PRELIMINARY EXPERIMENTS WITH IIDLE

Technical Report 9-01

Jason Brownlee
jbrownlee@ict.swin.edu.au

PhD Candidate
Master of Information Technology, Swinburne University of Technology, 2004
Bachelor of Applied Science, Computing, Swinburne University of Technology, 2002

Centre for Intelligent Systems and Complex Processes
Faculty of Information and Communication Technologies
Swinburne University of Technology
Melbourne, Victoria, Australia

September 2005

Abstract

The mammalian acquired immune system is a robust and powerful information processing system that demonstrates features such as decentralised control, parallel processing, adaptation, and learning through experience. Artificial Immune Systems (AIS) are a class of machine-learning algorithms that are imbued with some principles of the immune system, and attempt to take advantage of some of the benefits of the features of the immune system to tackle difficult problem domains. A novel artificial immune system called IIDLE – the immunological inspired distributed learning environment has been introduced previously in regard to the techniques inspiration, conceptualisation and rudimentary architecture and processes. IIDLE is inspired by the spatially distributed nature of the acquired immune system, and the clonal selection principle that describes how the immune system learns and adapts in response to stimulation from its environment. In this work, the IIDLE platform is preliminarily put to the test on simple combinatorial optimisation (TSP) and function optimisation problems (Schwefel's function). A number of experiments are performed to gain an initial understanding of the suitability and applicability of IIDLE in the context of dynamic function optimisation, multiple constraints, complementing objective functions and human interactive search. Some interesting and other surprising results are achieved that perhaps provide a first indication of the potential of this novel machine learning system. Further work is required, and a number of exciting additional experiments are proposed.

Table of Contents

1 INTRODUCTION **5**

2 COMBINATORIAL OPTIMISATION **6**

2.1 EXPERIMENT 1 – MULTIPLE CONSTRAINTS 7
2.1.1 Observations and Discussion *8*
2.2 EXPERIMENT 2 – MULTIPLE SEARCH STRATEGIES 9
2.2.1 Observations and Discussion *9*
2.3 EXPERIMENT 3 – HUMAN INTERACTION 10
2.3.1 Observations and Discussion *10*

3 FUNCTION OPTIMISATION **11**

3.1 EXPERIMENT 4 – SPLIT AXIS 13
3.1.1 Observations and Discussion *14*
3.2 EXPERIMENT 5 – AXIS FLIPPING 15
3.2.1 Observations and Discussion *15*
3.3 EXPERIMENT 6 – DYNAMIC FUNCTION OPTIMISATION 16
3.3.1 Observations and Discussion *17*

4 CONCLUSIONS AND FURTHER WORK **18**

5 BIBLIOGRAPHY **21**

6 APPENDIX A - CONFIGURATIONS **21**

6.1 EXPERIMENT 1 CONFIGURATION 21
6.2 EXPERIMENT 2 CONFIGURATION 22
6.3 EXPERIMENT 3 CONFIGURATION 22
6.4 EXPERIMENT 4 CONFIGURATION 23
6.5 EXPERIMENT 5 CONFIGURATION 23
6.6 EXPERIMENT 6 CONFIGURATION 23

List of Figures

FIGURE 1 - OPTIMAL SOLUTION FOR THE BERLIN-52 SOLUTION (TOUR LENGTH OF 7544.36 UNITS) 6
FIGURE 2 - GRAPH SHOWS THE PERFORMANCE OF MULTIPLE STIMULATION HEURISTICS ON BERLIN-52. .. 8
FIGURE 3 - GRAPH SHOWS THE PERFORMANCE OF IIDLE WHEN USED WITH ACO, LOCAL SEARCH AND BOTH STRATEGIES COMBINED AT THE SAME TIME. 10
FIGURE 4 - SCREENSHOT OF THE HUMAN-INTERACTIVE INTERFACE FOR THE TSP EXPERIMENT ON IIDLE, SHOWING NINE DIFFERENT CANDIDATE SOLUTIONS 11
FIGURE 5 - SCREENSHOT OF THE HUMAN-INTERACTIVE INTERFACE FOR THE TSP EXPERIMENT ON IIDLE, SHOWING FOUR DIFFERENT CANDIDATE SOLUTIONS WITH HUMAN FEEDBACK (RED FOR POSITIVE, BLUE FOR NEGATIVE) 11
FIGURE 6 - SHOWS A TWO-DIMENSIONAL CONTOUR PLOT OF SCHWEFEL'S FUNCTION 12
FIGURE 7 - SHOWS A THREE-DIMENSIONAL CONTOUR PLOT OF SCHWEFEL'S FUNCTION 12
FIGURE 8 - A SCREENSHOT OF THE CONFIGURED IIDLE SYSTEM SHOWING TWO PARTITIONS (RED AND BLUE). CIRCLE SIZE OF EACH LOCALITY REPRESENTS THE NUMBER UNITS IT CONTAINS IN ITS TAIL. 13
FIGURE 9 - PLOT OF THE MODIFIED OBJECTIVE FUNCTION. STARTING FROM THE LEFT, A) THE FUNCTION IN TWO DIMENSIONS, B) THE FIRST DIMENSION EXPOSED TO THE FIRST PARTITION, C) THE SECOND DIMENSION EXPOSED TO THE SECOND PARTITION 14
FIGURE 10 - SCREENSHOT OF FINAL POPULATION DISTRIBUTION. THE PLOT SHOWS UNITS WITHIN EACH PARTITION (RED AND BLUE) SPREAD ALONG THE AXIS OF EACH INDIVIDUAL OBJECTIVE FUNCTIONS MINIMA. 14
FIGURE 11 - SCREENSHOT OF THE OBJECTIVE FUNCTIONS AT THE MIDPOINT OF A CYCLE (COEFFICIENT OF 0.5). THE IMAGE SHOWS THE EQUALITY OF THE TWO PARTITIONS IN TERMS OF EVALUATION AT THE MIDPOINT. 16
FIGURE 12 - SURFACE PLOT OF THE POPULATION DISTRIBUTION AT THE END OF AN EXPERIMENTAL RUN. CLEARLY INDICATES THE SYSTEMS FOCUS AT THE POINT OF INTERSECTION BETWEEN OBJECTIVE FUNCTION MINIMA AS PREDICTED. 16

