A Particle Swarm Model of Organizational Adaptation

Anthony Brabazon[1], Arlindo Silva[2,3], Tiago Ferra de Sousa[3], Michael O'Neill[4], Robin Matthews[5], and Ernesto Costa[2]

[1] Faculty of Commerce, University College Dublin, Ireland.
anthony.Brabazon@ucd.ie
[2] Centro de Informatica e Sistemas da Universidade de Coimbra, Portugal.
ernesto@dei.uc.pt
[3] Escola Superior de Tecnologia, Instituto Politecnico de Castelo Branco, Portugal.
arlindo@est.ipcb.pt.
[4] Biocomputing & Development Systems, University of Limerick, Ireland.
Michael.ONeill@ul.ie
[5] Centre for International Business Policy, Kingston University, London.

Abstract. This study introduces the particle swarm metaphor to the domain of organizational adaptation. A simulation model (*OrgSwarm*) is constructed to examine the impact of strategic inertia, in the presence of errorful assessments of future payoffs to potential strategies, on the adaptation of the strategic fitness of a population of organizations. The results indicate that agent (organization) uncertainty as to the payoffs of potential strategies has the affect of lowering average payoffs obtained by a population of organizations. The results also indicate that a degree of strategic inertia, in the presence of an election mechanism, assists rather than hampers adaptive efforts in static and slowly changing strategic environments.

1 Introduction

The objective of this study is to investigate the affect of strategic inertia, in the presence of uncertainty as to future payoffs to potential strategies, on the rate of strategic adaptation of a population of organizations. Following a long-established metaphor of adaptation as search [14], strategic adaptation is considered in this study as an attempt to uncover peaks on a high-dimensional strategic landscape. Some strategic configurations produce high profits, others produce poor results. The search for good strategic configurations is difficult due to the vast number of configurations possible, uncertainty as to the nature of topology of the strategic landscape faced by an organization, and changes in the topology of this landscape over time. Despite these uncertainties, the search process for good strategies is not blind. 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 [9]. Hence, certain areas of the strategic landscape are illuminated. In an organizational setting, a strategy can be conceptualized as being the choice of what activities an organization will

K. Deb et al. (Eds.): GECCO 2004, LNCS 3102, pp. 12–23, 2004.

perform, and the subsequent choices as to how these activities will be performed [12]. These choices define the strategic configuration of the organization. Recent work by [10] and [13] 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. This study adopts a similar approach, and employs the NK framework to generate a series of strategic landscapes. The performance of a population of organizations in searching the landscapes for high payoff locations under differing search heuristics, is examined. A key characteristic of the framework which integrates the search heuristics examined in this study, is that organizations do not adapt in isolation, but interact with each other. Their efforts at strategic adaption are guided by 'social' as well as individual learning. The work of [5,8], drawing on a swarm metaphor, has emphasized similar learning mechanisms. We extend this work into the organizational domain, by constructing a simulation model (*OrgSwarm*) based on this metaphor to examine the affect of strategic inertia on the rate of strategic adaptation of a population of organizations.

2 Particle Swarm Algorithm

This section provides an introduction to the basic Particle Swarm algorithm (PSA).[1] A fuller description of this algorithm and the cultural model which inspired it is provided in [5,8]. 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 particle is assumed to have 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 (*pbest*), and knows the location of the best location found to date by all the particles in the population (*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, 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 description of the basic continuous version described by [8]. Each particle i has an associated current position in search space x_i, a current velocity v_i, and a personal best position in search space y_i. During each iteration of the algorithm, the location and velocity of each particle is updated using equations (1-2). Assuming a function f is to be maximized, that the swarm consists of n particles, and that r_1, r_2 are drawn from a uniform distribution in the range (0,1), the velocity update is as follows:

[1] The term PSA is used in place of PSO (Particle Swarm Optimization) in this paper, as the object is not to develop a tool for 'optimizing', but to adapt and apply the swarm metaphor as a model of organizational adaptation.

