Advances in Complex Systems

Organizational Strategic Adaptation in the Presence of Inertia

Anthony Brabazon
School of Business, University College Dublin,
Belfield, Dublin 4, Ireland
anthony.brabazon@ucd.ie

Arlindo Silva
Escola Superior de Tecnologia,
Instituto Politecnico de Castelo Branco,
Av. do Empresario, 6000 Castelo Branco, Portugal
arlindo@est.ipcb.pt

Tiago Ferra de Sousa
Escola Superior de Tecnologia,
Instituto Politecnico de Castelo Branco,
Av. do Empresario, 6000 Castelo Branco, Portugal

Michael O'Neill
School of Computer Science and Informatics, University College Dublin,
Belfield, Dublin 4, Ireland

Robin Matthews
Center for International Business Policy,
Kingston University, London
United Kingdom
r.matthews@kingston.ac.uk

Ernesto Costa
Centro de Informatica e Sistemas da Universidade de Coimbra,
Polo II-Pinhal de Marrocos,
3030 Coimbra, Portugal
ernesto@dei.uc.pt

Received (5 December 2003)
Revised (17 November 2004)
Revised (14 October 2005)

This paper extends the particle swarm metaphor into the domain of organization science. A simulator (*OrgSwarm*) which can be used to model the adaptation of a population of organizations on a strategic landscape is introduced. The simulator embeds a number of features of the process of organizational adaptation, including the resistance of organizations to change (strategic inertia), errorful assessments of the payoffs to proposed strategies, and market competition. These features allow the examination of a wide range of real-life scenarios in organizational adaptation. The paper reports the results of a number of simulation experiments and these suggest that agent (management) uncertainty as to the payoffs to potential strategies has the effect of lowering the average payoffs obtained by a population of organizations. The results also suggest that a degree of strategic inertia can assist rather than hamper adaptive efforts at a populational level.

1. Introduction

In an organizational setting a strategy consists of a choice of what activities an organization will perform, and related choices as to how these activities will be performed [1]. These choices define the strategic configuration of an organization. Recent work by [2] and [3] has recognized that strategic configurations consist of interlinked individual elements (decisions) and have applied general models of interconnected systems such as Kauffman's NK model to examine the implications of this for processes of organizational adaptation.

Following a long-established metaphor of adaptation as search on a landscape [4, 5], strategic adaptation can be metaphorically considered as an attempt by strategists to uncover peaks on a high-dimensional strategy landscape. Some strategic configurations produce high payoffs (profits), others produce poor results. The search for good strategic configurations is difficult due to the vast number of possible strategic configurations, uncertainty as to the nature of topology of the strategic landscape, and the dynamic nature of this landscape. Despite these problems the search process for good strategies is not hopeless. Decision-makers receive feedback on the success of their current and historic strategies, and can assess the payoffs received by the strategies of their competitors [6]. Hence, certain areas of the strategic landscape are illuminated.

In observing the adaptive efforts of real organizations it is clear that organizations interact with each other and receive feedback from their environment. Their efforts at strategic adaption are guided by *social* as well as individual learning. One group of models, combining both social and individual learning which has attracted interest in recent years are those drawn from a swarm metaphor [7, 8, 9, 10, 11, 12]. This paper introduces this metaphor to the domain of organization science, and constructs a simulation model based on the swarm metaphor. The simulator provides a new tool for the study of strategic adaptation.

1.1. *Organizational Adaptation*

In the literature on organizational adaptation there are two polar views concerning the ability of organizations to adapt. Adaptationists or advocates of strategic choice [13, 14, 15], broadly consider that managers or dominant coalitions in organizations scan the environment for current and future opportunities and threats, formulate strategic responses to these, and adjust organizational activities and structure appropriately [16]. Under this perspective, it is considered that organizations can successfully adapt based on what they observe in the environment, and based on their past experience. The current practitioner interest in 'change management' [17] exemplifies the belief that even substantial strategic adaptation is possible.

In contrast, the population ecology school [18] believes that the ability of organizations to accurately and consistently adapt in a world of high uncertainty, where connections between means and ends are unclear is doubtful [19, 20]. Population ecologists contend that an organization's fitness primarily arises because of good initial strategic choices, or luck, rather than reflecting post-founding adaptation [21]. While even ardent population ecologists admit that organizations do try to adapt, they consider that these adaptive efforts amount to little more than random search.

This study seeks to explore the adaptationists view, that the mechanisms of social learning and organizational memory (past experience) can indeed produce successful adaptation, by creating a simulation model in which the efficiency of these mechanisms in guiding adaptation is explored.

1.2. *Structure of Paper*

The remainder of this contribution is organized as follows. Section 2 describes the simulation model. The model consists of two main components, a strategic landscape and a swarm search algorithm. The third section provides the simulation results, followed by conclusions in section 4.

2. Simulation Model

The simulation model developed in this study can be classed as a multi-agent system (MAS). MASs focus attention on collective intelligence and on the emergence of behaviors through the interactions between the agents. MASs usually contain a world (environment), agents, definitions of the relations between the agents, a set of activities that the agents can perform, and changes to the environment as a result of these activities [22]. The key components of the simulation model are a landscape generator (which creates an environment), and the adaption of the canonical particle swarm algorithm to incorporate the activities and interactions of the agents (organizations). Each of these are described in the following subsections.

2.1. *Strategic Landscape*

The strategic landscape is defined using Kauffman's NK model [23, 24]. The NK model considers the behavior of systems which are comprised of a configuration (string) of N individual elements. Each of these elements are in turn interconnected to K other of the N elements (K<N). In a general description of such systems, each of the N elements can assume a finite number of states. If the number of states for each element is constant (S), the space of all possible configurations has N dimensions, and contains a total of $\prod_{i=1}^{N} S_i$ possible configurations.

In Kauffman's operationalization of this general framework, the number of states for each element is restricted to two (0 or 1). Therefore the configuration of N elements can be represented as a binary string. The parameter K, determines the degree of fitness interconnectedness of each of the N elements and can vary in value from 0 to N-1. In one limiting case where K=0, the contribution of each of the N elements to the overall fitness value (or worth) of the configuration are independent of each other. As K increases, this mapping becomes more complex, until at the upper limit when K=N-1, the fitness contribution of any of the N elements depends both on its own state, and the simultaneous states of all the other N-1 elements, describing a fully-connected graph.

If we let s_i represent the state of an individual element i, the contribution of this element (f_i) to the overall fitness (F) of the entire configuration is given by $f_i(s_i)$ when K=0. When K>0, the contribution of an individual element to overall fitness, depends both on its state, and the states of K other elements to which it is linked $(f_i(s_i : s_{i1}, \ldots, s_{ik}))$. A random fitness function $(U(0,1))$ is adopted, and the overall fitness of each configuration is calculated as the average of the fitness values of each of its individual elements. Therefore, if the fitness values of the individual elements are $f_1, \ldots, f_N$, overall fitness (F) is calculated as per **Eq.** 1.

$$F = \frac{\sum_{i=1}^{N} f_i}{N} \tag{1}$$

Altering the value of K effects the ruggedness of the described landscape (graph), and consequently impacts on the difficulty of search on this landscape [23, 24]. The strength of the NK model in the context of this study is that by tuning the value of K it can be used to generate strategic landscapes (graphs) of differing degrees of local-fitness correlation (ruggedness). A more detailed review and analysis of the NK model can be found in [25, 26, 27].

