, or $-1$ *and* $-2$, respectively. (Thus the probability of $+2$ by Red will be approximately $\frac{1}{2}\varepsilon_R$.) The probabilities that a given site does NOT receive Red or Blue deployment remain $(1 - \varepsilon_R) = \bar{\varepsilon}_R$, and $(1 - \varepsilon_B) = \bar{\varepsilon}_B$. When $n = 1$, each site has value $K_1(i, j) = v_{ij}^{(1)} = -2, -1, 0, +1, or +2$. The probability of each $v_{ij}^{(1)}$ for the first step of Case 2 is found from considering the nine possible permutations of $R$ and $B$:

$$p(v_{ij}^{(1)} = -2) = p\,(R \to 0) \times p\,(B \to -2)$$

$$= \bar{\varepsilon}_R(\tfrac{1}{2}\varepsilon_B)\,,$$

$$= \beta_1^{(-2)},$$

$$p(v_{ij}^{(1)} = -1) = p\,(R \to 0) \times p\,(B \to -1) + p(R \to 1) \times p(B \to -2)$$

$$= \bar{\varepsilon}_R(\tfrac{1}{2}\varepsilon_B) + (\tfrac{1}{2}\varepsilon_B)(\tfrac{1}{2}\varepsilon_R) = (1 - \tfrac{1}{2}\varepsilon_R)(\tfrac{1}{2}\varepsilon_B)\,,$$

$$= \beta_1^{(-1)},$$

$$p(v_{ij}^{(1)} = 0) = p\,(R \to 0) \times p\,(B \to 0) + p\,(R \to 1) \times p\,(B \to -1) + p(R \to 2) \times p(B \to -2)$$

$$= \bar{\varepsilon}_R\bar{\varepsilon}_B + \tfrac{1}{2}\varepsilon_R\varepsilon_B$$

$$= \beta_1^{(0)},$$

$$p(v_{ij}^{(1)} = +1) = p\,(R \to 1) \times p\,(B \to 0) + p(R \to 2) \times p(B \to -1)$$

$$= \bar{\varepsilon}_B(\tfrac{1}{2}\varepsilon_R) + (\tfrac{1}{2}\varepsilon_R)(\tfrac{1}{2}\varepsilon_B) = (1 - \tfrac{1}{2}\varepsilon_B)(\tfrac{1}{2}\varepsilon_R)$$

$$= \beta_1^{(+1)}\,,$$

$$p(v_{ij}^{(1)} = +2) = p\,(R \to 2) \times p\,(B \to 0)$$

$$= \bar{\varepsilon}_B\,(\tfrac{1}{2}\varepsilon_R)\,,$$

$$= \beta_1^{(+2)}.$$

The completeness relation is given by $\sum_{v=-2n}^{+2n} \beta_n^{(v)} = 1\,.$ (17)

We can conveniently express the probability density for one step as a vector:

$$D_1 = [\beta_1^{(-2)}\ \beta_1^{(-1)}\ \beta_1^{(0)}\ \beta_1^{(+1)}\ \beta_1^{(+2)}]. \qquad (18)$$

The discrete probability distribution for an arbitrary number of steps can be found by the same methods used for Case 1. We discuss the necessary modifications and the results in the following sections.

***The Distributions***

The discrete distribution values for $n = 2$ are found from inspection to be:

$$D_2 = [\beta_1^{(-2)^2} \quad (2\beta_1^{(-2)}\beta_1^{(-1)}) \quad (2\beta_1^{(-2)}\beta_1^{(0)} + \beta_1^{(-1)^2}) \quad 2(\beta_1^{(-2)}\beta_1^{(+1)} + \beta_1^{(-1)}\beta_1^{(0)}) \ldots \tag{19}$$

$$\ldots\beta_1^{(0)^2} + 2\sum_{i=1,2}\beta_1^{(-i)}\beta_1^{(+i)} \quad 2(\beta_1^{(0)}\beta_1^{(+1)} + \beta_1^{(-1)}\beta_1^{(+2)}) \quad (2\beta_1^{(+2)}\beta_1^{(0)} + \beta_1^{(+1)^2}) \quad (2\beta_1^{(+2)}\beta_1^{(+1)}) \quad \beta_1^{(+2)^2}].$$

To find the values for $D_n$ with the Markov chain method, the one-step transition matrices $P_{(n-1,n)}$ defining the transition probabilities from state $n-1$ to state $n$ will now need to be $(4n-3)\times(4n+1)$. For the $n^{th}$ iteration the corresponding matrix will be:

$$P_{(n-1,n)} = \begin{array}{c|ccccccc|} & -2n & (-2n+1) & (-2n+2) & \ldots & \ldots & \ldots & +2n \\ \hline -2n+2 & \beta_1^{(-2)} & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & \beta_1^{(+2)} & 0 & 0 \\ -2n+3 & 0 & \beta_1^{(-2)} & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & \beta_1^{(+2)} & 0 \\ \ddots & . & . & . & . & . & . & 0 \\ \ddots & . & . & . & . & . & . & 0 \\ 2n-2 & 0 & 0 & \beta_1^{(-2)} & \beta_1^{(-1)} & \beta_1^{(0)} & \beta_1^{(+1)} & \beta_1^{(+2)} \end{array}. \tag{20}$$

The probability density function of site values for the $n^{th}$ iteration can be found as before from matrix multiplication, or the inner product, of the single-step matrices:

$$D_n = P_{(0,1)} \cdot P_{(1,2)} \cdot P_{(2,3)} \ldots \cdot P_{(n-1,n)}, \tag{6}$$

yielding the desired $1\times(4n+1)$ vector containing the probabilities:

$$D_n = [\beta_n^{(-2n)} \quad \beta_n^{(-2n+1)} \quad \ldots \quad \ldots \quad \beta_n^{(2n-1)} \quad \beta_n^{(2n)}].$$

It is again possible to find the $n^{th}$ distribution algebraically using the completeness relation in equation (1). Taking $(\beta_1^{(-2)} + \beta_1^{(-1)} + \beta_1^{(0)} + \beta_1^{(+1)} + \beta_1^{(+2)})^n$ and collecting the terms appropriately, one obtains a summation of the distribution values.