$$v_i(t+1) = \Upsilon(Wv_i(t) + c_1r_1(y_i - x_i(t)) + c_2r_2(\hat{y} - x_i(t))) \quad (1)$$

where ŷ is the location of the global-best solution found by all the particles. In every iteration of the algorithm, each particle's velocity is stochastically accelerated towards its previous best position and towards a neighborhood (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. A practical affect of the random coefficients r_1 and r_2, is that neither the individual nor the social learning terms are always dominant. Sometimes one or the other will dominate [8]. Although the velocity update has a stochastic component, the search process is not 'random'. It is guided by the memory of past 'good' solutions (corresponding to a psychological tendency for individuals to repeat strategies which have worked for them in the past [6]), and by the global best solution found by all particles thus far. 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 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 learning processes [6]. The value of v_{max} is usually chosen to be $k * x_{max}$, where $0 < k < 1$. Υ represents a *constriction coefficient* which reduces in value during iterations of the algorithm. This ensures that particles tend to converge over time, as the amplitude of their oscillations (caused by the velocity equation) decreases [8]. Once the velocity update for particle i is determined, its position is updated and pbest is updated (equations 3-4) if necessary.

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

$$y_i(t+1) = y_i(t) \text{ if, } f(x_i(t)) \leq f(y_i(t)), \quad (3)$$

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

After all particles have been updated, a check is made to determine whether gbest needs to be updated (equation 5).

$$\hat{y} \in (y_0, y_1, ..., y_n) | f(\hat{y}) = \max\, (f(y_0), f(y_1), ..., f(y_n)) \quad (5)$$

Despite its simplicity, the algorithm is capable of capturing a surprising level of complexity, as individual particles are capable of both individual and social learning. Learning is 'distributed' and parallel. Communication (interactions) between agents (individuals) in a social system may be direct or indirect. An example of the former could arise when two organizations trade with one another. Examples of the latter include

i. The observation of the success (or otherwise) of a strategy being pursued by another organization, and
ii. 'Stigmergy' which arises when an organization modifies the environment, which in turn causes an alteration of the actions of another organization at a later time

The mechanisms of the Particle Swarm algorithm bear *prima facie* similarities to those of the domain of interest, organizational adaptation. It embeds concepts of a population of entities which are capable of individual and social learning. However, the model requires modification before it can employed as a plausible model of organizational adaptation. These modifications, along with a definition of the strategic landscape used in this study are discussed in the next section.

3 Simulation Model

The two key components of the simulation model are the landscape generator (environment), and the adaption of the basic Particle Swarm algorithm to incorporate the activities and interactions of the agents (organizations).

3.1 Strategic Landscape

In this study, the strategic landscape is defined using Kauffman's NK model [3, 4]. It is noted *ab initio* that application of the NK model to define a strategic landscape is not atypical and has support from existing literature in organizational science [10,13], [2]. 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 [4], 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 ($f_i(s_i : s_{i1}, ..., s_{ik})$). 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 $f_1, ..., f_N$, overall fitness (F) is $F = \left[\frac{\sum_{i=1}^{N} f_i}{N}\right]$. Altering the value of K affects the ruggedness of the described landscape (graph), and consequently impacts on the difficulty of search on this landscape [3], [4]. As K increases, the landscape becomes more rugged, and the best peaks on the landscape become higher, but harder to find. 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). The strategy of an organization is characterized as consisting of N attributes [10]. Each of these attributes represents a strategic decision or policy choice, that an organization faces. Hence, a specific strategic configuration $\mathbf{s}$, is represented as a vector $s_1, \ldots, s_N$ where each attribute can assume a value of 0 or 1 [13]. The vector of attributes represents an entire organizational form, hence it embeds a choice of markets, products, method of competing in a chosen market, and method of internally structuring the organization [13]. Good consistent sets of strategic decisions - configurations, correspond to peaks on the strategic landscape. The definition of an organization as a vector of strategic attributes finds resonance in the work of Porter [11,12], 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 among elements of the organization's strategy [13].

3.2 Simulation Model

Five characteristics of the problem domain which impact on the design of a plausible simulation model are:

i. The environment is dynamic
ii. Organizations are prone to strategic anchoring (inertia)
iii. Organizations do not knowingly select poorer strategies than the one they already have (election operator)
iv. Organizations make errorful *ex-ante* assessments of fitness
v. Organizations co-evolve

In this study, our experiments consider the first four of these factors. Future work will include the fifth factor. We note that this model bears passing resemblance to the 'eleMentals' model of [7], 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 'strategic' model developed in this study is differentiated from the eleMental model, not just on grounds of application domain, but because of the inclusion of an 'inertia' operator, and also through the investigation of both static and dynamic environments.