FIGURE 13 - THREE SCREENSHOTS OF THE UNIT DISTRIBUTION OVER THE COURSE OF A SINGLE RUN. SHOWS VARIOUS STAGES OF THE DYNAMIC FUNCTION'S CYCLE AND THE UNITS OF THE SYSTEM TRACKING THE GLOBAL MINIMA (DARK BLUE AND PURPLE) 18

List of Equations

EQUATION 1 - THE TOTAL NUMBER OF SOLUTIONS FOR A SYMMETRICAL TSP 6
EQUATION 2 - SCHWEFEL'S FUNCTION 12
EQUATION 3 - SCHWEFEL'S FUNCTION 13
EQUATION 4 – PARTITIONED SCHWEFEL'S FUNCTION 15
EQUATION 5 – MODIFIED INPUTS FOR A DYNAMIC SCHWEFEL'S FUNCTION 17

1 Introduction

The mammalian immune system is an intriguing natural defence system that has been shown to be capable of learning, memory, and adaptation. Conceptually, the active acquired immune system can be taken as a spatially distributed yet circulating (immuno-surveillance) and heterogeneous population of specialised discrete units that provide a homogeneous defence against external pathogenic material. The clonal selection principle is a theory that is core to the understanding of the acquired immune system and is accepted as the most plausible explanation for the systems learning, adaptive and memory behaviours. The Immunological Inspired Distributed Learning Environment (IIDLE) is a machine-learning platform inspired by the spatially distributed nature of the acquired immune system and the clonal selection principle. A good introduction to artificial immune systems (AIS) and the biological immune systems is provide by de Castro and Timmis [6]. The inspiration, conceptualisation, architecture, and processes of IIDLE are described by Brownlee [5].

IIDLE is a novel platform for machine learning and although the inspiration and conceptualisation of the technique have been described, there is as yet no work providing indication as to the applicability and suitability of the platform in regard to configurations and or general problem domains. The intent of this work is to provide a preliminary investigation of IIDLE in the context of a widely used and well-understood combinatorial optimisation problem (TSP), and a canonical benchmark function optimisation problem to assess the search and optimisation behaviour of an optimisation technique.

Six different experiments are performed and discussed that reflect the properties under investigation in IIDLE. Three of the experiments are performed on a specific case of a TSP combinatorial optimisation problem (Berlin-52), the remaining three are performed on a specific case function optimisation problem (Schwefel's function).

No.	**Experiment Title**
1	Combinatorial Optimisation – Multiple Constraints
2	Combinatorial Optimisation – Multiple Search Strategies
3	Combinatorial Optimisation – Human Interaction
4	Function Optimisation – Split Axis
5	Function Optimisation – Axis Flipping
6	Function Optimisation – Dynamic Function

Specifically this work seeks to assess five properties of IIDLE in the context of the specified problem domains, as follows:

1. The potential of IIDLE on multiple objective or multiple constraint problems where competing or complementing objectives are exploited to direct the search process. (exp. 1, 4, 5)
2. The potential for IIDLE on a problem domain that posses individual constraints that dynamically change, providing a moving optimisation target for each constraint used. (exp. 5, 6)

3. The potential of the IIDLE framework on a problem domain when multiple competing or complementing search strategies are used within the spatially distributed structure in parallel. (exp. 2)
4. The potential of IIDLE as a platform for human-interactive machine learning such as search or optimisation tasks. (exp. 3)
5. The potential of IIDLE to achieving niching like effects when the spatially distributed architecture is partitioned in regards to constraints, objective function and search strategy. (exp. 1, 2, 4, 5)

The experimental results will not reflect the optimal efficacy or efficiency of IIDLE on the specific experimental setup as this is not the primary interest in this work. Instead, the focus will be on general system performance and behaviours as this is a preliminary work to initially gauge areas of further research that may be more or less promising for the IIDLE platform.

2 Combinatorial Optimisation

Combinatorial optimisation is a search process for locating set of feasible solutions (feasible concerning some quality measure) in which the search domain is discrete or can be made discrete. A combinatorial optimisation problem commonly used as a benchmark for search algorithms is the travelling salesman problem (TSP) [3]. TSP is a useful benchmark problem because it belongs to a class of problems that is very hard to solve (NP-complete). The TSP is relevant to transportation problems, logistic problems, as well as hole-drilling for components on printed electronic circuit boards (PCB).