2.1.1. *Mapping Organizational Strategies to an NK Landscape*

It is noted *ab initio* that application of the NK model to define a strategic landscape is not atypical and has substantial support from existing literature in organizational science [2, 3, 28], and related work on technological innovation [29, 30, 31]. The strategy of an organization is characterized as consisting of N attributes [2]. Each of these attributes represents a strategic decision or policy choice, that an organization faces. Hence, a specific strategic configuration **s** is represented as a vector $s_1, \ldots, s_N$ where each attribute can assume a value of 0 or 1 [3]. The vector of attributes represents an entire organizational form. Therefore it embeds a choice of markets, products, internal structure, and method of competing in a chosen market. Good consistent sets of strategic decisions (strategic configurations) correspond to payoff peaks on the strategic landscape.

The definition of an organization as a vector of strategic attributes finds resonance in the work of Porter [13, 1], where organizations are conceptualized as a series of activities forming a value-chain. The choice of what activities to perform, and subsequent decisions as to how to perform these activities, defines the strategy of the organization. The individual attributes of an organization's strategy interact. For example, the value of an efficient manufacturing process is enhanced when combined with a high-quality sales force. Differing values for K correspond to varying degrees of payoff-interaction amongst elements of the organization's strategy [3]. As K increases, the difficulty of the task facing strategic decision makers is magnified. Local-search attempts to improve their organization's position on the strategic landscape become ensnared in a web of conflicting constraints.

2.2. *Particle Swarm Algorithm*

The PSA [7, 8] is based on a metaphor of human social interaction [9] and has been widely used for function optimization. The term PSA is used in place of PSO (Particle Swarm Optimization) in this study, as the object is not to develop a tool for 'optimizing', but to apply the particle swarm metaphor as a model of organizational adaptation. Under the particle swarm metaphor, a swarm of particles (entities) are assumed to move (fly) through an n-dimensional space, typically looking for a function optimum. Each of the particles has two associated properties, a current position and a velocity. Each particle also has a memory of the best location in the search space that it has found so far (the vector **pbest**), and knows the best location found to date by all the particles in the population (the vector **gbest**). At each step of the algorithm, particles are displaced from their current position by applying a velocity vector to them. The size and direction of this velocity is influenced by the velocity in the previous iteration of the algorithm (simulates 'momentum'), and the current location of a particle relative to its **pbest** and **gbest**. Therefore, at each step of the algorithm, the size and direction of each particle's move is a function of its *own history* (experience), and the *social influence* of its peer group. A number of variants of the PSA exist. The following paragraphs provide a short description of the *continuous* version of the PSA [7].

2.2.1. *Continuous Version of the PSA*

Visual intuition on the workings of the algorithm is provided in **Fig.** 1. Each particle i has an associated current position in search space $\mathbf{x_i(t)} = (x_{i1}(t), \ldots, x_{iN}(t))$, a current velocity of $\mathbf{v_i(t)} = (v_{i1}(t), \ldots, v_{iN}(t))$, and a **pbest** position of $\mathbf{y_i(t)} = (y_{i1}(t), \ldots, y_{iN}(t))$. The position of the particle at time $t+1$ is a determined by $\mathbf{x_i(t)} + \mathbf{v_i(t+1)}$, where $\mathbf{v_i(t+1)}$ is obtained by a stochastic blending of $\mathbf{v_i(t)}$, an acceleration towards **gbest** ($\mathbf{v_{gbest}}$) and an acceleration towards **pbest** ($\mathbf{v_{pbest}}$). During each iteration of the algorithm, $\mathbf{v_i(t+1)}$ and $\mathbf{x_i(t)}$ are updated using **Eqs.** 2 & 3.

Assuming a function f is to be maximized, that the swarm consists of m particles, and that r_1, r_2 are drawn from a uniform distribution in the range (0,1), the velocity update is as per **Eq.** 2.

$$\mathbf{v_i(t+1)} = W\mathbf{v_i(t)} + c_1 r_1(\mathbf{y_i} - \mathbf{x_i(t)}) + c_2 r_2(\hat{\mathbf{y}} - \mathbf{x_i(t)}) \tag{2}$$

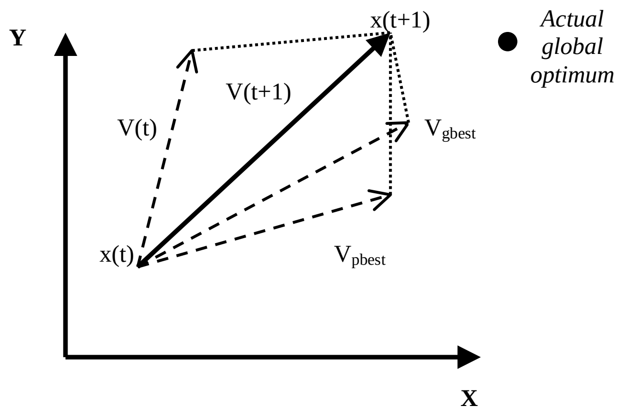

Fig. 1. A representation of the particle position update process.

where $\hat{\mathbf{y}}$ is the location of the global-best solution found by all the particles. A variant on the canonical algorithm is to use a local rather than a global version of **gbest**, whereby **gbest** is replaced by **lbest** (or local best). In this formulation of **Eq.** 2, **lbest** is set independently for each particle, based on the best point found thus far by any particle within a *neighborhood* (defined as a linked subset of particles) of each individual particle. In a social setting, the idea of **lbest** can be considered as a social network, whose members influence one another.

In every iteration of the algorithm, each particle's velocity is stochastically accelerated towards its previous best position and towards a neighborhood (local or global) best position. The weight-coefficients c_1 and c_2 control the relative impact of **pbest** and **gbest** locations on the velocity of a particle. The parameters r_1 and r_2 ensure that the algorithm is stochastic. Although the velocity update has a stochastic component, the search process is not random. It is guided by a memory of past good solutions, and by the neighborhood best solution. A practical effect of the random coefficients r_1 and r_2, is that the relative influence of the individual and the social learning terms in the velocity update equation (**Eq.** 2) will vary.

In **Eq.** 2, W represents a momentum coefficient which controls the impact of a particle's prior-period velocity on its current velocity. Each component of a velocity vector $\mathbf{v_i}$ is restricted to a range $[-v_{max}, v_{max}]$ to ensure that individual particles do not leave the search space. The implementation of a v_{max} parameter can also be interpreted as simulating the incremental nature of most social learning processes [9]. The value of v_{max} is usually chosen to be $k * x_{max}$, where $0 < k < 1$. Once the velocity update for particle i is determined, its position is updated (**Eq.** 3) and its pbest is updated if necessary, as described in **Eqs.** 4 & 5.

$$\mathbf{x_i(t+1) = x_i(t) + v_i(t+1)} \tag{3}$$

$$\mathbf{y_i(t+1)} = \mathbf{y_i(t)} \text{ if, } f(\mathbf{x_i(t)}) \leq f(\mathbf{y_i(t)}) \quad (4)$$

$$\mathbf{y_i(t+1)} = \mathbf{x_i(t)} \text{ if, } f(\mathbf{x_i(t)}) > f(\mathbf{y_i(t)}) \quad (5)$$

After all particles have been updated, a check is made to determine whether **gbest** needs to be updated (**Eq.** 6).

$$\mathbf{\hat{y}} \in (\mathbf{y_0}, \mathbf{y_1}, ..., \mathbf{y_n}) | f(\mathbf{\hat{y}}) = \max\ (f(\mathbf{y_0}), f(\mathbf{y_1}), ..., f(\mathbf{y_m})) \quad (6)$$

2.2.2. *Particle Swarm as a Metaphor for Organizational Adaptation*

Although particle swarm algorithms have been used extensively in function optimization (Particle Swarm Optimization), the original inspiration for PSAs arose from observations of animal and human social behavior [8]. Kennedy has published a series of papers which emphasize the social aspects of particle swarm [9, 35, 32] and this work was given prominence in the first major book on particle swarm [7].