The normal distribution approximation of $D_n$, $f_n(x) = \frac{1}{\sqrt{2\pi\sigma_n}} e^{(-(x-\mu_n)^2/2\sigma_n)}$, will now have an average or mean value $\mu_n = \frac{3}{2} n(\varepsilon_R - \varepsilon_B)$ and variance $\sigma_n = [\frac{5}{2} n\varepsilon_B (1 - \frac{9}{10}\varepsilon_B) + \frac{5}{2} n\varepsilon_R (1 - \frac{9}{10}\varepsilon_R)]$. Fig. 7 illustrates this result. The total negative site probability can be found from the probability distribution function, $F(n) = \sum_{i=-2n}^{-1} D_{n,i}$, where the sum is over all possible magnitudes of a site held by Blue forces after $n$ iterations. Using the normal approximation,

$$F(n) \sim \begin{cases} \beta_1^{(-2)} + \beta_1^{(-1)} & n = 1 \\ \sum_{x=-2n}^{-1} \frac{1}{\sqrt{2\pi\sigma_n}} e^{(-(x-\mu_n)^2/2\sigma_n)} & n > 1 \end{cases}. \tag{21}$$

As with Case 1, simpler approximations to $F_n(n)$ can be obtained using the Error function:

$$F(n) \sim \begin{cases} \beta_1^{(-2)} + \beta_1^{(-1)} & n = 1 \\ \frac{1}{2}(1 - erf\, \frac{(0.5 + n\mu_1)}{\sqrt{2n\sigma_1}}) & n > 1 \end{cases}. \tag{22}$$

The results of equations (21) and (22) are shown in Fig. 7(b).

Fig. 8 illustrates the corresponding results of the Case 2 random model simulation found using an actual random generator. Fig. 8(a) illustrates the actual random distribution for $n = 20$ iterations, and 8(b) shows the theoretical and actual values of $F(n)$.

### *3.1.4. Case 3: General random approach*

Extension of the random battle model to larger strength numbers and/or different strength value partitions can be made along the following lines. We begin by allowing Blue attack forces having strength values of $B = \{(-1, w_{-1}), (-2, w_{-2}), (-3, w_{-3}), ...(-a, w_{-a})\}$ (where $w_m$ is a weight factor for $m$ having a value $0 \le w_m \le 1$) deploy at the attack sites, and Red forces with values $R = \{(+1, w_{+1}), (+2, w_{+2}), (+3, w_{+3}), ...(b, w_{+b})\}$ counter at the randomly chosen defense sites. The initial probability vector $P_1 = [\beta_1^{(-a)} \cdots \beta_1^{(-1)}\, \beta_1^{(0)}\, \beta_1^{(+1)} \cdots \beta_1^{(+b)}]$ can be defined in the usual manner. For example,

$$p(v_{ij}^{(1)} = -a) = p\,(R \to 0) \times p\,(B \to -a) = (1 - \varepsilon_R)(w_{-a}\varepsilon_B) = \beta_1^{(-a)} . \tag{23}$$

The completeness relation becomes $\sum_{v=-an}^{+bn} \beta_n^{(v)} = 1$. Transition matrices $P_{(n-1,n)}$ will have $(a+b)(n-1)+1$ rows and $(a+b)n+1$ columns. For the Gaussian approximation, the average and variance can be found from:

$$\mu_1 = P_{(0,1)} \cdot M_{-a,b} \,, \;\; \mu_n = n\mu_1 \,, \tag{24}$$

$$\sigma_1 = \; P_{(0,1)} \cdot [(M_{-a,b} - \mu_1)^2] \,, \;\; \sigma_n = n\sigma_1 \,. \tag{25}$$

### *3.2. Strategic Battles: The Model*

In this section we consider strategic battles between well matched parties. We focus our attention on battles where the two sides possess the same fire power and troop reserves. Given the characteristic distinctions between the two sides, as mentioned in Sec. 2, we prefer to call the sides Attackers and Insurgents instead of say Red and Blue. As before, individual Attackers and Insurgents are assigned numeric strengths. The rule based deployment strategies result in occupation numbers being assigned to each of the lattice sites at any given time step. Casualties result when opposing forces occupy the same site. Combatants do not actually move on the battlefield, but rather occupy their given sites, engage with the adversaries by drawing from the reserve troops, and provide field information, which may be used for determining further deployments.

Battles continue until one side has exhausted the reserve forces. If both sides have essentially deployed all forces to the conflict, i.e., have no more troop reserves left, at the same time, the battle is a draw. The conflict also ceases if the aggressor is completely eradicated from the field. The method of the study [1, 20] is described below.

***3.2.1. Model Details***

We use an $L \times L$, $L = 50$ lattice, here as before. We have studied larger systems and found that enlarging the system does not affect our results. Each side is typically allotted an initial reserve of $500{,}000$ resource points or "troops." Changing the troop size may end up influencing the time span across which the battle wages. It is hence typical to keep the troop reserves sufficiently large. In our studies, we have used smaller as well as much larger troop reserves. The sizes of the troop reserves are set to be sufficiently large such that the outcome is not dependent on the details of the number.

The Insurgents begin by occupying the lattice at random sites with resources typically having a mix of strength values such as $d = +1, +2, +3 ... d_{\max}$, at occupation density, $D$ (the percentage of sites in the $50 \times 50$ matrix), so that each $K(i, j) \to 0, +1, +2, +3$, etc. One can imagine $d$ being the equivalent of force levels used in these strategic battles. A precise connection between what level of force $d = +1$ or $d = +2$ and so on might mean in the context of firepower in a real battle is only qualitative.

The battle begins with the Attackers deploying at random locations on the battlefield with an attack density, $A$, and strength values of $a = -1, -2, -3 ... -a_{\max}$, where $a_{\max}$ is the maximum value allowed for $a$. The simpler case using values of $d = +1$ and $a = -1$ for all deployments will be addressed first. This case can be thought of as one where the parties have limitations in the kinds of weapons available. The case with $d = +1, +2$, and $a = -1, -2$, yields a richer battle and is hence addressed in more detail. This latter study can be thought of as a battle between parties with more advanced weapons. The site values are summed with each iteration, resulting in a one-to-one casualty ratio. Sites with values of zero are neutral. During every deployment, the strength deployed is subtracted from the respective resource base. We have conducted preliminary analyses of battles where $a$ and $d$ values can vary in magnitude between $1$ and $3$. While the details are affected when compared to the battles with $a$ and $d$ values ranging between $1$ and $2$, the broad conclusions are not. These models and models with unequal magnitudes of $a$ and $d$, will be addressed elsewhere.

Following the initial attack, some fraction of the sites will have negative values (i.e., are Attacker held). On subsequent iterations, the Attackers and the Insurgents deploy additional resources from their reserves via the algorithm described below. The occupation density is the value used to determine how many sites will get resources applied to them in a given iteration. If the Insurgent occupation density $D = 50\%$, then $\approx 1250$ sites (on a 2500 site lattice) will get values of $+1, +2$, etc. applied to them. This step is independent of how many sites the Insurgents already hold. While $D$ remains constant, the total number of sites with positive values in the matrix will vary depending on all the other parameters. Observe also that $A$ is only used in the initial attack, after which deployments are always made from the sites already captured by the Attackers. Thus $A$ necessarily means $A(t = 0)$.