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 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.

Inertia. Organizations do not have complete freedom to alter their current strategy. Their adaptive processes are subject to 'conservatism' arising from inertia. Inertia springs from the organization's culture, history, and the mental models of its management. 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 for the simulations, equation 1 is altered by adding an additional 'inertia' term:

$$v_i(t+1) = v_i(t) + R_1(y_i - x_i(t)) + R_2(\hat{y} - x_i(t) + R_3(a_i - x_i(t)) \qquad (6)$$

where a_i represents the position of the anchor for organization i (a full description of the other terms such as R_1 is provided in the pseudo-code below). 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. Both the weight attached to the anchor parameter (relative to those attached to pbest and gbest), can be altered by the modeler. Two other alterations are made to the velocity update equation as originally stated in equation 1. The momentum term W and the constriction coefficient Υ are omitted on the grounds that these factors implicitly embed an inertia component. Including these terms could therefore bias the comparison of populations of organizations operating with/without an inertia heuristic.

Election operator. Real-world organizations do not usually intentionally move to 'poorer' strategies. Hence, an 'election' operator is implemented, whereby position updates which would worsen an organization's strategic fitness are discarded. In these cases, an organization remains at its current location. 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. The simulation incorporates a *conditional update* or *ratchet* operator option, which if turned on, ensures that an organization only updates (alters) its strategy if the new strategy being considered is better than its current strategy. Unfortunately, such evaluations in the real-world, are subject to error. Strategists do not evaluate proposed strategies perfectly due to uncertainty and bounded rationality. The affect of errorful assessments is simulated by subjecting the assessments to 'noise' using the following formula:

$$\text{fitness estimate} = \text{actual fitness of the new strategy} * (1+ \text{`error'}) \quad (7)$$

where *error* is drawn from a normal distribution with a mean of zero and a modeler-defined standard deviation. Hence, despite the election operator, a strategist may sometimes choose a 'bad' strategy because of an incorrect assessment of its fitness.

Outline of algorithm. A number of further modifications to the basic PSA are required. As the strategic landscape is defined using a binary representation, the basic PSA is adapted for the binary case. The pseudocode for the algorithm is as follows:

```
For each dimension n
    v[n]=v[n]+R1*(p[n]-x[n])+R2*(l[n]-x[n])+R3*(a[n]-x[n])
    If(v[n]>Max) v[n]=Vmax
    If(v[n]<-Vmax) v[n]=-Vmax
    If(Pr<S(v[n]))t[n]=1
    Else t[n]=0
If(fitness(t)*err)>fitness(x))  //conditional move
    For each dimension n
        x[n]=t[n]
UpdateAnchor(a)         //if iteratively update anchor
                         //option is selected
```

$R1$, $R2$ and $R3$ are random weights drawn from a uniform distribution ranging from 0 to $R1_{max}$, $R2_{max}$ and $R3_{max}$ respectively, and they weight the importance attached to gbest, lbest and the anchor in each iteration of the algorithm. $R1$, $R2$ and $R3$ are constrained to sum up to 4.0. Therefore changing the weight value of the anchor alters its significance in the adaptive process, relative to the importance of gbest and lbest. x is the particle's actual position, p is its past best position, l the local best and a is the position of the particle's anchor. V_{max} is set to 4.0. Pr is a probability value drawn from a uniform distribution ranging from 0 to 1, and S is the sigmoid function: $S(x) = \frac{1}{1+exp(-x)}$, which squashes v into a 0 to 1 range, in order to implement a binary PSA. t is a temporary record which is used in order to implement conditional moving. If the new strategy is accepted, t is copied into x, otherwise t is discarded and x remains unchanged. err is the error or noise, injected in the fitness evaluation, in order to mimic an errorful forecast of the payoff to a proposed strategy.

4 Results

All reported fitnesses are the average population fitnesses, and average environment best fitnesses, across 30 separate simulation runs at the conclusion of the 5,000 iterations. On each simulation run, the NK landscape is specified anew, and the positions and velocities of particles are randomly initialized at the start of each run. The simulations employ a population of 20 particles, with a circular neighborhood of size 18. 'Real-world' strategy vectors consist of a large array of strategic decisions. A value of N=96 is selected (arbitrary) in defining the landscapes in this simulation. A series of landscapes of differing K values (0,4 and 10), representing differing degrees of fitness interconnectivity, were used in the simulations.