The velocity update formula (**Eq.** 2) can be divided into *cognitive* and *social* components [9], with the former relating to the adaptive history of a particle, individual or in this study, an organization. The cognitive term can be considered as an interpretation of Thorndike's *Law of Effect* [33], which states that a behavior which is followed by a (positive) reinforcement becomes more likely in the future, corresponding to a form of learning from experience. The individual learning component in the velocity update formula ($\mathbf{y_i(t)} - \mathbf{x_i(t)}$) introduces a stochastic tendency to return to previously rewarded strategies, mimicking a psychological tendency for managers to repeat strategies which have worked for them in the past [9]. The social learning component of the formula ($\mathbf{\hat{y}_i(t)} - \mathbf{x_i(t)}$) bears comparison with social 'no-trial learning' [34], where the observation of a peer being reinforced for a behavior, will increase the probability of the observer engaging in the same behavior.

The mechanisms of the canonical PSA bear *prima facie* similarities to those of the domain of interest, organizational adaptation. The PSA adopts a populational perspective, and learning in the algorithm just as in populations of organizations, is both distributed and parallel. Organizations persist in employing already discovered good strategies, and are attracted to, and frequently imitate, good product ideas and business practices discovered by other organizations. However, the canonical PSA requires modification before it can employed as a component of a plausible simulation model of organizational adaptation. These modifications are discussed in the next subsection.

2.3. *Domain Characteristics*

Five characteristics of organizational adaptation which impact on the design of a plausible simulation model are:

i. Organizations are prone to strategic anchoring (inertia), and find it difficult to alter their strategy.

ii. Organizations do not intentionally select strategies which are expected to produce lower payoffs than the strategy they already have.
iii. Organizations make errorful assessments of fitness *ex-ante* the implementation of a new strategy.
iv. Organizations co-evolve.
v. The environment which organizations inhabit is dynamic.

Each of these factors is embedded in our simulation model, and our simulation experiments explore the first four characteristics. We note that the developed model bears passing resemblance to Kennedy's 'eleMentals' model [35], which combined a swarm algorithm and an NK landscape, to investigate the development of culture and intelligence in a population of hypothetical beings called 'eleMentals'. However, the simulator developed in this paper is differentiated from the eleMental model because of the incorporation of the five domain characteristics above.

2.3.1. *Strategic Anchoring*

Organizations do not have complete freedom to alter their current strategy. Their adaptive processes are subject to strategic inertia. This inertia springs from the organization's culture, history, and the mental models of its management [36]. In the simulation model, strategic inertia is mimicked by implementing a *strategic anchor*. The degree of inertia can be varied from zero to high. In the latter case, the organization is highly constrained from altering its current strategic stance. By allowing the weight of this anchor to vary, adaptation processes corresponding to different industries each with different levels of inertia, can be simulated.

Inertia could be incorporated into the PSA in a variety of ways. We have chosen to incorporate it into the velocity update equation, so that the velocity and direction of the particle at each iteration is also a function of the location of its strategic anchor. Therefore in coding the simulator, **Eq.** 2 is altered by adding an additional anchor term producing **Eq.** 7

$$\mathbf{v_i(t+1)} = \mathbf{v_i(t)} + R_1(\mathbf{y_i} - \mathbf{x_i(t)}) + R_2(\hat{\mathbf{l}} - \mathbf{x_i(t)} + R_3(\mathbf{a_i} - \mathbf{x_i(t)}) \tag{7}$$

As before, $\mathbf{v_i(t)}$ and $\mathbf{x_i(t)}$ represent the velocity and location of particle i at time t. $\mathbf{y_i}$ is the location of **pbest** for particle i, and $\hat{\mathbf{l}}$ is the location of the **lbest** of that particle. R_1, R_2 and R_3 are random weights drawn from a uniform distribution ranging from 0 to R_{1max}, R_{2max} and R_{3max} respectively, and they weight the importance attached to **pbest**, **lbest**, and particle i's anchor in each velocity update. The vector $\mathbf{a_i}$ represents the position of the anchor for organization i. The position of the anchor can be fixed at the initial position of the particle at the start of the algorithm, or it can be allowed to 'drag', thereby being responsive to the recent adaptive history of the particle. In the latter case, the position of the anchor for each particle corresponds to the position of that particle 'x' iterations ago. The weight attached to the anchor vector (relative to those attached to **pbest** and **lbest**), is set by the modeler at the beginning of the simulation run.

2.3.2. *Election operator*

Real-world organizations do not usually intentionally move to strategies which produce lower expected payoffs than their current strategy. This represents a search heuristic 'do not give up a good idea until a better one comes along'. Hence, a *ratchet operator* is implemented in the simulator, which when turned on, ensures that strategic changes which would worsen an organization's expected payoff are discarded. In these cases, an organization remains at its current location on the strategy landscape. One economic interpretation of the election operator, is that strategists carry out a mental simulation or thought experiment. If the expected fitness of the new strategy appears unattractive, the 'bad idea' is discarded [37, 7]. Ratchet mechanisms abound in business settings, and include formal project appraisal & review processes. A typical example of the application of a ratchet mechanism is the rejection of a proposed investment because it is not expected to generate a positive net present value (NPV) payoff. The decision heuristic 'reject negative NPV projects', corresponds to a ratchet mechanism.

2.3.3. *Errorful fitness assessment*

The concept of a ratchet operator raises a subtle but important issue on the interpretation of a strategy's payoff in the context of a model of strategic adaptation. The model would be flawed if it assumed that strategists could perfectly anticipate the *ex-ante* payoff of as yet, untested strategies. In real-world directed organizational adaptation, managers are guided by their guess as to the expected payoff of the proposed strategy.

Payoff expectations in the real-world, are subject to error. Strategists do not evaluate the worth of proposed strategies perfectly due to uncertainty and bounded rationality. The practical effect of this noise component is that managers may sometimes inadvertently replace their current strategy with one which ultimately produces a lower payoff. To the extent that managers have good understanding of the strategic landscape on which they operate the degree of noise when making payoff estimates is lessened. The effect of making errorful payoff assessments can be simulated by subjecting the assessments to noise using **Eq.** 8.

$$\text{fitness estimate} = \text{actual fitness of the new strategy} * (1+ \text{error}) \tag{8}$$

where *error* is drawn from a normal distribution with a mean of zero and a modeler-defined standard deviation. Hence, despite the ratchet operator, in the simulation experiments a strategist may sometimes choose a bad strategy because of an incorrect *ex-ante* assessment of its payoff.

2.3.4. *Co-evolution*

A key factor which impacts on the return to any organization is the degree of competition it faces from other organizations. If there are several organizations pursuing similar strategies they compete for the same customer segment, and the returns to each organization are likely to be lower than they would be in the absence of competition. In the model of strategic adaptation it is assumed that strategists employ a heuristic of 'take account of expected competition' (payoff sharing) when forming their expectations as to the likely payoff to a novel strategy. Consequently, a fitness-sharing mechanism is implemented in order to mimic the effect of competition between organizations. The fitness-sharing mechanism is defined in **Eq.** 9.

$$f'(i) = \frac{f(i)}{\sum_{j=1}^{n} s(d(i,j))} \tag{9}$$

where $f(i)$ represents the original raw fitness of particle i (from the NK landscape), $f'(i)$ represents the *shared fitness* of particle i, and $d(i,j)$ is the normalized distance between two particles i and j.[a] Drawing on [38], the *sharing function* $s(d)$, is defined as per **Eq.** 10.

$$s(d) = \left\{ \begin{array}{l} 1 - \left(\frac{d}{t}\right)^{\alpha} \text{ , if d<t} \\ 0 \text{ , otherwise} \end{array} \right\} \tag{10}$$

where t defines a radius or *neighborhood* within which particles share their fitness. The sharing function $s(d)$ provides a measure of the *density* of organizations within a 'neighborhood' of any individual organization i on the strategy landscape. If two particles (organizations) are more than t apart they are not considered to compete for the same market and $s(d)$=0; if they are co-located (zero distance apart) on the strategy landscape they compete for the same customers and $s(d)$=1; otherwise $s(d)$ produces a value in the range (0,1). α is a scaling constant, and the values of both t and α are adjustable in the simulation model.