The Attacker tries to maximize the effect of its attack by always deploying resources in the direction with the greatest density of Insurgents pursuant to the available information. A neighbor list is defined by an $n \times m$ matrix about each attack site, $K_A(i, j)$, where $i$ and $j$ are indices for row and column, respectively. The size of the neighbor list is dictated by a range variable $r$, which refers to the percent of the field size along $i$ or $j$ for which information is available at site $K_A(i, j)$ [21]. For an $L \times L$ matrix, we have $[\frac{1}{L} \le r \le \frac{1}{2}] \times 100\%$ where the $r$ values chosen in our *outcome diagrams* (in analogy with phase diagrams) were in increments of $4\%$. The Attacker receives information from a neighbor list with maximum size $(2r+1) \times (2r+1)$ [18]. The occupation or resource values of all sites in each quadrant of the neighbor list at $K_A(i, j)$ are summed.

If the computed sum is largest in, say, the first quadrant, then site

$$K(i-1, j+1) \to K(i-1, j+1) - a_{\max}, \qquad (26)$$

if this site is still within the battlefield, and so on. Eq. (26) hence describes deployment along the four diagonals. If adjacent quadrants, for example quadrants I and IV, have equal values, then we let

$$K(i, j+1) \to K(i, j+1) - a_{\max}. \qquad (27)$$

Eq. (27) hence provides for deployments to up, down, left, and right. If opposing quadrants (i.e., I and III or II and IV) or if three or all four quadrants of the neighbor list have the same value, then there is no new deployment from that site during that particular iteration. However, this case is rarely realized. The rationale for using maximum strength values for the Attackers is based on their strategic need to maximize effectiveness. The battlefield is not updated until all such decisions have been made. It may be noted that in our model, the deployments at any given time step are made by the Attacker to some neighbor site only according to the specifications given in Eqs. (26) and (27) above. When $r$ is large, this deployment is based on the averaged information over each quadrant.

The Insurgents then again randomly deploy reinforcements to the battlefield, with values $d = +1, +2, +3 ... d_{\max}$ at the occupation density $D$, such that for any reinforced site

$$K(i, j) \to K(i, j) + d. \qquad (28)$$

### *3.2.2. Results - Battles with* $d = +1, a = -1$

Here we present results based on our model studies. We start by considering the simplest strategic battles in which $d = +1$ and $a = -1$. The limited availability of fire power for each side means that more troops must be present on the lattice sites to bring about a win. Calculations carried out with $A$ values ranging from $5\%$ to $100\%$ of the field in increments of $5\%$ and with $r$ varying from $2\%$ to $48\%$ in $4\%$ increments for a total of 240 battle cases are summarized for two different values of $D$ in Fig. 9. The color coding in Fig. 9 is determined as follows: the percentage of the ratio of the difference between the total number of Insurgent reserves and the total number of Attacker reserves at the end of the battle to the original number of Insurgents or Attackers determines the numbers on the vertical color bar. Thus, $-80$ implies $80\%$ of the Attacker reserves remained intact in a battle in which the Insurgents were left with no reserves. The results suggest that when the attack density $A$ is sufficiently low, it is unlikely that the Attackers will win when the attacks are targeted and highly local (i.e., when $r$ is sufficiently small). Presence of more Insurgents on the battle field (i.e., larger $D$) only makes it harder for the Attackers to win unless their numbers are sufficiently large, as can be seen from Fig. 9(b).

For the case where $d = +1$ and $a = -1$ only, the probability that the Attackers may be completely eradicated from the lattice is greatly increased, especially at low values of $A$ and $r$. This occurs due to less efficient deployments, which may fail to wear down the resistance or to establish and reinforce an adequate foothold on a territory.

For comparable values of $A$ and $D$ (in the range of $40\%$ to $60\%$ ), an Attacker victory becomes more likely. The Attacker is able to establish control over some territory and subsequently deploys resources more *slowly* than do the Insurgents. The net effect is to wear down the defense reserves. For $A > D$, the Attacker will establish control over large territory, but will subsequently deploy resources too rapidly, resulting in a likely draw or defense win. Fig. 9 shows outcome diagrams for this case for parameter values similar to those used later in Fig. 11.

***3.2.3. Results - Battles with*** $d = +1, +2, a = -1, -2$

More interesting and perhaps more realistic battles result when we set $d = +1, +2,$ and $a = -1, -2$. The increased force levels allow for more aggressive attacks with smaller troop levels. The outcome diagrams are hence different for this case compared to the $d = +1, a = -1$ cases. As we shall see, our studies suggest that small enough $r$ and low values of $A$ are the most important for the Attacker to win these battles. Hence, this case is distinct from the $d = +1, a = -1$ battles where values of $A$ and $D$ must be comparable. Hence the $A$ values needed for the Attackers to win had to be much larger than what is typical in the present case.

In a battle with "medium range" intelligence (i.e., $r \approx 24\%$), the Attacker deploys resources to one of its eight neighbor sites in each of the quadrants according to the recipe in Eqs. (1) and (2). So the determination of which neighbor site will get the deployment is based upon a neighbor list of sites from within a square of dimensions $(2r+1)\times(2r+1)$ with the site from where the deployment is to be made being at the center of the square. The intelligence information allows for the coordination of the attack from differing directions. An information-driven pattern of attack site clusters emerges with cluster dimensions $\sim r$, indicating that most newly deployed forces remain within a localized area securing a region of sites within that area. Sites held by the Attackers which share the same information act so as to reinforce each other. Fig. 10 illustrates the results of a battle with $D = 40\%, A = 10\%, r = 26\%,$ which resulted in a draw.

At large range (i.e., as $r$ approaches $48\%$ in these studies) the local deployments are influenced by information from distant sites, resulting in a mix of clusters and "strings" on the battlefield that show regions that are occupied by the Attacker. Overlapping and competing information results in more random and less efficient deployments. Since the neighbor list includes more sites toward the center of the field, the Attackers will tend to move away from the edges of the field, which in turn will be dominated by the Insurgents. The Attackers will spread resources far from the original attack sites, which may (being less strongly guarded) subsequently be recaptured by the Insurgents. Despite having an apparent lack of strategy, the Insurgents may gain the advantage and actually win the battle, even at very low attack densities. Our calculations reveal that the sophisticated technology implied by larger communication distances has limited value if too-distant data is used in the decision making that enters into the local moves.

The outcomes in terms of Insurgent win, Attacker win, or draw given the same initial parameters change with increasing range, $r$, and with increasing attack density, $A$. Fig. 11 shows outcome diagrams for the model, with several different values of $D$. Each diagram represents the results of testing the model once for each input data set, with $A$ varying from $5\%$ to $100\%$ of the field in increments of $5\%$, and $r$ varying from $2\%$ to $48\%$ in $4\%$ increments, for a total of $240$ battles. In these figures, the three lightest shades represent values for which the outcome varies with the random generator between wins for the Insurgents, the Attackers, and draw. Darker regions represent values for which the win goes consistently to the Attackers.