The specific type of TSP selected for these experiments on IIDLE are referred to as two-dimensional symmetric Euclidean TSP's. The problem is defined as a number of vertices with Cartesian or polar coordinates on a plane. The graph is assumed to be fully connected, and each city must be visited once, thus the problem is described as minimising the distance travelled or overall tour length. A specific well understood problem instance was selected for experimentation called the Berlin-52 problem named after the datasets origin from a map of Berlin [2]. The problem is a relatively small TSP with a known optimal solution, and provides a suitable non-trivial combinatorial optimisation problem that is easy to visualise and interpret results.

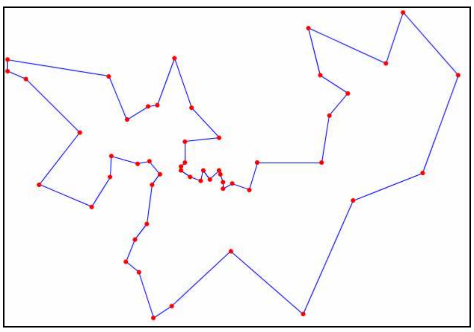

Figure 1 - Optimal solution for the Berlin-52 solution (tour length of 7544.36 units)

$$totalSolutions = \frac{(totalCities - 1)!}{2}$$

Equation 1 - The total number of solutions for a symmetrical TSP

A permutation (city-list) based representation was selected for units in the IIDLE system. This is the most common form of representation for this type of combinatorial problem. A widely used course-grained (perhaps global) optimisation technique known for quickly locating good approximate solutions for TSP is the Ant Colony Optimisation algorithm (ACO) [7]. An implementation of this technique called Ant Systems (AS) was employed as the proliferation strategy, although the technique was modified to exploit the population-based history in IIDLE rather than the conventional pheromone-based search history used in all ACO techniques.

2.1 Experiment 1 – Multiple Constraints

It is standard practice to use a nearest-neighbour heuristic when constructing candidate solutions in the ACO algorithm, when applied to the TSP. The heuristic is based on the premise that an optimal tour will likely consist of many nearest neighbour connections, although this heuristic alone is not expected to lead to a suitably optimal solution. The spatially distributed nature of IIDLE permits the configuration across the system to be varied. The intent of this experiment is to investigate the viability of varying the manner in which solutions are evaluated (quality measure) across the IIDLE data structure.

The conventional AS algorithm uses tour length as a candidate solution quality measure, which in turn influences the contribution each candidate solution has on the search. The summation of candidate solution contribution is combined with the problem-specific heuristic (already described) to produce additional candidate solutions (re-sample the search space). This experiment varies the quality measure assigned to candidate solutions, and thus their contribution and ultimately the directedness and focus of the search process. Two additional different candidate solution measures are used, that are expected to be as useful as the traditional tour length (to be minimised) at assessing the quality of a tour, they are; the number of intersections in a tour (to be minimised), and the number of nearest neighbour connections in a tour (to be maximised). These two additional measures (constraints) were selected as alone they are likely to provide suitable solutions, and they are expected to complement each other when combined.

Five experiments are performed using combinations of the selected quality measures. Each of the three measures is tested individually, and then in the remaining two experiments all three are used at the same time within the system. In the case of the forth configuration, the system is partitioned into three segments where the units in each partition are exposed to a single type of quality feedback (stimulation). This configuration is intended to test the whether there is an implicit niching of solutions within each partition facilitated by the movement process. The fifth configuration permits all three-stimulation types access to the entire un-partitioned data structure, comparing the use of all three-quality measures, and contrasting the effect of partitioning with configuration four.

No.	Experimental Configuration
1	Tour Length (minimised)
2	Tour Intersections (minimised)
3	Nearest Neighbour Connections (maximised)
4	All three partitioned
5	All three mixed

In all five configuration setups, the system is configured consistently. The intent of the experiment was not to demonstrate the efficacy or efficiency of the system on TSP, and thus reasonable default configurations were selected and not optimised. In particular, the amount of stimulation (external information or fitness evaluation) the system is exposed in the case of multiple stimulation types was kept constant at an overall system level. The ant systems variant used was configured to pay more attention to the history element than the heuristic element as is the canonical configuration when constructing new solutions, see Appendix A for complete configuration details.

2.1.1 Observations and Discussion

Three things were surprising about the results. The first is that the number of intersections as a quality measure provided results practically on par with those when using tour length as the quality measure. Perhaps with further investigation the number of intersections could be used as an alternative quality measure and or construction heuristic in ACO algorithms. The following figure shows the experimental results with mean and standard deviation tour length (result of interest) for all five experimental setups. It should be noted that the difference between all sets of results is less 300 units, which is minor, although all tests were repeated 100 times to increase statistical significance.

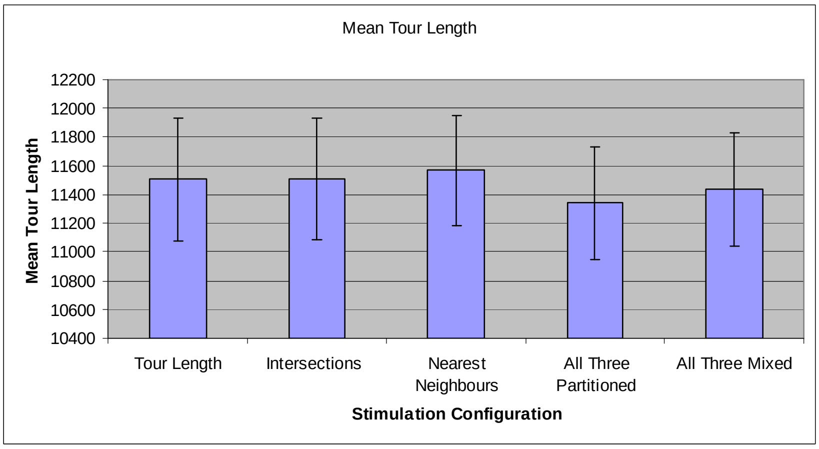

Figure 2 - Graph shows the performance of multiple stimulation heuristics on Berlin-52.