The key concept embedded in the sharing function, is that close imitation of the strategy of another organization is *not precluded*, rather it is discouraged. If a particular strategy strongly dominates all others, it is likely that many organizations will attempt to implement it. The price paid for this is increased head-on competition. An intuitive way of thinking about the fitness-sharing mechanism is that it encourages organizations to search for good niches on the strategic landscape, balancing the profit potential of each niche, with the degree of competition that niche currently faces. Another way of considering the sharing mechanism is that it embeds a co-evolutionary aspect into the strategic environment. The actions of each organization deforms the fitness landscape faced by its competitors.

2.3.5. *Dynamic environment*

Organizations do not compete in a static environment. The environment may alter as a result of exogenous events, for example a regime change such as the emergence

[a] Measured using the Norm of the difference between the two strategic configurations, divided by *sqrt*(length of strategy vector). Hence, distance can be considered as related to the Hamming distance between the two (binary) strategic configurations.

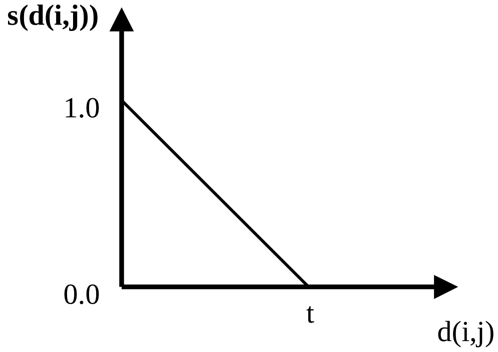

Fig. 2. A sharing function. As the distance between two organization's strategy increases, the degree of competition between them decreases.

of a new technology, or a change in customer preferences. This can be mimicked in the simulation by stochastically respecifing the strategic landscape during the course of a simulation run. These respecifications simulate a dynamic environment, and a change in the environment may at least partially negate the value of past learning (adaptation) by organizations. Minor respecifications are simulated by altering the fitness values associated with one of the N dimensions in the NK model, whereas in major changes, the fitness of the entire NK landscape is redefined.

2.4. *Outline of Swarm Algorithm*

As the strategic landscape is described using a binary representation (the NK model), the canonical PSA is adapted for the binary case using the *BinPSO* version of the algorithm [39]. The binary version of the PSA is inspired by the idea that an agent's probability of making a binary decision (yes/no, true/false) is a function of both personal history and social factors. The probability that an agent chooses a value of (for example) 1 for a particular decision in the next time period, is a function of the agent's history ($\mathbf{x_i(t)}, \mathbf{v_i(t)}$ & **pbest**), and social factors (**lbest**) (see **Eq.** 11).

$$Prob(\mathbf{x_i(t+1)} = 1) = f(\mathbf{x_i(t)}, \mathbf{v_i(t)}, \mathbf{pbest}, \mathbf{lbest}) \tag{11}$$

The vector $\mathbf{v_i}$ is interpreted as organization i's predisposition to set each of the N binary strategic choices that it faces to one. The higher the value of v_i^j for an individual decision j, the more likely that organization i will choose to set decision $j = 1$, with lower values of v_i^j favoring the choice of decision $j = 0$.

In order to model the tendency of managers to repeat historically good strategies, values for each dimension of $\mathbf{x_i}$ which match those of **pbest**, should become more probable in the future. Adding the difference between $pbest_i^j$ and x_i^j for organization i to v_i^j will increase the likelihood that organization i will choose to set decision

$j = 1$ if the difference is positive (when $pbest_i^j = 1$ and $x_i^j = 0$). If the difference between $pbest_i^j$ and x_i^j for organization i is negative (when $pbest_i^j = 0$, and $x_i^j = 1$), adding the difference to v_i^j will decrease v_i^j.[b]

In each iteration of the algorithm, the agent adjusts his decision-vector ($\mathbf{x_i(t)}$), taking account of his historical experience (**pbest**), and the best strategy found by his peer-group (**lbest**). Hence, the velocity update equation used in the continuous version of the PSA (**Eq.** 7) can still be used, although now, $\mathbf{v_i(t+1)}$ is interpreted as the updated vector of an agent's predisposition (or probability thresholds) to set each of the N binary strategic choices that it faces to one.

$$\mathbf{v_i(t+1)}=\mathbf{v_i(t)}+R_1(\mathbf{pbest_i}-\mathbf{x_i(t)})+R_2(\mathbf{lbest_i}-\mathbf{x_i(t)}+R_3(\mathbf{anchor_i}-\mathbf{x_i(t)}) \tag{12}$$

To ensure that each element of the vector $\mathbf{v_i(t+1)}$ is mapped into (0,1), a sigmoid transformation is performed on each element j of $\mathbf{v_i(t+1)}$ (see **Eq.** 13).

$$Sig(v_i^j(t+1)) = \frac{1}{1 + exp(-v_i^j(t+1))} \tag{13}$$

Finally, the transformed vector of probability thresholds is used to determine the values of each element of $x_i(t+1)$, by comparing each element of $Sig(\mathbf{v_i(t)})$ with a random number drawn from $U(0,1)$ (see **Eq.** 14).

$$\text{If } U(0,1) < Sig(v_i^j(t+1)), \text{ then } x_i^j(t+1) = 1; \text{ else } x_i^j(t+1) = 0 \tag{14}$$

In the binary version of the algorithm, trajectories / velocities are changes in the probability that a coordinate will take on a zero or a one value. $Sig(v_i^j)$ represents the probability of bit x_i^j taking the value 1 [39]. Therefore, if $Sig(v_i^j) = 0.3$ there is a thirty percent chance that $x_i^j = 1$, and a seventy percent chance it is zero.

2.4.1. *Pseudocode for Algorithm*

The pseudo-code for the swarm algorithm in the simulator is as follows:

```
For each entity in turn

For each dimension (strategic decision) n

v[n]=v[n]+R1*(pbest[n]-x[n])+R2*(lbest[n]-x[n])+R3*(a[n]-x[n])
If(v[n]>Max) v[n]=Vmax
    If(v[n]<-Vmax) v[n]=-Vmax
    If(Pr<Sig(v[n]))t[n]=1
    Else t[n]=0
If(fitness(t)*(1+error))>fitness(x))  //ratchet operator
    For each dimension n
        x[n]=t[n]
UpdateAnchor(a)        //if iteratively update anchor
                       //option is selected
```

[b]The difference in each case is weighted by a random number drawn from U(0,1). Therefore, if $pbest_i^j = 1$, $(pbest_i^j - x_i^j) * U(0,1)$ will be non-negative. Adding this to v_i^j will increase v_i^j, and therefore also increase the probability that $x_i^j = 1$. On the other hand if $pbest_i^j = 0$, v_i^j will tend to decrease, and $Prob(x_i^j) = 1$ becomes smaller.