Tables 1 and 2 provide the results for each of ten distinct PSA 'variants', at the end of 5,000 iterations, across three static and dynamic NK landscape 'scenarios'. In each scenario, the same series of simulations are undertaken. Initially, a basic PSA is employed, without an anchor or a conditional move operator. This simulates a population of organizations searching a strategic landscape, where members of the population have no strategic inertia, and where organizations do not utilize a ratchet (conditional move) operator in deciding whether to alter their position on the strategic landscape. The basic PSA is then supplemented by a series of strategic anchor formulations, ranging from a fixed position (fixed at a randomly chosen initial position) anchor which does not change position during the simulation, 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 equation, and a time-lag of 20 periods is used for the moving anchor. In the experiments concerning the affect of error when assessing the future payoffs to potential strategies, three values of *error* are examined, 0, 0.05 (5%) and 0.20 (20%).

4.1 Static Landscape

Table 1 provides the results for the static NK landscape. Examining these results suggests that the basic PSA, without inertia or ratchet operators, 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 'social communication' between organizations (gbest), and a memory of good past strategies (pbest) is not sufficient to achieve high levels of populational fitness, even when organizations can make error-free assessments of the 'payoff' of potential strategies. When a ratchet operator is added to the basic PSA (Ratchet PSA-No Anchor), a significant improvement (statistically significant at the 5% level) in both average populational, and average environment best 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.

Errorful Assessment of Strategic Fitness. In real-world organizations, assessments of the payoffs to potential strategies are not error-free. *A priori* we do not know whether this could impact positive or negatively on the evolution of populational fitness, as permitting errorful assessments of payoff could allow an organization to escape from a local optimum on the strategic landscape, and

Table 1. Average fitness after 5,000 iterations, static landscape.

Algorithm	Fitness (N=96, K=0)	(N=96, K=4)	(N=96, K=10)
Basic PSA	0.4641	0.5002	0.4991
Ratchet PSA-No Anchor	0.5756	0.6896	0.6789
Ratchet-No Anchor, e=0.05	0.4860	0.6454	0.6701
Ratchet-No Anchor, e=0.20	0.4919	0.5744	0.5789
Ratchet-Initial Anchor, w=1	0.6067	0.6991	0.6884
Ratchet-Initial Anchor, w=1, e=0.05	0.5297	0.6630	0.6764
Ratchet-Initial Anchor, w=1, e=0.20	0.4914	0.5847	0.5911
Ratchet-Mov. Anchor (20,1)	0.6692	0.7211	0.6976
Ratchet-Mov. Anchor (20,1, e=0.05)	0.5567	0.6675	0.6770
Ratchet-Mov. Anchor (20,1, e=0.20)	0.4879	0.5757	0.5837

possibly therefore to uncover a new 'gbest'. In essence, an errorful assessment of payoff may allow a short-term 'wrong-way' move (one which temporarily reduces an organization's payoff), but which in the longer-term leads to higher payoffs. Conversely, it could lead to the loss of a promising but under-developed strategy, if an organization is led away from a promising part of the strategic landscape by an incorrect payoff assessment. To examine the impact of errorful payoff assessment, results are reported for the Ratchet PSA-No Anchor, for values of the error ratio of 0.05 and 0.20. Examining Table 1 shows that these produce lower results (statistically significant at 5%) than the error-free case. As the size of the error ratio increases, the 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.

The experiments implementing strategic inertia (initial anchor with weight=1, and moving anchor on a 20-lag period with weight=1) for each of the three values of the error ratio generally indicate that the addition of strategic inertia enhances average populational fitness. Comparing the results for the two forms of strategic inertia indicates that a moving anchor performs better when organizations can make error-free assessments of the payoff to potential strategies, but when these payoffs are subject to error neither form of strategic inertia clearly dominates the other in terms of producing the higher average populational fitness. In summary, the results for the static landscape scenario do not support a hypothesis that errorful assessments of payoffs to potential strategies are beneficial for populations of organizations. In addition, the results broadly suggest that strategic inertia, when combined with an election operator, produces higher average populational fitness, but the benefits of this combination dissipates when the level of error in assessing *ex-ante* payoffs gets large.