## 4. Summary and Conclusions

In this work we investigate several processes which could be described as battle-like. Motivated by recent attempts to describe such complex adaptive systems, we wanted to establish a statistical framework for the simple case of a completely random battle, and to propose an adaptive version. We have focused on battles between well-matched adversaries, hence all the cases considered are highly contentious. Here a strategy has been presented which introduces a variable parameter – namely the information available from a neighborhood of sites/locations - which in turn is utilized to make interactive decisions. We have shown that such decisions may significantly affect the process outcome.

The completely random striking of a battlefield or lattice by two non-interacting opponents can be completely described in terms of a Markov type process, defined as a process which is *not* history dependent. Expected values at each field or lattice location are based solely on the previous value and the probabilities of each opponent striking or not striking at that location. Theoretical apparatus is developed to express distributions of probable site magnitudes for the simple case of $\pm 1$ deployments, and for the more complex case of $\pm 1, \pm 2$ deployments. Several approaches are described – graphical, algebraic, Markov process, and a Gaussian approximation, which can be utilized to predict resulting matrix or lattice values. Extension to more general and asymmetric deployment values is straightforward.

The discrete distribution of site values is found to be approximately Gaussian. Long term progress of the battle is determined by the differential between the deployment percentages, $\varepsilon_R$ and $\varepsilon_B$, of the two opponents with eventual elimination of the weaker opponent from all sites. The opponent deploying more resources will gain territorial control unless they exhaust their resources first. As seen in Figs. 5-8, values produced by a random generator compare well with theoretical approximations.

The strategic land battle model allows an intelligent attacker to obtain knowledge of the presence of forces on neighboring sites, and to deploy resources according to a predetermined plan. As the insurgent continues to deploy randomly (unpredictably), the attacker must interact with and adapt to the ever-changing environment.

As seen in Figs. 9 and 11, the availability and accuracy of this neighborhood information can decisively affect the ability of the attacker to deploy efficiently and not over-expose themselves on the field. If the attack density is low and the information is localized (highly accurate) the attacker tends to remain localized, conserving resources and building strongly fortified areas. Note particularly the case shown in Fig. 11, with $D = 50\%$. In a completely random battle, if $A > 50\%$, the attacker would be expected to deploy more resources and thus exhaust their reserves first. However with a strategic approach and very short range intelligence $(\sim 2\%)$ the Attacker deploys fewer resources reinforcing very small areas and thus a draw or even an attack win may result. This effect becomes even more pronounced at $D = 60\%$. Contrast this with Fig. 9, where for $D = 60\%$ and deployments of $+/-1$ only, a low range may result in complete eradication of the attacker from the matrix - being too conservative in gaining new territory subjects the attacker to higher probability of simply being overrun before gaining an adequate foothold. As the available information or intelligence becomes more diffuse, the attacker will spread to more sites (reflecting the uncertainty in locating strongly enemy dominated regions) and will thus deploy more and more reserves. The increase in exposure is thus balanced against how strongly or weakly the territory is held. A large attacker presence at a particular location withstands the insurgent assaults – a small one does not.

The interplay of random and directed actions in the strategic model leads to a variety of behaviors under different parameter values, thus we believe the model holds potential to find application with a variety of processes from the study of modern land battles to the spread or containment of disease in the body, etc.  Further studies are under way to address more complex algorithms for battle actions, including asymmetric battles and network approaches.

*Acknowledgements*:

We are grateful to the ARO for research support. M. Ramanathan, R.P. Simion and S.D. Sun are acknowledged for their interest in this work.

*Bibliography:*


[1] J. von Neumann in: Theory of Self Reproducing Automata, A.W. Burks ed., Univ. Illinois, Urbana, 1966; S. Ulam, Annu. Rev. Bio. 12 (1974) 255; S. Wolfram, Revs. Mod. Phys. 55 (1983) 601.

[2] For early work on interesting collective behavior of simple objects on a network and evolving in rule based fashion, see S.A. Kauffman, J. Theor. Biol. 22 (1969) 437.

[3] See e.g. N.H. Gregersen ed., From Complexity to Life, Oxford, Oxford, 2003.

[4] B. Kozma, A. Barrat, Phys. Rev. E 77 (2008) 016102.

[5] C. Kamp, S. Bornholdt, Phys. Rev. Lett. 88 (2002) 068104; S. Bonhoeffer, P. Sniegowski, Nature 420 (2002) 367.

[6] See e.g., M. Mattia, P. Del Guidice, Neural Comput. 12 (2000) 2305; J. Reutimann, M. Giogliano, S. Fusi, *ibid*. 15 (2003) 811; R. Savit, R. Manuca, R. Riolo, Phys. Rev. Lett. 82 (1999) 2203; K. Jain, Phys. Rev. E 76 (2007) 031922; I. Bena, M. Droz, J. Szwabinski, A. Pekalski, *ibid* 76 (2007) 011908; M. Hall, K. Christensen, A.A. di Collobiano, H.J. Jensen, *ibid* 66 (2002) 011904; and M. Marsili, R. Mulet, F. Ricci-Tersenghi, R. Zechhina, Phys. Rev. Lett. 87 (2001) 208701.

[7] Predator-prey models have been studied for many years, see for instance, N. Boccara, O. Roblin, M. Roger, Phys. Rev. E 50 (1994) 4531.

[8] Rule based evolution of damage has been studied for example in H. Hinrichsen, J.S. Weitz, E. Domany, J. Stat. Phys. 88 (1997) 617.

[9] L.D. Miller, Jr., M.F. Sulcoski, B.A. Farmer, A.I.P. Conf. Proc. 411 (1997) 187.

[10] J.T. Dockery, A.E.R. Woodcock, eds., The Military Landscape: Mathematical Models of Combat, Woodhead, Cambridge, 1993.

[11] M. Gell-Mann, in D.S. Alberts, T.J. Czerwinski, Eds., Complexity, Global Politics and National Security, National Defense University, Washington, 1997, p. 45.

[12] C.J. Ancker, Nav. Res. Logistics 42 (1995) 311.

[13] N.K. Jaiswal, B.S. Nagabhushana in J. Bracken, M. Kress, R.E. Rosenthal Eds., Warfare Modeling, Wiley, New York, 1975.

[14] L.R. Speight, D. Rowland, Mil. Opns. Res. 4 (1999) 45.

[15] T.D. Lee, C.N. Yang, Phys. Rev. 87 (1952) 410.

[16] See for example G.M. Wysin, Am. J. Phys. 61 (1993) 734.

[17] The idea of using a neighbor list is inspired by its usage in molecular dynamics simulations of many body systems, see for example in M.P. Allen, D.J. Tildesley, Computer Simulations of Liquids, Clarendon, Oxford, 1987.

[18] A. B. Clarke, R. L. Disney, Probability and Random Processes for Engineers and Scientists John Wiley & Sons Inc., New York, 1970, pp. 217-237.

[19] G. Chartrand, Introductory Graph Theory, Dover, NY, 1984.