R_1, R_2 and R_3 are random weights drawn from a uniform distribution ranging from 0 to R_{1max}, R_{2max} and R_{3max} respectively, and they weight the importance attached to pbest, lbest and anchor in each iteration of the algorithm. R_{1max}, R_{2max} and R_{3max} are constrained to sum up to 4.0 in line with the *BinPSO* alogrithm of [39]. The particle's actual position is denoted by x, *pbest* is its past best position, *lbest* its local best and a is the position of its anchor. V_{max} is set to 4.0 to ensure that $Sig(v[n])$ does not get too close to either 0 or 1, therefore ensuring that there is a non-zero possibility that a bit will flip state during each iteration. Pr is a random value drawn from U(0,1), Sig is the sigmoid function: $Sig(x) = \frac{1}{1+exp(-x)}$, which squashes v into the range $0 \rightarrow 1$, and t is a temporary record which is used in order to implement the ratchet operator. If the new strategy is considered better than the organization's existing strategy, it is accepted and t is copied into x. Otherwise t is discarded and x remains unchanged. The fitness evaluation step is subject to error (parameterized as *error*), in order to mimic a noisy forecast of a strategy's payoff.

2.5. *Simulator Model*

Although the underlying code for the *OrgSwarm* simulator is written in C++, the user interacts with the simulator through a series of easy-to-use screens (Fig. 3 shows one of the screens in the main control menu for the simulator). These screens allow the user to select and alter a wide variety of parameters which determine the nature of the simulation run. The simulator, along with a full manual, is available for download from http://ncra.ucd.ie.

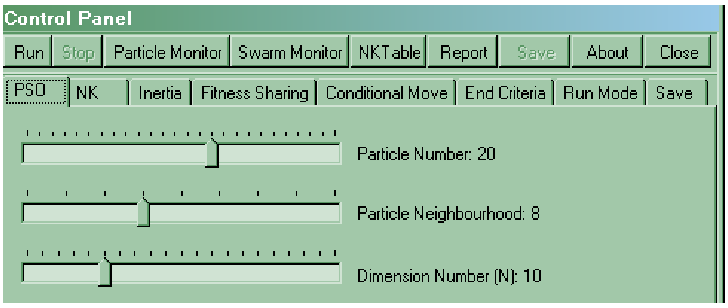

Fig. 3. Main control screen for OrgSwarm.

3. Experimental Set-up & Results

As otulined above, the key issue addressed in the simulation experiments is whether there is evidence that social learning, organizational memory, and the related mechanisms of inertia and ratchet, are sufficient to generate successful organizational adaption. Before running the simulations, a variety of parameters must be selected and these are described below. The choices for the parameters are not intended to be exhaustive, as it is not possible to combinatorially examine and report results from, every possible set of parameter settings in a single paper.

3.1. *Parameters Governing the Strategic Landscape*

'Real-world' strategy vectors consist of a large array of strategic decisions. A value of N=96 is selected in defining the landscapes in this simulation. There is no unique value of N that could be selected, however a binary string of length 96 provides a large number of distinct choices of strategy (2^{96} is approximately 10^{28}). It is noted that we would expect the dimensionality of the strategy vector to exceed the number of organizations in the population, hence the size of the population is kept below 96, and a population of 20 particles is employed, with a neighborhood of size 18.[c] The particles are considered to be arranged in a regular circular (ring) lattice, and each particle is connected to the nine particles on either side of it, and the particles are considered to be arranged in a circular structure in which particle 20 is 'beside' particle 1.[d]

In selecting the values of K for the simulations, we are guided by the work of [40] in which a distinction is drawn between *generic* and *firm-specific* activities of organizations. Generic activities (or table-stakes[e]) are those which have a similar optimal configuration for many firms, for example the possession of an accounting system. Generic activities, whilst important for the successful operation of the firm, are not strongly fitness-interconnected with the non-generic activities of the firm, and 'good' settings for these decision variables can typically be uncovered by managers. Hence, generic activities do not pose undue difficulty in the strategy determination process. In contrast, the firm-specific element of strategy are typically fitness-interconnected, as they embed choices involving trade-offs between alternative strategic configurations [1, 40]. Hence the NK landscape can be considered to represent these non-generic, elements of the strategy vector, rendering the assumption of a non-zero value of K plausible. On the other hand, given the multiplicity of organizations which exist and persist, organizations do not appear to inhabit highly rugged landscapes on which the slightest mis-step will prove fatal. We select three different K values (0,4 and 10) for our simulation experiments.

[c] Experiments with smaller neighborhood sizes suggested that the results were not qualitatively sensitive to neighborhood size.

[d] This connection structure resembles the *circle* communication pattern, as described in [41]. The choice of a high value for the neighborhood, relative to the size of the population, arises from the observation that real-world organizations generally know the profitability of their competitors, as financial statements of commercial organizations are public information in many countries.

[e] Called table-stakes as organizations must carry out these activities to gain entrance to 'play the game'.

3.2. *Parameters Governing the Swarm Algorithm*

In the simulations which consider the effect of a strategic anchor, both a fixed position anchor (fixed at the initial position of the particle at the start of the simulation) and a moving anchor are examined. In the latter case, the anchor is assumed to move with a lag of 20 iterations. Therefore, the position of the strategic anchor for an organization at t_n is the position of the organization at t_{n-20}. In the simulations for both the fixed position, and moving strategic anchor, $R_{3max} = 1$, and $R_{1max} = R_{2max} = 1.5$.

In the simulations which consider the effect of noisy or errorful payoff evaluation on the part of strategists, three values for the standard deviation for *error* are examined, 0, 0.05 and 0.20. These values representing a range of quality of the mental model of strategists' in assessing the payoff to proposed strategies.

In the fitness-sharing experiments, t is fixed at 0.5, and α is set at 1.[f] When selecting a value for α we were guided by values used in previous applications of the fitness-sharing formula [42, 43] and set $\alpha = 1$.

3.3. *Experiments*

As the adaptive process is stochastic, a single simulation run describes a single sample-path through time [44]. There are many possible sample-paths, so the results of the simulations are averaged over multiple (30) runs in order to uncover prevalent characteristics of the range of sample paths to which the system can give rise. All simulations were run for 5,000 iterations, and all reported fitnesses are the average population fitnesses, across the 30 separate simulation runs. On each of these runs, a new NK landscape is created, the positions and velocities of particles are randomly initialized, and new pbest and lbest positions are determined for each particle.
A total of seventeen distinct simulation experiments were undertaken, with each experiment being repeated on three different NK landscapes. Initially, a basic PSA is employed (*Basic PSA, No Anchor, No Error*), without an anchor or ratchet operator, and strategists are assumed to make error-free assessments of the worth of proposed strategic configurations. This simulates a population of organizations adapting on a strategic landscape, where members of the population have no strategic inertia, where organizations do not utilize a ratchet operator in deciding whether to alter their position on the strategic landscape, and where error-free fitness assessments can be made. Fitness sharing is not considered in this experiment.

A ratchet-inclusive version of the above search heuristic is then tested, in which payoffs to strategies can be assessed without error, and where organizations do not have a strategic anchor. Two variants on this (*Ratchet PSA, No Anchor, Error=0.05 & 0.20*), wherein strategists make errorful assessments of the worth of proto-strategies are also examined.

[f] Results for alpha=3 were also examined, and were found to be qualitatively similar to those for alpha=1.