4.2 Dynamic Landscapes

The real world is rarely static, and changes in the environment can trigger adaptive behavior by agents in a system [1]. When the strategic landscape is wholly or partially respecified, the benefits of past strategic learning by organizations is eroded. In this simulation, two specific scenarios are examined. Table 2 provides the results for the case where a single dimension of the NK landscape is respecified in each iteration of the algorithm with a probability of P=0.00025, and also the results for the case where the entire NK landscape is respecified with the same probability (Figures 1 and 2 provides a graphic of the adaptive trajectories of each search heuristic for K=4 and K=10, on both the static and dynamic 'full respecification' landscapes, and demonstrate that the simulation results are not qualitatively sensitive to the choice of end-point). Qualitatively, the results from both scenarios are similar to those obtained on the static landscape. The basic PSA does not perform any better than random search. Supplementing the basic PSA with the ratchet mechanism leads to a significant improvement in populational fitness, with a further improvement in fitness occurring when the ratchet is combined with an anchor. Adding errorful assessment of the payoffs to potential strategies leads to a deterioration in populational fitnesses as the error ratio increases, but as for the static landscape case, the addition of strategic inertia generally enhances average populational fitness for lower levels of error in assessing payoffs. Comparing the results for the two forms of strategic inertia indicates that a moving anchor performs better when organizations can make error-free assessments of the payoff to potential strategies, but when these payoffs are subject to error, neither form of strategic inertia dominates the other in terms of producing the higher average populational fitness.

Table 2. Average fitness after 5,000 iterations, one dimension (entire landscape) respecified stochastically.

Algorithm	**Fitness**		
	(N=96, K=0)	**(N=96, K=4)**	**(N=96, K=10)**
Basic PSA	0.4667 (0.4761)	0.4987 (0.4886)	0.4955 (0.4961)
Ratchet PSA-No Anchor	0.5783 (0.5877)	0.6859 (0.6802)	0.6808 (0.6754)
R-No Anchor, e=0.05	0.4927 (0.5143)	0.6458 (0.6309)	0.6673 (0.6568)
R-No Anchor, e=0.20	0.4945 (0.5027)	0.5769 (0.5672)	0.5810 (0.5779)
R-Initial Anchor, w=1	0.6207 (0.6187)	0.6994 (0.6874)	0.6895 (0.6764)
R-Initial Anchor, w=1, e=0.05	0.5390 (0.5612)	0.6636 (0.6551)	0.6766 (0.6599)
R-Initial Anchor, w=1, e=0.20	0.4914 (0.5045)	0.5848 (0.5819)	0.5881 (0.5873)
R-Mov. Anchor (20,1)	0.6689 (0.6575)	0.7193 (0.7152)	0.6974 (0.6819)
R- Mov. Anchor (20,1, e=0.05)	0.5612 (0.5613)	0.6679 (0.6622)	0.6814 (0.6670)
R- Mov. Anchor (20,1, e=0.20)	0.4926 (0.5004)	0.5785 (0.5689)	0.5830 (0.5810)

5 Conclusions

In this paper, a novel synthesis of a strategic landscape defined using the NK model, and a Particle Swarm metaphor is used to model the strategic adaption of organizations. The results suggest that a degree of strategic inertia, in the presence of an election operator, can generally assist rather than hamper the adaptive efforts of populations of organizations in static and slowly changing strategic environments, when organizations can accurately assess payoffs to future strategies. The results also suggest that errorful assessments of the payoffs

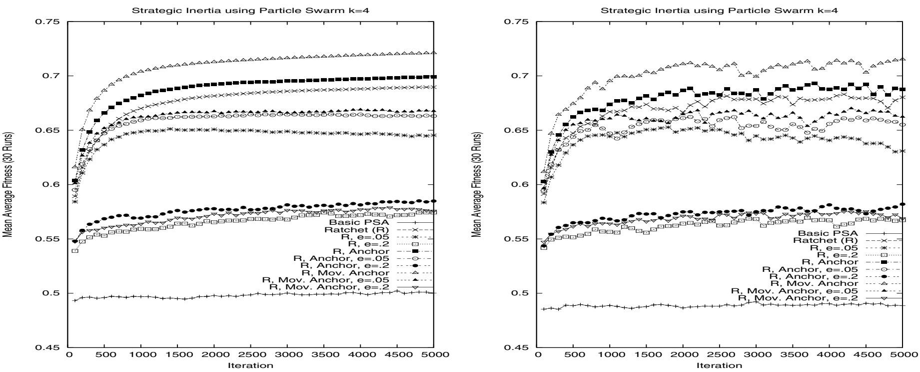

Fig. 1. Plot of the mean average fitness on the static (left) and dynamic (right) landscape where k=4.