[20] J.H. Miller, S.E Page, Complex Adaptive Systems: An Introduction to Computational Models of Social Life, Princeton University Press, Princeton and Oxford, 2007.

[21] For an introduction to how intelligence is gathered see for example, P.C. Hew, Mil. Opns. Res. 10 (2005) 17.

*Figure captions:*

Fig. 1: Weighted, directed graph for Markovian battle, Case 1: First step. Any path must contain only one arrow. Set of directed edges, $A = \{(0,-1), (0,0), (0,+1)\}$

Fig. 2: Weighted, directed graph for Markovian battle, Case 1: Step two ( $n = 2$ ). Available paths must contain 2 steps. Arrows beginning and ending on the same value can be traversed as many times as needed. $A = \{(-1,-2), (-1,-1), (-1,0), (0,-1), (0,0), (0,1), (1,0), (1,1), (1,2)\}$

Fig. 3: Probability distribution for Markovian battle, Case 1 with $\varepsilon_R = 0.6, \varepsilon_B = 0.5, n = 20.$ Gaussian approximation shown as solid line using, $\mu_n = n(\varepsilon_R - \varepsilon_B) = 2$ , $\sigma = [n(\varepsilon_B \bar{\varepsilon}_B + \varepsilon_R \bar{\varepsilon}_R)] = 9.8$ .

Fig. 4: Probability of Blue occupation (negative sites) for Markov dependent Case 1 process with $\varepsilon_R = 0.6, \varepsilon_B = 0.5, n = 20$. Fig. 4 (a): Probability of Blue occupation vs. iteration. 4 (b): Probability vs. average site value. * = actual, solid line = approximation using Eq. (15).

Fig. 5 (a): Case 1 Distribution of site values from random number generator. $\varepsilon_R = 0.6, \varepsilon_B = 0.5, n = 60$. 5 (b): Probability of negative occupation vs. iteration – theoretical (dashed) and actual.
[ $\mu_n = 6$ , $\sigma = 29.4$ ].

Fig. 6 (a): Case 1 Distribution of site values from random generator. $\varepsilon_R = 0.2, \varepsilon_B = 0.4, n = 20.$ 6 (b): Probability of negative occupation vs. iteration – theoretical (dashed) and actual [ $\mu_n = -4$ , $\sigma = 8.0$ ].

Fig. 7 (a): Case 2 probability distribution with $\varepsilon_R = 0.6, \varepsilon_B = 0.5, n = 20.$ $\mu_n = \frac{3}{2} n(\varepsilon_R - \varepsilon_B) = 3$ , $\sigma_n = [\frac{5}{2} n\varepsilon_B (1 - \frac{9}{10}\varepsilon_B) + \frac{5}{2} n\varepsilon_R (1 - \frac{9}{10}\varepsilon_R)]$=27.55. 7 (b): Probability of Blue occupation. Markov values – starred; and Gaussian approximations - solid.

Fig. 8 (a): Case 2 Distribution of site values from random number generator. $\varepsilon_R = 0.6, \varepsilon_B = 0.5, n = 60$. 8 (b): Probability of negative occupation vs. iteration – theoretical (dashed) and actual.

Fig. 9: Outcome diagrams for battles using strengths of $d = +1$ and $a = -1$. $A$ vs $r$ values are shown for fixed $D$. Dark grey areas (lower left) in right panel indicate complete eradication of Attackers from the lattice. Lightest areas in left panel are Insurgent victories.

Fig. 10: Cluster formation in a medium range battle. The strengths of resources at each site using a grey scale.

Fig. 11: Outcome diagram plots for battles with a wide range of $A$ and $r$ values are shown for fixed $D$ values with $d = +1,+2$ and $a = -1,-2$. The scale depicts the resource left at the end of

the battle. The three light shades are areas where the outcome varies depending upon the random number generator, i.e., the outcome could go either way. The darker shades describe attack wins. The results suggest that low values of $A$ and $r$ with sufficiently large values of $D$ favors attacker wins.

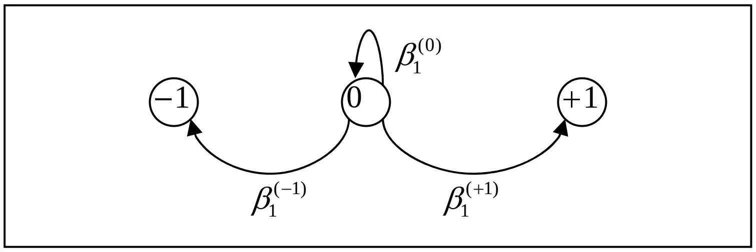


Figure 1

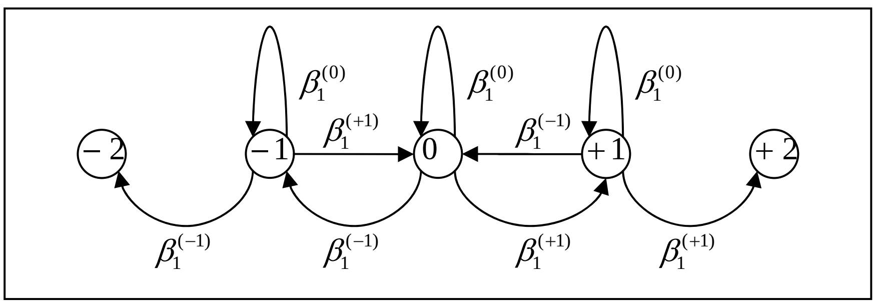


Figure 2

(Shanahan)

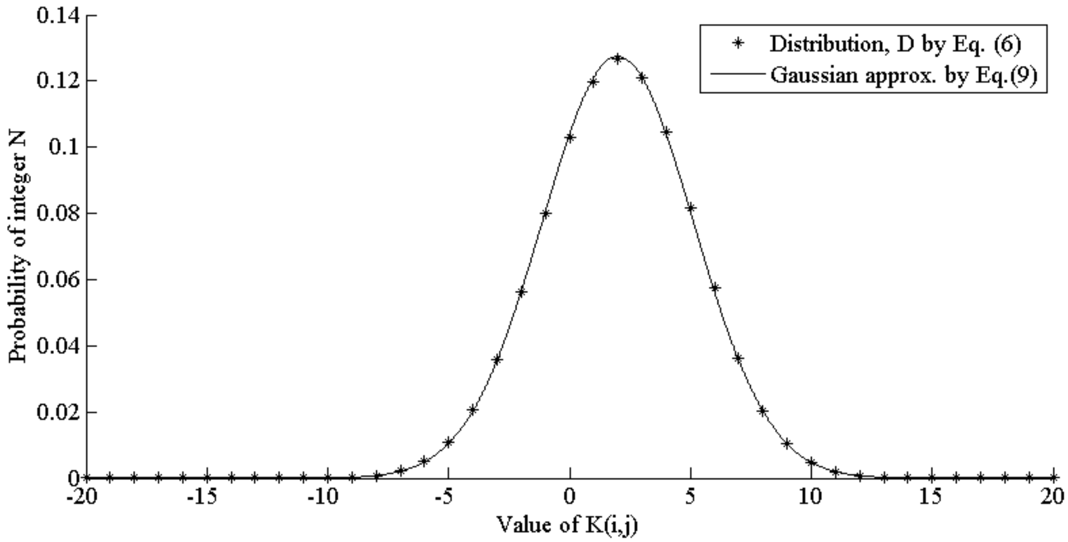


Figure 3

(Shanahan)