Following this, a series of experiments which include a strategic anchor, both fixed and dragging (*Ratchet PSA, Initial Anchor, Weight=1, ..., Ratchet PSA, Moving Anchor(20,1), Error=0.20*), with / without errorful assessment of the fitness of proposed strategies are conducted. The strategic anchor ranges from a fixed position (fixed at an organization's initial position at the start of the simulation) anchor, to one which adapts after a time-lag (moving anchor). In both the initial and moving anchor experiments, a weight value of 1 is attached to the inertia term in the velocity update equation, and a time-lag of 20 periods is used for the moving anchor.[g]

The remaining experiments involve Ratchet PSAs with fitness-sharing, differing strategic anchor formulations, and differing levels of error in assessing the payoffs to potential strategies. In the fitness-sharing experiments, the radius is fixed at 0.5, and alpha is set at 1, therefore fitness-sharing will occur between strings which are separated by a Hamming distance of four or less.

3.4. *Results from Experiments*

Tables 1 - 2 and **Fig.** 4 provide the results for the simulation experiments which do not embed a fitness-sharing mechanism. **tables** 3 - 4 provide results for the fitness-sharing simulations. Examining the results in **table** 1 suggests that the basic PSA, without inertia or ratchet operators (*Basic PSA, No Anchor, No Error*), performs poorly on a static landscape even when there is no error in assessing the payoffs to potential strategies. The average populational fitness (averaged over each population, across all 30 simulation runs) obtained after 5,000 iterations is not better than random search, suggesting that unfettered adaptive efforts based on communication between organizations (lbest), and a memory of good past strategies (pbest), are not sufficient to achieve high levels of populational fitness, even when organizations can make error-free assessments of the payoff of potential strategies.

Algorithm	**Fitness (K=0)**	**(K=4)**	**(K=10)**
Basic PSA, No Anchor, No Error	0.4641	0.5002	0.4991
Ratchet PSA, No Anchor, No Error	0.5756	0.6896	0.6789
Ratchet PSA, No Anchor, Error=0.05	0.4860	0.6454	0.6701
Ratchet PSA, No Anchor, Error=0.20	0.4919	0.5744	0.5789
Ratchet PSA, Initial Anchor, Weight=1, No Error	0.6067	0.6991	0.6884
Ratchet PSA, Initial Anchor, Weight=1, Error=0.05	0.5297	0.6630	0.6764
Ratchet PSA, Initial Anchor, Weight=1, Error=0.20	0.4914	0.5847	0.5911
Ratchet PSA, Mov. Anchor (20,1), No Error	0.6692	0.7211	0.6976
Ratchet PSA, Mov. Anchor (20,1), Error=0.05	0.5567	0.6675	0.6770
Ratchet PSA, Mov. Anchor (20,1) Error=0.20	0.4879	0.5757	0.5837

Table 1. Average population fitness after 5,000 iterations.

[g]Therefore, the position of the strategic anchor for an organization at t_n is the position of the organization at t_{n-20}.

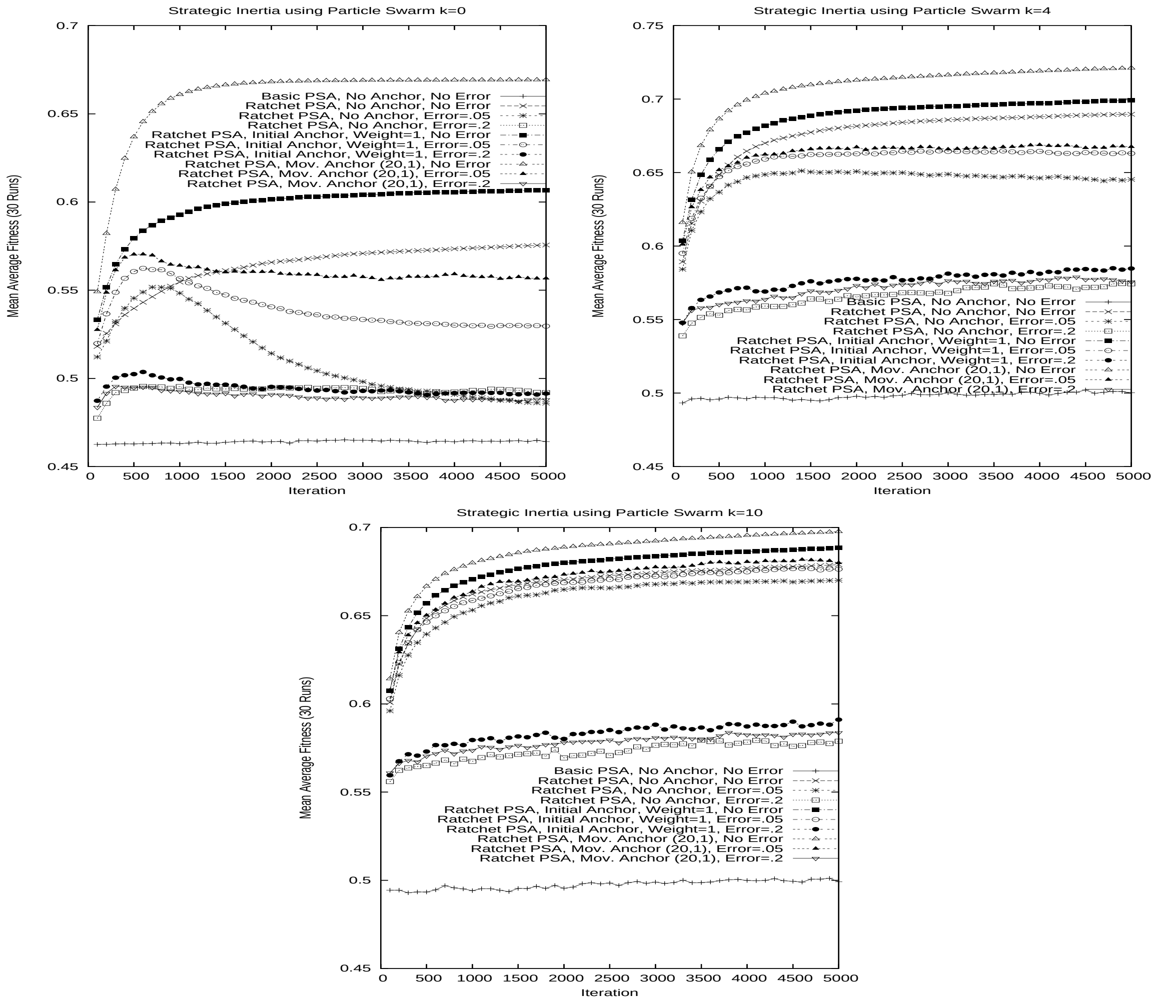

Fig. 4. Plot of the mean average fitness on the static landscape where k=0 (top left), k=4 (top right), and k=10 (bottom).

When a ratchet operator is added to the basic PSA (*Ratchet PSA, No Anchor, No Error*), a significant (at the 5% level) improvement in average populational fitness is obtained across landscapes of all K values, suggesting that the simple decision heuristic of *only abandon a current strategy for a better one* leads to notable increases in populational fitness.

In real-world organizations, assessments of the payoffs to potential strategies are not error-free. *A priori* we do not know whether this factor could impact positively or negatively on the populational fitness, as permitting errorful assessments of pay-off could allow an organization to escape from a local optimum on the strategic

Algorithm	**Fitness** **(K=0)**	**(K=4)**	**(K=10)**
Basic PSA, No Anchor, No Error	0.0274	0.0177	0.0113
Ratchet PSA, No Anchor, No Error	0.0324	0.0098	0.0079
Ratchet PSA, No Anchor, Error=0.05	0.0159	0.0103	0.0048
Ratchet PSA, No Anchor, Error=0.20	0.0205	0.0098	0.0069
Ratchet PSA, Initial Anchor, Weight=1, No Error	0.0288	0.0086	0.0058
Ratchet PSA, Initial Anchor, Weight=1, Error=0.05	0.0160	0.0095	0.0044
Ratchet PSA, Initial Anchor, Weight=1, Error=0.20	0.0239	0.0081	0.0074
Ratchet PSA, Mov. Anchor (20,1), No Error	0.0240	0.0080	0.0037
Ratchet PSA, Mov. Anchor (20,1), Error=0.05	0.0192	0.0067	0.0051
Ratchet PSA, Mov. Anchor (20,1) Error=0.20	0.0213	0.0094	0.0049

Table 2. Standard deviation of average populational fitness after 5,000 iterations.