The second interesting point about the results is that in both cases when all three quality measures are used at the same time in the system, better performance is achieved than any of three used independently. This is likely because of the additional problem-specific information the system is exposed to. This not only highlights the potential benefit of combining quality measures and perhaps constraints, but also the potential use of multiple quality measures and or heuristics in conventional ACO algorithms.

Finally, the results show that when the system is partitioned with regard to the three quality measures, it performs slightly better than when all three quality measures are applied uniformly across the spatial structure of IIDLE. This intriguing result implies that given the size of each partition (~16 localities) that the system is capable of buffeting the effects of movement to niche solutions within each partition, although, the complementary quality measures (in regards to directedness of search) are capable of being exploited by the movement process at the interface of each partition (where one partition connects to another). Clearly, this implied implicit niching and information sharing effect merits further investigation.

2.2 Experiment 2 – Multiple Search Strategies

It is conventional to run a local search technique after completing a more course search such as with an ant-based algorithm. This can be considered a sequential form of hybrid search that uses two different yet complementing search techniques. The intent of this experiment is to preliminarily investigate IIDLE's capability of executing a course and a fine-grained search (hybrid search) in parallel. Specifically this experiment briefly investigates the use of two different search strategies embedded in the framework in parallel – parallel hybrid search with respect to a modified version of the ant systems algorithm, and a simple mutation based local search.

This experiment consists of four configuration setups that first test each algorithm (course and fine) independently, and then combined. As in the previous experiment, the complementing search elements are tested in a partitioned and mixed configuration to contrast the proposed implicit niching expectation. Although in this case it is the proliferation (*search*) strategy which is varied, rather than the stimulation (*evaluation*) strategy.

No.	Experimental Configuration
1	ACO
2	Mutation-based local search
3	Both partitioned
4	Both combined

2.2.1 Observations and Discussion

The results for this experiment were surprising. When combining the two search strategies in IIDLE, performance slightly decreased in regards to efficacy (tour length), rather than increase as expected. The following figure provides a summary of results showing mean and standard deviation of tour length (efficacy measure of interest) for each experimental setup.

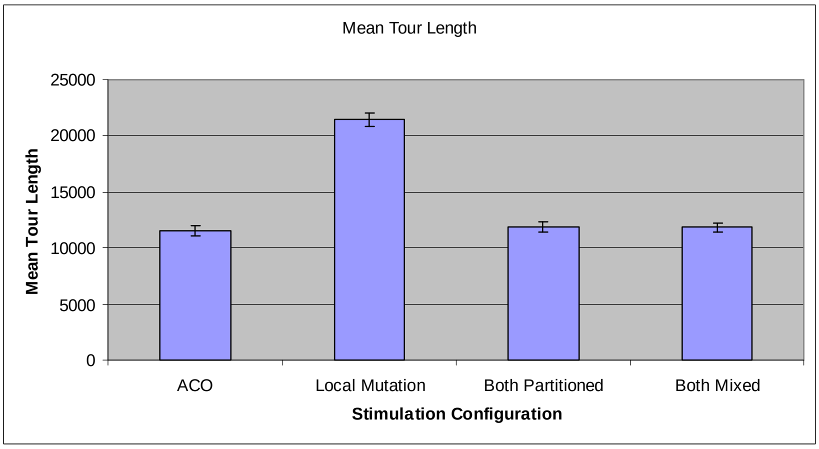

Figure 3 - Graph shows the performance of IIDLE when used with ACO, local search and both strategies combined at the same time.

The unexpected poor result may have been caused by the poor selection of configuration parameters, or perhaps the selection of a poor local search technique. It is speculated that the combination of complementing search strategies in a parallel configuration will result in improved results, although the difficulty is demonstrated to occur in the selection of techniques that complement each other in a parallel configuration. To effectively evaluate IIDLE in this promising area of research, a lot more experimentation is required with different search approaches on varied problem domains. Ultimately, it is possible that the utility of parallel-hybrid search will be less than that of the widely accepted and exploited sequential hybrid search approach.

2.3 Experiment 3 – Human Interaction

Human interactive search is an area of search and optimisation research that exploits human feedback to guide a search process. The IIDLE is amenable to such ad hoc, sporadic, and specialised feedback given the system is configured to respond to external stimuli with internal activity in a parallel and decentralised manner. The TSP is a widely used combinatorial optimisation problem as discussed, and this experiment seeks to investigate whether IIDLE can be configured to exploit human feedback whilst optimising a tour for a TSP.

For this experiment, a specialised interactive TSP module was created for IIDLE that permitted the user to be shown one or more candidate solutions and comment on their quality positively or negatively. The module is capable of showing, one, two, four, or nine candidate TSP solutions to the user for a set period (a number of seconds). The user responds to the visual candidate solutions in three ways; doing nothing (indifference), left clicking for positive feedback (up to +5 points), and right clicking for negative feedback (up to -5 points), one point per mouse click. The module shows an entire tail of candidate solutions to a user (a set number per screen), after which time selection and proliferation strategies are executed as in a conventional expansion process in IIDLE.