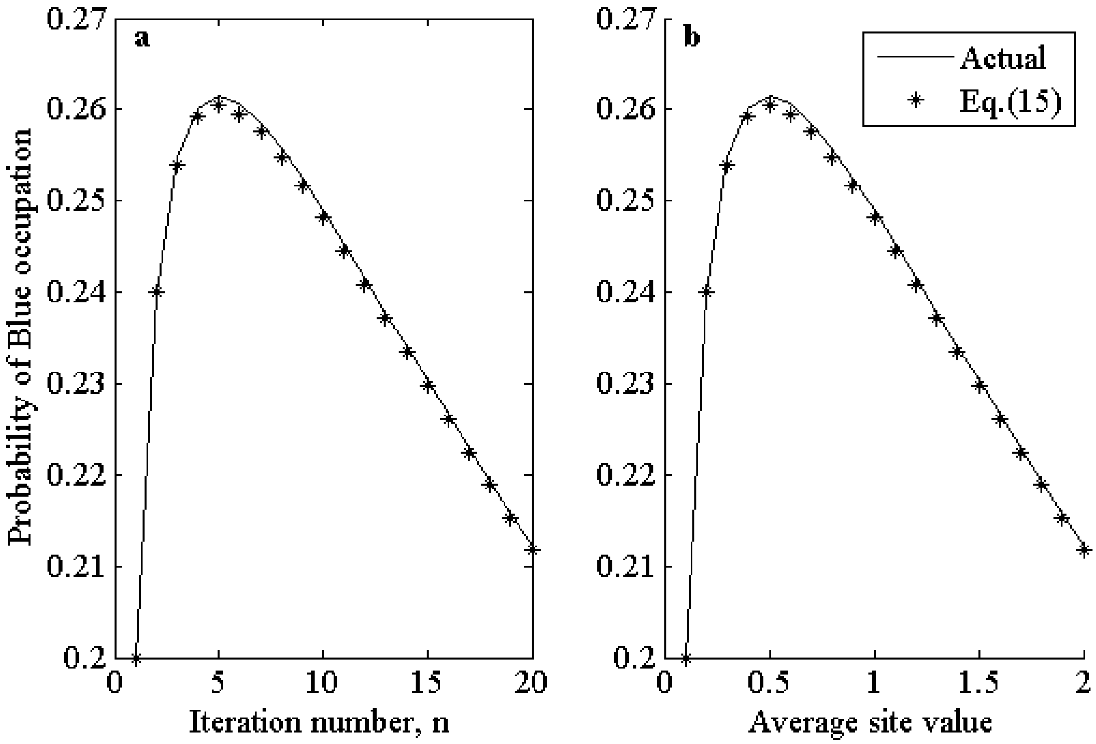


Figure 4

(Shanahan)

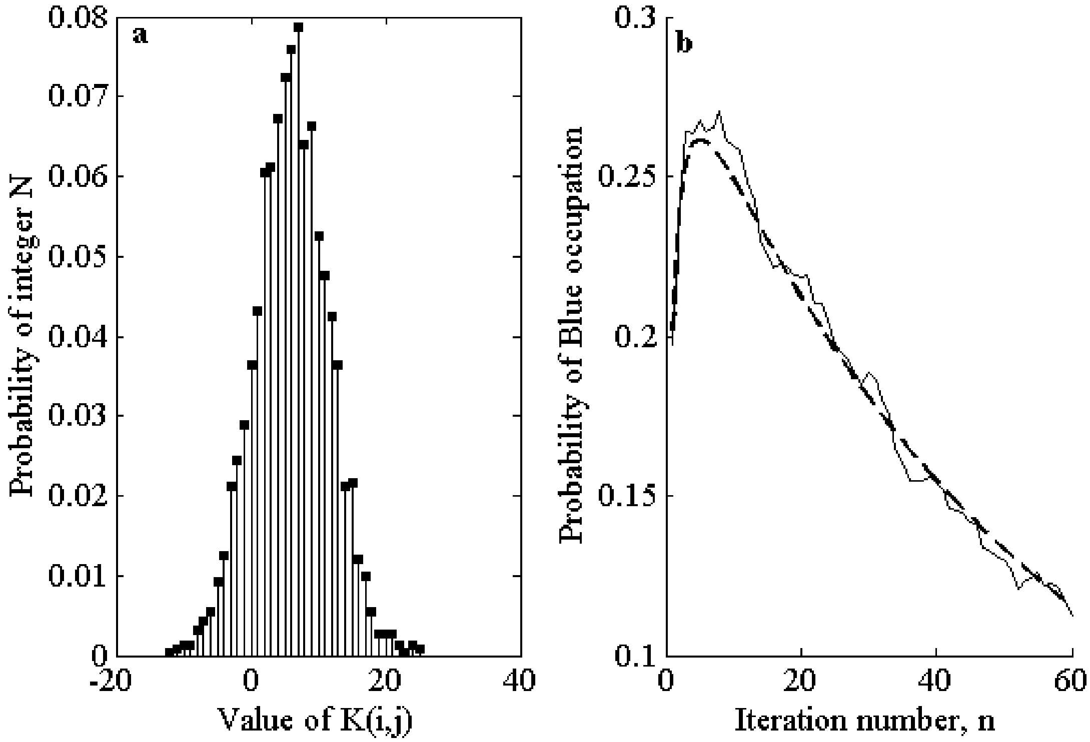


Figure 5

(Shanahan)