Algorithm	**Fitness** **(K=0)**	**(K=4)**	**(K=10)**
Ratchet PSA, No Anchor, No Error	0.5698	0.6862	0.6785
Ratchet PSA, No Anchor, Error=0.05	0.4851	0.6381	0.6682
Ratchet PSA, No Anchor, Error=0.20	0.4760	0.5478	0.5646
Ratchet PSA, Initial Anchor, Weight=1, Error=0.05	0.5338	0.6604	0.6774
Ratchet PSA, Initial Anchor, Weight=1, Error=0.20	0.4929	0.5781	0.5879
Ratchet PSA, Mov. Anchor (20,1), Error=0.05	0.5597	0.6671	0.6815
Ratchet PSA, Mov. Anchor (20,1), Error=0.20	0.4809	0.5721	0.5826

Table 3. Average population fitness after 5,000 iterations for fitness-sharing experiments.

Algorithm	**Fitness** **(K=0)**	**(K=4)**	**(K=10)**
Ratchet PSA, No Anchor, No Error	0.0250	0.0112	0.0073
Ratchet PSA, No Anchor, Error=0.05	0.0195	0.0112	0.0046
Ratchet PSA, No Anchor, Error=0.20	0.0304	0.0216	0.0123
Ratchet PSA, Initial Anchor, Weight=1, Error=0.05	0.0236	0.0085	0.0048
Ratchet PSA, Initial Anchor, Weight=1, Error=0.20	0.0220	0.0112	0.0071
Ratchet PSA, Mov. Anchor (20,1), Error=0.05	0.0204	0.0090	0.0044
Ratchet PSA, Mov. Anchor (20,1), Error=0.20	0.0221	0.0111	0.0059

Table 4. Standard deviation of average populational fitness after 5,000 iterations for fitness-sharing experiments.

landscape, and possibly therefore to uncover a new lbest. In essence, an errorful assessment of payoff may allow a short-term 'wrong-way' move (one which temporarily reduces an organization's strategic payoff), but which in the longer-term leads to higher payoffs. Conversely, it could lead to the loss of a promising but underdeveloped strategy, if an organization is led away from a promising part of the strategic landscape by an incorrect payoff assessment. The results from the simulation experiments when noise (error) is injected into the fitness assessment of strategists suggest that errorful *ex-ante* assessment of strategic worth is harmful, as average populational fitness falls for all values of K.

When the two forms of strategic anchoring are incorporated into the experiments, it is noted that strategic anchoring, combined with a ratchet operator leads to an increase (statistically significant at the 5% level) in average populational fitness over the basic no anchor, ratchet mechanism. As the error level increases from zero to 0.20, the anchor-ratchet combination produces better average populational fitness than the ratchet mechanism alone, but the differences narrow and are not statistically significant once the error level reaches 0.20. Hence, *inertia enhances average populational fitness* when strategists can assess *ex-ante* strategic payoffs with accuracy. Comparing the results for the two forms of strategic anchoring indicates that a moving anchor performs better than a fixed anchor, when organizations can make error-free assessments of the payoff to potential strategies (statistically significant at the 5% level), but when these payoffs are subject to error, neither form of strategic anchor clearly dominates the other in terms of producing the higher average populational fitness.

Examining the results for the simulations which embed fitness-sharing (competition between organizations), the same general picture is observed as for the experiments with no fitness-sharing. The implementation of a ratchet operator (with no strategic anchor and no error in assessing fitness) notably improves average populational fitness over the basic PSA with no anchor and no error. When errorful assessment of fitness is introduced, it is seen that average populational fitness declines, suggesting that the utility of the ratchet operator decreases as the level of error in assessing the payoff to potential strategies rises.

In summary, the results suggest that a ratchet mechanism can act to notably increase the fitness of a population of organizations which are searching on a strategic landscape. The results also suggest that strategic anchoring produces higher average populational fitness once strategists can accurately assess payoffs to proposed new strategies. Error in the ability of strategists to assess these payoffs, leads to lower average populational payoffs.

4. Conclusions

In this study a novel simulation model, a synthesis of a strategic landscape defined using the NK framework and a particle swarm algorithm, is used to examine whether social learning and past organizational experience could prove useful in organizational adaption. Adoption of the swarm metaphor allows the incorporation of both social and individual learning mechanisms, and the basic particle swarm algorithm can be easily adapted to include other search heuristics such as ratchet and anchoring.

The results from the simulation experiments suggest that a degree of strategic anchoring can assist rather than hamper the adaptive efforts of populations of organizations in static strategic environments, once strategists can make reasonably accurate predictions as to the payoffs from proposed strategies. This result provides an interesting perspective on the claim by [19] that strategic anchoring may be a consequence of market-selection processes. The results suggest that there may be good reasons from a populational perspective, for market selection processes to encourage the development of populations of organizations which exhibit strategic anchoring.

The simulation results also suggest that despite the claim for the importance of social learning and past experience in populations, social learning and past experience alone is not enough, unless already-learnt lessons can be maintained by means of a ratchet mechanism. Examples of ratchet mechanisms abound in business, ranging from formal project appraisal systems, to procedures for monitoring the performance of on-going product development projects. The results support the assertion that managers should undertake a formal assessment of the worth of a proto-strategy, before its implementation. Additionally, the results support the assertion that the utility of a ratchet mechanism is reasonably robust, as it remains useful across strategy landscapes of differing levels of connectedness, and in conditions where strategists make imperfect or noisy assessments of the payoffs to proposed strategies.

The benefits of strategic anchoring come at a price. The effect of lbest, pbest and anchoring, is to 'pin' each organization to a region of the strategic landscape. To the extent that the entire population of organizations have converged to a relatively small region of the strategic landscape, they may find it difficult to migrate to a new high-fitness region if that region is far away from their current location. This suggests that the benefits of an inertia heuristic (strategic anchoring) for a population of organizations comes at a price, the risk of catastrophic failure of the entire population to adapt to a major change in the strategic landscape. In real-world environments, this is compensated for by the birth of new organizations.

It is not possible in a single set of simulation experiments to exhaustively examine every possible combination of settings for each parameter in the simulation model. Future work will extend the range of settings examined. However, the initial results cast an interesting light on the role of anchoring in organizational adaptation, and the development of the swarm-landscape simulator extends the methodologies available to researchers to conceptualize and examine the process of organizational adaptation.

As already noted in the introduction to this paper, the population ecology school of organizational adaptation is pessimistic on the ability of managers to correctly impound information from both their past experience and the environment, and to consequently engage in successful strategic adaptation. Under this perspective, organizational adaption primarily occurs through random search and the replacement of poorly performing organizations by new ones. An interesting avenue for future research would be to compare the results from the social learning-based models in this study with the results generated by variants of a simple random search mechanism (with/without ratchet, anchor mechanisms etc.). This would allow focus to be placed on the question: does social learning and organizational memory help organizational adaptation? In other words, wherein lies the balance between the adaptionist and the population ecology viewpoints?

Finally, it is noted that the concept of anchoring developed in this paper is not limited to organizations, but is plausibly a general feature of social systems. Hence, the extension of the social swarm model through the incorporation of an anchoring term may prove useful beyond this study.