The system was configured to use a the modified ant systems algorithm as the proliferation strategy for the same reasons as in it was used for previous TSP experiments, that is a widely accepted as an effective approximation technique on this problem.

2.3.1 Observations and Discussion

The intent of the experiment was to gauge the impact the user had influencing the direction of a search over an extended time. A single human user completed ten runs of an extended interactive period. The results from the human interactive experimental setup were compared to the same setup without any user intervention (ACO without a quality measure). The results were not statistically different between the two experimental setups, and thus were omitted from this work. The following figures provide an indication of the human-interactive interface to the search process used during the experiment.

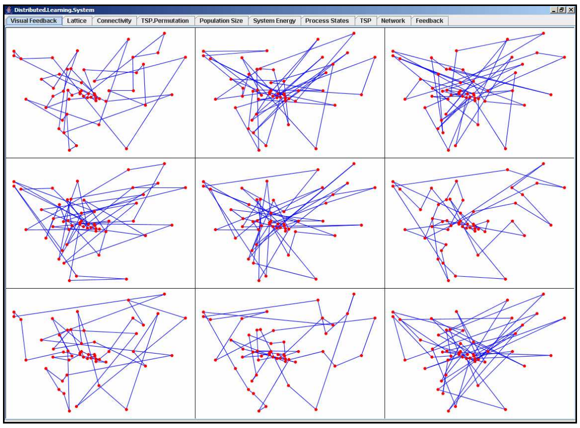

Figure 4 - Screenshot of the human-interactive interface for the TSP experiment on IIDLE, showing nine different candidate solutions

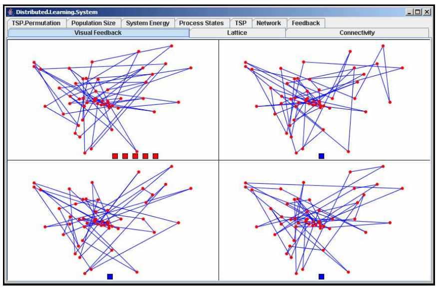

Figure 5 - Screenshot of the human-interactive interface for the TSP experiment on IIDLE, showing four different candidate solutions with human feedback (red for positive, blue for negative)

It is interesting to note a few observed behaviours whilst performing the interactive experiment. Firstly, feedback on candidate solutions was in response to solution "*messiness*" which although vague, is an interesting and human property of optimal TSP solutions. The second point, is that with concerted effort, the human operator was able to "*clean-up*" selected regions of the tour's proposed by the system, specifically the right of the dataset where the points are spaced out. Whether this was a real effect of influencing the search is interesting (and somewhat likely), and remains an area for further research. The TSP it seems may not have been the best choose to demonstrate the human-interactive capabilities of IIDLE, and thus to gain a better understanding of IIDLE's suitability to this type of problem, additional problem domains need to be investigated, perhaps taken from the field of interactive evolutionary computation.

3 Function Optimisation

Function optimisation is a search process that given a defined objective function and inputs to the objective function, seeks to find extrema (minima or maxima) of that function. The process involves optimising the function parameters that typically have complex and non-linear interrelationships. The general problem of function optimisation is representative of a large number of specific problem cases from domains such as bioinformatics, operations research, and computational physics.

It is common practice to optimise canonical test objective functions when investigating and testing search and optimisation algorithms for application in the area of function optimisation. Test functions are typically easy to define algebraically and typically easy to scale in terms of dimensionality. Test functions are designed to possess specific characteristics to evaluate the behaviour of an algorithm concerning the selected characteristic, such as modality (number of desired or deceptive extrema), dimensionality (scalability), or general performance on a well-understood surface. Lower dimensionality functions (two or three-dimensional for example) are useful not for measuring the efficiency or efficacy of the technique on the objective function, rather to evaluate (measure and visualise) and analyse the "learning" or sampling behaviour of the technique.

A well-known function proposed by and named after its designer Schwefel [4] was selected as a base function to investigate the behaviour and performance of IIDLE concerning online (dynamic) function optimisation. The Schwefel function is multimodal minimisation objective function, possessing many deceptive (false) minima that can disrupt a search strategy. Further, the function was selected because its objective surface is non-trivial, and because the shape of the function is the same in all dimensions, meaning that it is easy to manipulate and changes are predictable.

$$q = \sum_{n}^{i=1} -x_i \cdot \sin(\sqrt{|x_i|})$$

Equation 2 - Schwefel's function

Where n represents the number of dimensions (in this case two), and $x \in [-500,500]$.