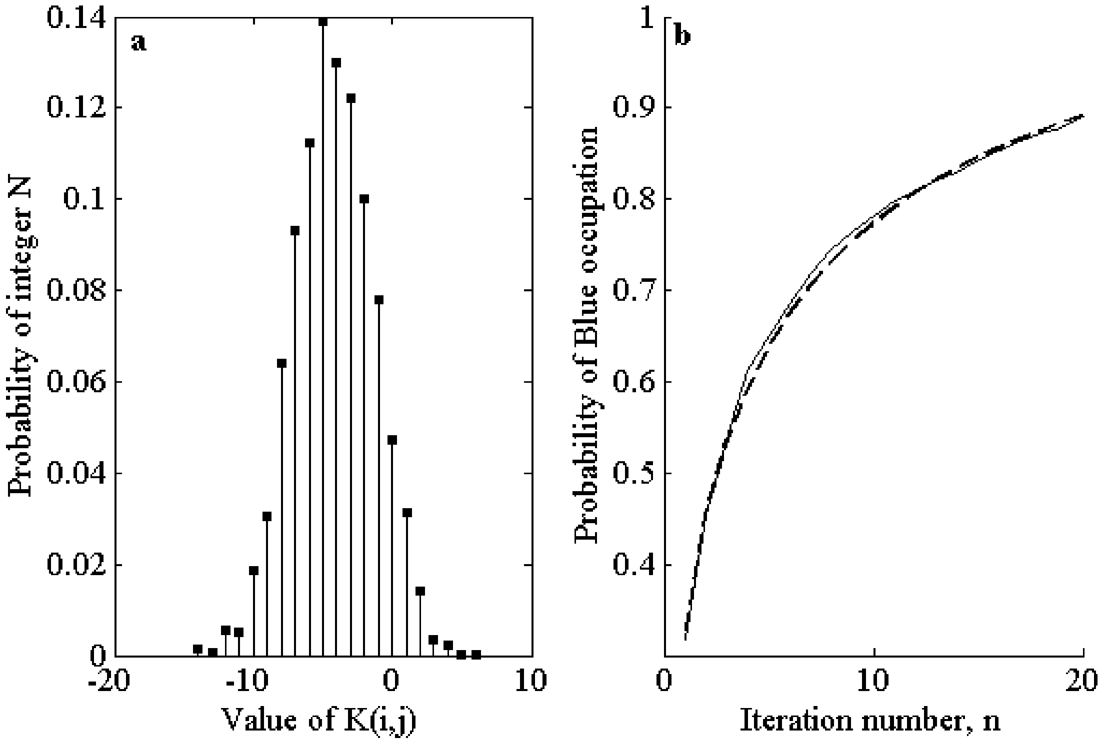


Figure 6

(Shanahan)

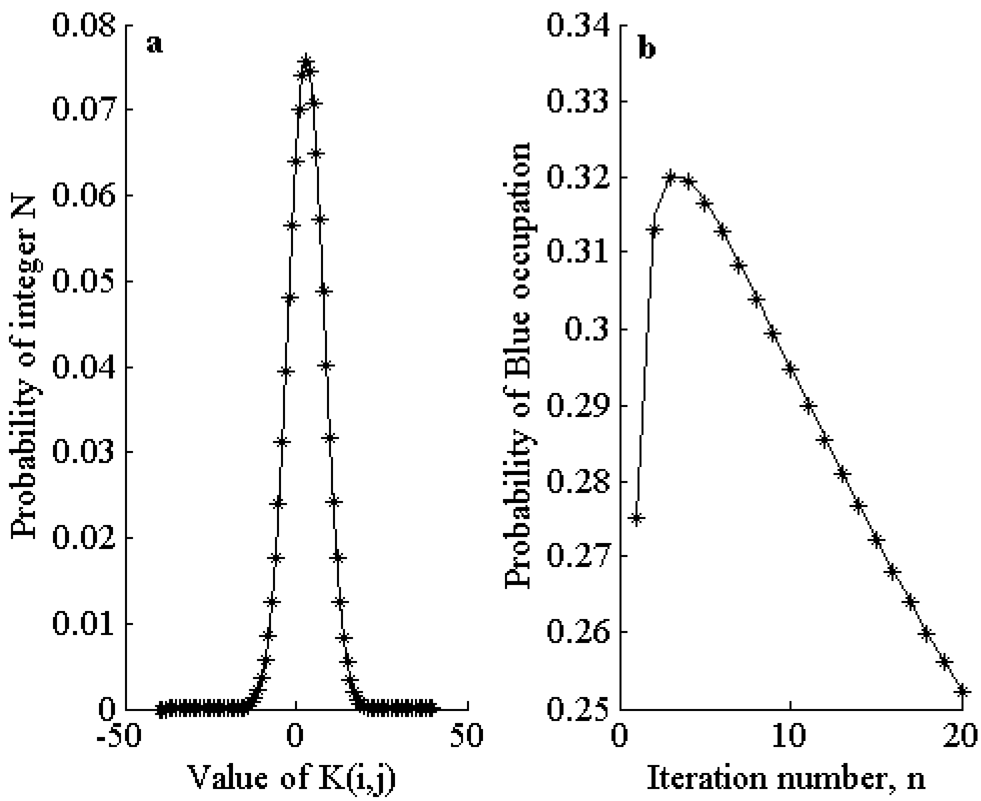


Figure 7

(Shanahan)

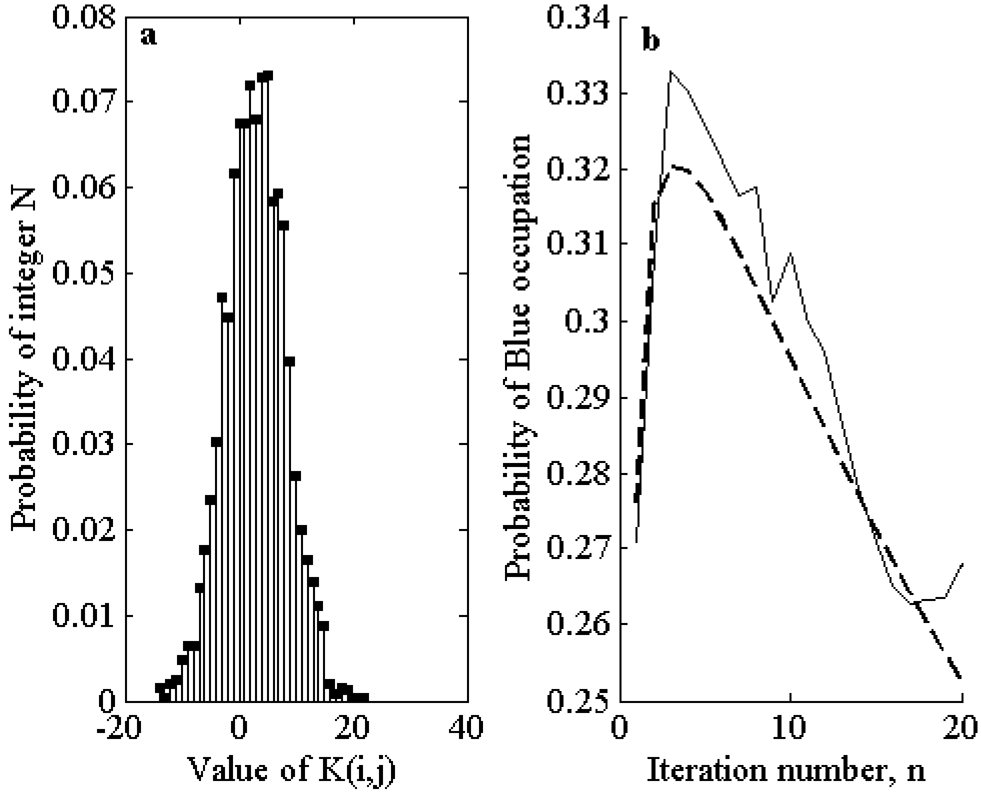


Figure 8

(Shanahan)