Acknowledgements. The authors wish to thank James Kennedy for his helpful feedback on an earlier version of this work. We also wish to thank the anonymous reviewers.

References

[1] Porter, M., What is Strategy?, *Harvard Business Review*, **70**, 61-78, (1996).
[2] Levinthal, D., Adaptation on Rugged Landscapes, *Management Science*, **43**(7), 934-950, (1997).
[3] Rivkin, J., Imitation of Complex Strategies, *Management Science*, **46**(6), 824-844, (2000).
[4] Wright, S., The roles of mutation, inbreeding, crossbreeding and selection in evolution, *Proceedings of the Sixth International Congress on Genetics*, **1**, 356-366, (1932).
[5] Levinthal, D. and March, J., A Model of Adaptive Organizational Search, *Journal of Economic Behavior and Organization*, **2**, 307-333, (1981).
[6] Kitts, B., Edvinsson, L. and Beding, T., Intellectual capital: from intangible assets to fitness landscapes, *Expert Systems with Applications*, **20**, 35-50, (2001).
[7] Kennedy, J., Eberhart, R. and Shi, Y., *Swarm Intelligence*, (Morgan Kauffman, San Mateo, California, 2001).
[8] Kennedy, J. and Eberhart, R., Particle swarm optimization, *Proceedings of the IEEE International Conference on Neural Networks*, 1942-1948, (IEEE Press, 1995).
[9] Kennedy, J., The particle swarm: Social adaptation of knowledge, *Proceedings of the International Conference on Evolutionary Computation*, 303-308, (IEEE Press, 1997).
[10] Hu, X., Shi, Y. and Eberhart, R., Recent Advances in Particle Swarm, *Proceedings of the International Conference on Evolutionary Computation*, 90-97, (IEEE Press, 2004).
[11] Bonabeau, E., Dorigo, M. and Theraulaz, G., *Swarm Intelligence: From natural to artificial systems*, (Oxford University Press, Oxford, 1999).
[12] Bonabeau, E. and Meyer, C., Swarm intelligence: A whole new way to think about business, *Harvard Business Review*, **79**(5), 106-114, (2001).
[13] Porter, M., *Competitive Advantage: Creating and Sustaining Superior Performance*, (The Free Press, New York, 1985).
[14] Simon, H., Strategy and Organizational Evolution, *Strategic Management Journal*, **14**, 131-142, (1993).
[15] March, J., Footnotes to Organizational Change, *Administrative Science Quarterly*, **26**, 563-577, (1981).
[16] Child, J., Organizational Structure, Environment and Performance: The Role of Strategic Choice, *Sociology*, **6**,2-22, (1972).
[17] Hammer, M. and Champy, J., Reengineering the corporation: A manifesto for business revolution, (HarperBusiness, New York, 2001).
[18] Hannan, M. and Freeman, J., The Populational Ecology of Organizations, *American Journal of Sociology*, **82**(5), 929-964, (1977).
[19] Hannan, M. and Freeman, J., Structural Inertia and Organizational Change, *American Sociological Review*, **49**, 149-164, (1984).

[20] Carroll, G. and Hannan, T., *Organizations in Industry: strategy, structure and selection*, (Oxford University Press, New York, 1995).
[21] Barnett, W. and Hansen, M., The Red Queen in Organizational Evolution, *Strategic Management Journal*, **17**, 139-157, (1996).
[22] Tanev, I. and Shimohara, K., On Role of Implicit Interaction and Explicit Communications in Emergence of Social Behaviour in Continuous Predators-Prey Pursuit Problem, *Proceedings of GECCO 2003*, Lecture Notes in Computer Science (2723), 74-85, (Springer-Verlag, Berlin, 2003).
[23] Kauffman, S. and Levin, S., Towards a General Theory of Adaptive Walks on Rugged Landscapes, *Journal of Theoretical Biology*, **128**, 11-45, (1987).
[24] Kauffman, S., *The Origins of Order*, (Oxford University Press, Oxford, 1993).
[25] Reidys, C. and Stadler, P., Combinatorial Landscapes, *SIAM Review*, **44**, 3-54, (2002).
[26] Stadler, P., Fitness Landscapes, *in Biological Evolution and Statistical Physics*, M. Lssig and A. Valleriani (eds.), 187-207, (Springer-Verlag, Berlin, 2002).
[27] Stadler, P., Towards a Theory of Landscapes, *in Complex Systems and Binary Networks (Proceedings of the Guanajuato Lectures 1995)*, R. Lopz-Pea, R. Capovilla, R. Garca-Pelayo, H. Waelbroeck, and F. Zertuche (eds.), 77-163, (Springer-Verlag, Berlin, 1995).
[28] Gavetti, G. and Levinthal, D., Looking Forward and Looking Backward: Cognitive and Experiential Search, *Administrative Science Quarterly*, **45**, 113-137, (2000).
[29] Lobo, J. and MacReady, W., Landscapes: A Natural Extension of Search Theory, *Working Paper 99-05-037*, (Santa Fe Institute, 1999).
[30] Kauffman, S., Lobo, J. and MacReady, W., Optimal Search on a Technology Landscape, *Working Paper 98-10-091*, (Santa Fe Institute, 1998).
[31] Strumsky, D. and Lobo, J., If it isn't broken, don't fix it: Extremal search on a technology landscape, *Working Paper 03-02-003*, (Santa Fe Institute, 2002).
[32] Kennedy, J., Small worlds and mega-minds: effects of neighbourhood topology on particle swarm performance, *Proceedings of the International Conference on Evolutionary Computation*, 1931-1938, (IEEE Press, 1999).
[33] Thorndike, E. Animal Intelligence, (Macmillian, New York, 1911).
[34] Bandura, A. *Social foundations of thought and action: A social cognitive theory*, (Englewood Cliffs, New Jersey: Prentice Hall, 1986).
[35] Kennedy, J., Minds and Cultures: Particle Swam Implications for Beings in Sociocognitive Space, *Adaptive Behavior*, **7**(3/4), 269-288, (1999).
[36] Boeker, W., Strategic Change: The Effects of Founding and History, *Academy of Management Journal*, **32**(3), 489-515, (1989).
[37] Birchenhall, C., Modular Technical Change and Genetic Algorithms, *Computational Economics*, **8**, 233-253, (1995).
[38] Mahfoud, S., Population size and genetic drift in fitness sharing, *Foundations of Genetic Algorithms 3*, Whitley, L. and Vose, M. (eds), (Morgan Kauffmann, San Francisco, 1995).
[39] Kennedy, J. and Eberhart, R. A discrete binary version of the particle swarm algorithm, *Proceedings of the Conference on Systems, Man and Cybernetics*, 4104-4109, (IEEE Press, Piscataway, New Jersey, 1997).
[40] Porter, M. and Siggelkow, N., Contextuality within Activity Systems, *Working Paper Series No. 01-053-2001*, (Harvard Business School, 2001).
[41] Leavitt, H. Some effects of certain communication patterns on group performance, *Journal of Abnormal and Social Psychology*, **46**, 38-50, (1951).
[42] Mahfoud, S. (1995). Niching Methods for Genetic Algorithms, *Unpublished PhD*

Thesis, Department of General Engineering, University of Illinois at Urbana-Champaign.

[43] Zhang, J., Yuan, X. and Buckles, B. Subspace FDC for Sharing Distance Estimation, in *Proceedings of CEC 2004*, 2:1735-1741, (IEEE Press, Piscataway, New Jersey, 2004).

[44] Ballot, G. and Weisbuch, G., Introduction: Why Simulation in Social Sciences, *Advances in Complex Systems*, **3**, 9-17, (2000).