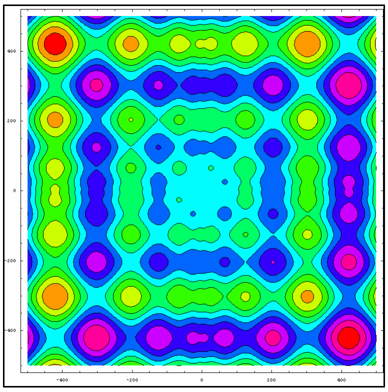

Figure 6 - Shows a two-dimensional contour plot of Schwefel's function

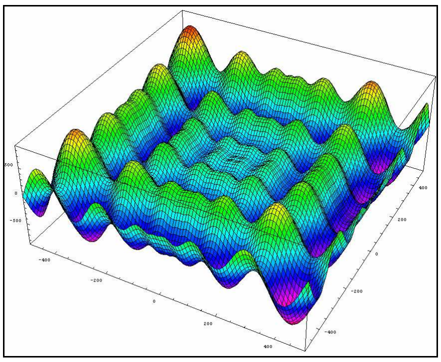

Figure 7 - Shows a three-dimensional contour plot of Schwefel's function

The function takes continuos input values, and thus the representation of a candidate solution (set of parameters) can be taken as the continuous variables. An alternative representation common in the field of evolutionary computation, specifically genetic algorithms is to represent each parameter in base-2 as a string of bits. This representation was selected due to its simplicity to encode in software. Further, this representation was selected because the proliferation strategy employed for the proposed tests was the canonical or simple genetic algorithm [1], which is known to be an effective global optimisation technique. All function optimisation experiments used the selected test function (Schwefel's function) in two dimensions for ease of visualisation, and the majority of observations were taken from the various graphs and plots provided by the IIDLE software. The input parameters for the function (x, y) where represented with 32-bits each, providing a bitstring representation of length 64. The system configurations for each experiment are located in Appendix A.

3.1 Experiment 4 – Split Axis

The intent of this experiment is to evaluate the sampling behaviour of IIDLE when the objective function is split into two distinct objective functions and spatially disparate elements of the system are exposed to these distinct objective functions. The objective function was adjusted such that the second dimension was inverted. This was done so that the global optima of the function would be placed in the lower left-hand corner of the two-dimensional surface, thus causing the input value to be different for each of the two axes (large value for the x-axis, a small value for the y-axis). This change is required for experiment five (flipping axes), and was made here for consistency and contrast between the two experiments (with and without flipping axes in a partitioned system).

$$q = \left(-x \cdot \sin(\sqrt{|x|})\right) - \left(-y \cdot \sin(\sqrt{|y|})\right)$$

Equation 3 – Modified Schwefel's function

Further, the localities of the system were partitioned into two groups. In the first group (or red group) units of the affected localities were evaluated only based upon the first input value (x), and in the second group (or blue group), the affected localities were evaluated based solely upon the units second input value (y). The following screenshot of the IIDLE system demonstrates the partitioning of the localities.

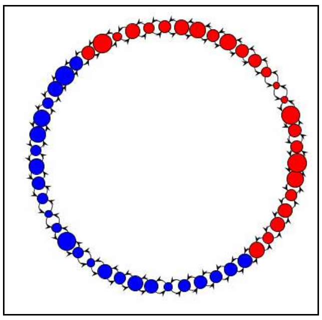

Figure 8 - A screenshot of the configured IIDLE system showing two partitions (red and blue). Circle size of each locality represents the number units it contains in its tail.

In an attempt to further clarify the experiment setup, the following diagram provides an overview of the objective functions used in each partition. Starting from the left, the first plot shows the modified objective function in two dimensions. The remaining two plots show the first parameter (x) objective function for the first partition and the second parameter (y) objective function applied to the second partition. It is clear that the there is an interaction of the minima in the two individual functions in the lower right of the surface. A sample found in this location (large value on the x-axis, small value on the y-axis), is capable of existing successfully in either partition.

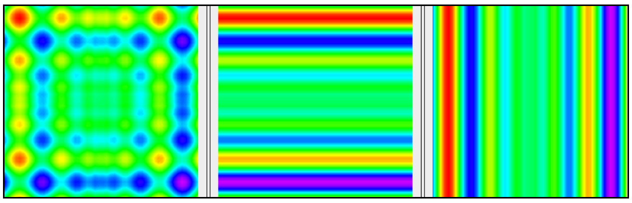

Figure 9 - Plot of the modified objective function. Starting from the left, a) the function in two dimensions, b) the first dimension exposed to the first partition, c) the second dimension exposed to the second partition

Given the experimental setup, the intent is to observe whether the movement process upon the units alone is sufficient to encourage the system to focus attention on the minima intersection or instead to maintain niche solutions within each of the two partitions.

3.1.1 Observations and Discussion

The experiment was repeated a number of times and the results were consistent. Instead of focusing the search at the point of minima intersection between the two objective functions, the system maintained niche solutions within each partition. The following screenshot captured at the end of execution of one of the experiments demonstrates the system maintaining samples (units) along the axis of each individual objective functions minima.

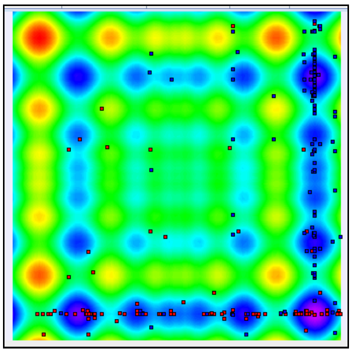

Figure 10 - Screenshot of final population distribution. The plot shows units within each partition (red and blue) spread along the axis of each individual objective functions minima.

This is an interesting systemic behaviour. Although the movement was enforced at each locality with a probability of one (one unit will be moved from each locality to a neighbour per execution), the system was able to maintain implicit solution niches. This is likely due to the moderate number of localities used (50 overall, 25 per partition) that likely provided a buffeting effect between the cores of each partition. This implicit behaviour may be useful in multi-objective domains in which one system is desired that is capable of focusing on each objective independently. The niching behaviour would be useful in this situation if it could be shown that the information sharing between the partitions was of benefit to the search process within each partition. This remains an area for further investigation.