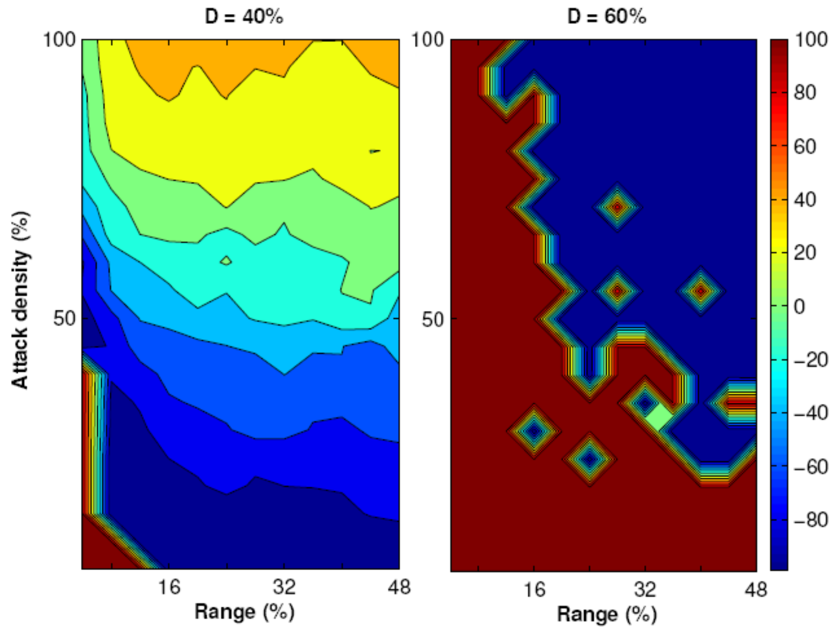


Figure 9

(Shanahan)

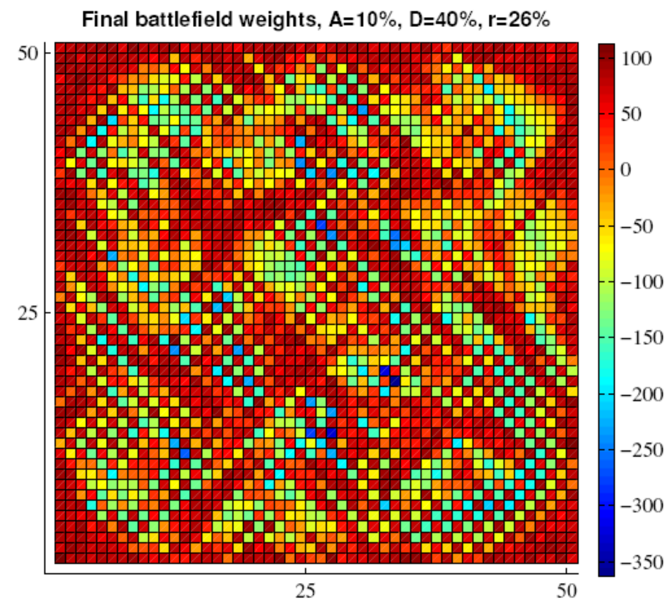


Figure 10

(Shanahan)

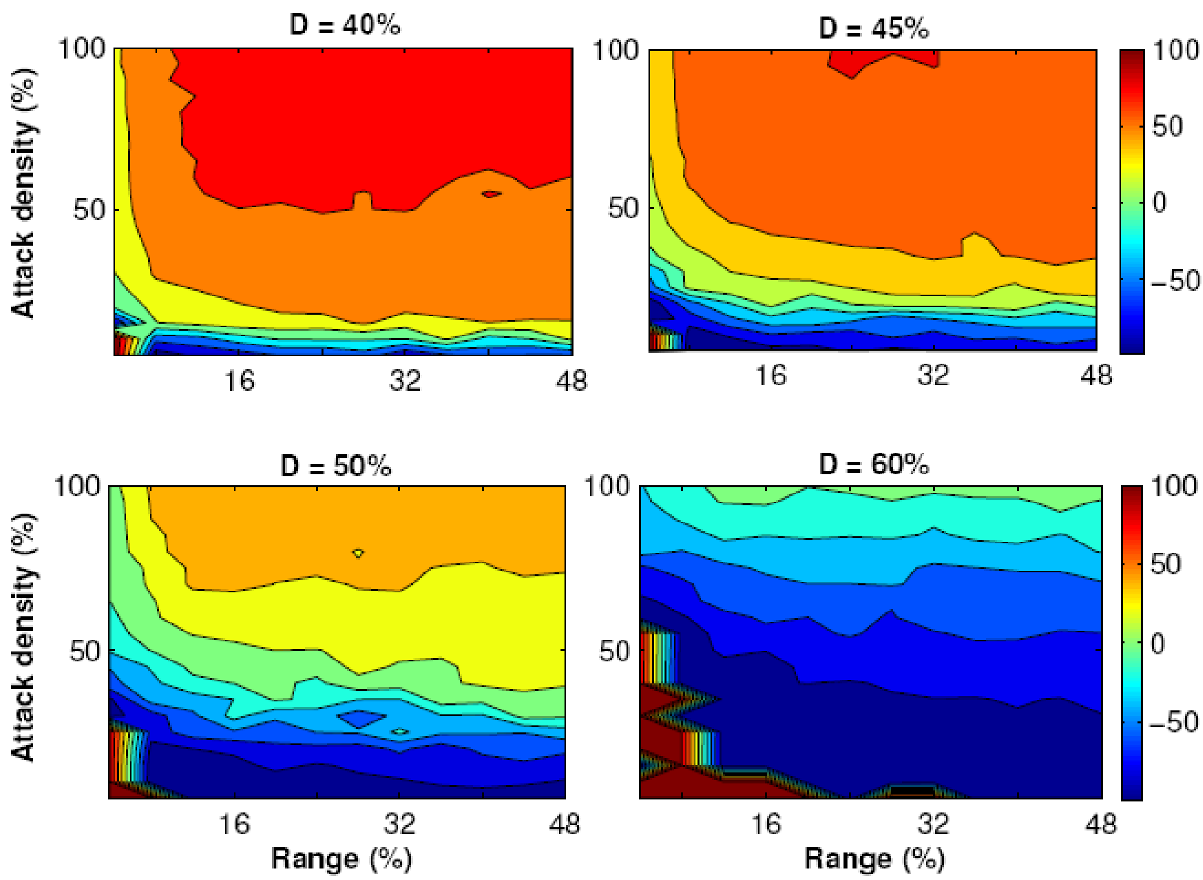


Figure 11

(Shanahan)