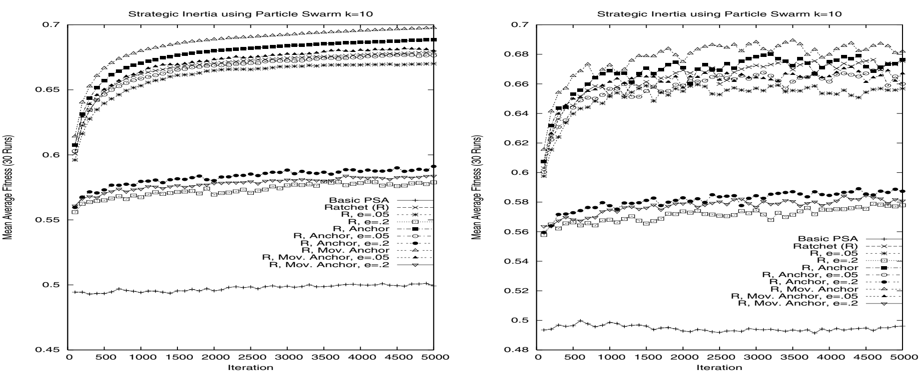

Fig. 2. Plot of the mean average fitness on the static (left) and dynamic (right) landscape where k=10.

to potential strategies leads to a deterioration in populational fitnesses as the error ratio increases. It is also noted that despite the claim for the importance of social learning in populations of agents, the results suggest that social learning is not always enough, unless learnt lessons can be maintained by means of an election mechanism.

No search heuristic will perform equally on all landscapes and across all scales of environmental change. Hence, we acknowledge that the results of this study will not generalize to all possible forms of landscape, and all rates of environmental change. The affect of gbest, pbest and inertia terms, 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 landscape, they may find it impossible 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 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. Finally, it is noted that the concept of inertia or '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 to incorporate inertia may prove useful beyond this study.

References

1. Blackwell, T. (2003). Swarms in Dynamic Environments, *Proceedings of GECCO 2003*, Lecture Notes in Computer Science (2723), Springer-Verlag, Berlin, pp. 1-12.
2. Gavetti, G. and Levinthal, D. (2000). Looking Forward and Looking Backward: Cognitive and Experiential Search, *Administrative Science Quarterly*, 45:113-137.
3. Kauffman, S. and Levin, S. (1987). Towards a General Theory of Adaptive Walks on Rugged Landscapes, *Journal of Theoretical Biology*, 128:11-45.
4. Kauffman, S. (1993). *The Origins of Order*, Oxford,England: Oxford University Press.
5. Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization, *Proceedings of the IEEE International Conference on Neural Networks*, December 1995, pp.1942-1948.
6. Kennedy, J. (1997). The particle swarm: Social adaptation of knowledge, *Proceedings of the International Conference on Evolutionary Computation*, pp. 303-308: IEEE Press.
7. Kennedy, J. (1999). Minds and Cultures: Particle Swam Implications for Beings in Sociocognitive Space, *Adaptive Behavior*, 7(3/4):269-288.
8. Kennedy, J., Eberhart, R. and Shi, Y. (2001). *Swarm Intelligence*, San Mateo, California: Morgan Kauffman.
9. Kitts, B., Edvinsson, L. and Beding, T. (2001). Intellectual capital: from intangible assets to fitness landscapes, *Expert Systems with Applications*, 20:35-50.
10. Levinthal, D. (1997). Adaptation on Rugged Landscapes, *Management Science*, 43(7):934-950.
11. Porter, M. (1985). *Competitive Advantage:Creating and Sustaining Superior Performance*, New York: The Free Press.
12. Porter, M. (1996). What is Strategy?, *Harvard Business Review*, Nov-Dec, 61-78.
13. Rivkin, J. (2000). Imitation of Complex Strategies, *Management Science*, 46(6):824- 844.
14. Wright, S. (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution, *Proceedings of the Sixth International Congress on Genetics*, 1:356-366.