3.2 Experiment 5 – Axis Flipping

One of the speculated outcomes for the previous experiment was that the system would focus the overall global search on the intersection of the minima of each individual objective function. The intent of this experiment is to encourage the system to focus the search on the elusive minima intersection without encoding domain specific information. This outcome is interesting and desirable as it provides a base understanding of the systems behaviour for search and optimisation problems that have multiple objectives and or constraints in which a global solution is required that best (perhaps a good approximation) satisfies each of the encoded objectives.

The objective function was modified once more such that the individual partition objective functions change over time. As in the previous experiment, each objective function evaluates a single input parameter (x or y), though in this experiment the function changes over time from the x-axis objective function to the y-axis objective function, and back again. This was implemented by adjusting the function such that the scoring provided from each function was taken as a combination of each axis using a coefficient that varied between zero and one and back to zero in a cycle of a set number of simulation time steps. The following provides the equations for this experiment.

$$
\begin{aligned}
tx &= \left(-x \cdot \sin(\sqrt{|x|})\right) \\
ty &= -\left(-y \cdot \sin(\sqrt{|y|})\right) \\
q1 &= \alpha \cdot tx + (1-\alpha) \cdot ty \\
q2 &= \alpha \cdot ty + (1-\alpha) \cdot tx
\end{aligned}
$$

Equation 4 – Partitioned Schwefel's function

Where α is the coefficient calculated as a ratio of the current positing in a cycle from the total number of algorithm iterations configured in the cycle. The coefficient does not wrap around upon reaching the limits (zero and one), instead it reverses direction providing a smooth harmonic motion around the mid point (0.5).

3.2.1 Observations and Discussion

Observing a plot of the objective functions over the course of a simulation provided an understanding of the behaviour of the objective functions and thus the selective pressures the units in each partition are exposed to. At the beginning of a cycle (coefficient of zero), the objective functions for each partition resemble those from the previous experiment (see Figure 9). At the mid point of the cycle (coefficient of 0.5), the objective functions of each partition resemble a scaled version of the standard (although adjusted) Schwefel’s function (see Figure 11). Finally at the end of a cycle (coefficient of one), the objective functions between the first and second partitions visually appear to have reversed. At this point, the cycle is reversed.

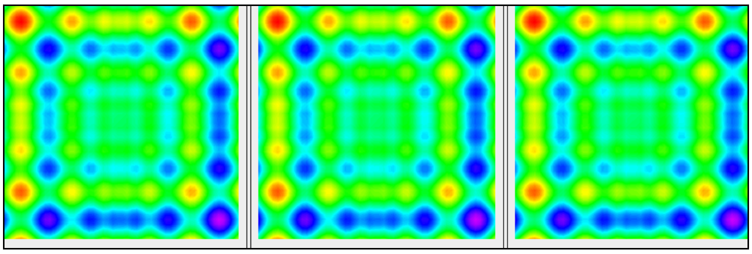

Figure 11 - Screenshot of the objective functions at the midpoint of a cycle (coefficient of 0.5). The image shows the equality of the two partitions in terms of evaluation at the midpoint.

The result of the cycled changes to the objective functions resulted in the predicted global search behaviour. At the completion of the first cycle, the system quickly focused the search attention of both partitions to the region where the minima of each individual objective function intersected. The following figure shows the final population distribution at the end of a run, clearly showing the clustering of units across the entire system at the lower right of the surface plot.

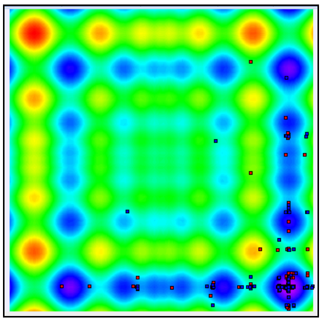

Figure 12 - Surface plot of the population distribution at the end of an experimental run. Clearly indicates the systems focus at the point of intersection between objective function minima as predicted.

Knowing that there is an intersection between the optima of each constraint is prior knowledge about the problem domain that is likely to be unavailable for more realistic problem domains. The result demonstrates that the system can be persuaded to (at least) seek a compromise between multiple (in this case two) objectives, if the objectives are varied across the spatial structure of localities. It is expected that a similar though perhaps less smooth effect could be achieved by applying the two different objective evaluations across the entire scope of the spatial structure, rather than confining them to specific partitions as in the case of this experiment. The efficiency and efficacy of various approaches to multiple objective problems within IIDLE remains an area for further research.

3.3 Experiment 6 – Dynamic Function Optimisation

The previous experiment had dynamic objective functions, although given the intersection of minima; the combined global objective function had a single static solution. The intent of this experiment is to observe IIDLE's behaviour with a single uniformly applied objective function that has a moving global solution. This is a dynamic function optimisation problem.

The base objective function was modified once more to facilitate this contrived problem scenario. Similar to the previous example, a coefficient was added to the equation, which changed over the course of a cycle. In this case, instead of being applied to the evaluation values, flipping the axes of the output surface, the coefficients were applied to the input values, flipping their sign. The effect was a positive input on the x-axis became the negative input on the x-axis, with the same behaviour on the y-axis, facilitated by the symmetrical (in terms of sign) range of input values. This is essentially a gradient one-dimensional transposition where at the limits results in the original objection function or the same function rotated 180 degrees on the two-dimensional plane.

$$x' = \alpha \cdot x + (1-\alpha) \cdot -x$$
$$y' = \alpha \cdot y + (1-\alpha) \cdot -y$$

Equation 5 – Modified inputs for a dynamic Schwefel's function

Where α is the coefficient calculated as a ratio of the current positing in a cycle from the total number of algorithm iterations configured in the cycle.

3.3.1 Observations and Discussion

The modification to the selected objective function demonstrated the desired effect of moving global solution. Interestingly, when observing a two-dimensional plot of the resulting fitness-surface, the appearance (illusion) was given of the function zooming in, and then back out as the cycle executed. At the mid-point of the cycle (where the coefficient equals 0.5), the surface became a uniformly flat plane.

Observations of the systems performance over the course of a number of runs revealed some interesting and useful behaviours. Firstly, the base configuration selected (the same as in the previous two experiments) was shown to be insufficient. The movement of the peaks (duration of the cycle) was too fast, and thus the system was not given a sufficient amount of time to re-sample the function before it changed again (An important lesson for future work with dynamic problems). The configuration was adjusted to greatly increase the duration of the cycle (slow peak movement), and increase the external stimulation (access to the objective function).

Once reconfigured, the observed results were impressive. The system demonstrated (at least visually) it’s ability to track the peaks as they migrated outward from the centre of the surface to the edge of the plot and back in again as the function completed its cycles. As the function’s cycle approached and departed the midpoint of the cycle (a point where surface features were lost), the system redistributed its resources (the unit samples) somewhat uniformly across the function. As peaks migrated to the surface boundary, the system tracked the movement with units. When the surface boundary was encountered by the units, they clustered along the boundary before being redistributed to areas of use. The following series of screenshots statically capture some of the described behaviours.

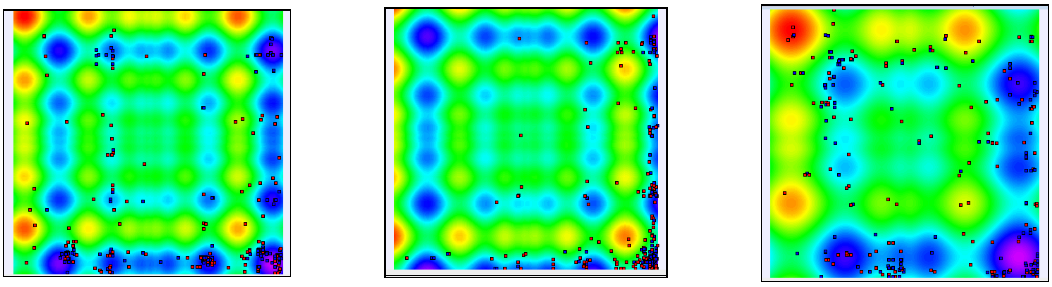

Figure 13 - Three screenshots of the unit distribution over the course of a single run. Shows various stages of the dynamic function's cycle and the units of the system tracking the global minima (dark blue and purple)

The results were not indicative of efficacy or efficiency of IIDLE on dynamic problem domains, which once configured effectively, the system is capable of demonstrating desirable behaviour for dynamic function optimisation (an encouraging sign). Although the dynamic objective function operated on a cycle, the system did not maintain units from one-step in the cycle that was used at the same point in the next cycle. The reason for this was the manner in which the system was configured in that it was unable due to compete for limited energy imposed by the decay process. This may not be a useful feature in cyclic dynamic functions, although maybe a characteristic of rapid adaptation useful in acyclic function cycles. Dynamic function optimisation is a problem area in which the application if IIDLE is expected to be very useful, and with the encouraging preliminary results observed it remains a promising area for future research.

Perhaps an extension of this work that may be useful concerning cyclic dynamic problem domains would be the use of long-lived memory cells inspired by the long-lived lymphocyte cells in the acquired immune system. In this way, specific knowledge could be retained across the problem's cycle, rather than configuring the system to replace all knowledge as quickly as possible to adapt to the current state of the problem's objective function.

4 Conclusions and Further Work

The results from all six experiments revealed information about IIDLE concerning both potential ways in which the system can be configured, and potential behaviours and ultimately types of problem domains for which the platform may be applicable. There were five primary (although preliminary) outcomes from this work, as follows:

1. **Implicit niching**: An implicit niching effect was observed with multiple objective functions, multiple constraints and perhaps occurred with multiple search strategies. Partitioning the spatially distributed system facilitates this niching behaviour, and information sharing facilitated by movement permits neighbouring niches to exploit complementary solutions. This effect was demonstrated to be beneficial in two experiments (one and four), and demands further investigation, given the potential benefits it may offer in search and optimisation.

2. **Complementing objective functions**: Adding more problem specific information to the search is known to improve results. Through partitioning and information sharing, the system demonstrated that multiple

complementing objective functions were shown to lead to an improved search strategy. More work is needed in this area both in determining additional complementary objective functions for a given problem domain, as well as comparing complementing and competing objective functions with a partitioned and non-partitioned configurations.

3. **Complementing and competing search strategies**: It was assumed that sequentially complementing search strategies would also be parallel complementing search strategies. IIDLE demonstrated that it is amenable to embedding multiple different search strategies that execute in parallel, which is an interesting result worthy of additional investigation (parallel hybrid search). The result also highlighted that point that selection of strategies is key, and that perhaps competing strategies may be more effective than complementary strategies, in such a configuration.

4. **Human-interactive search**: The IIDLE is naturally amenable to human interaction, particularly concerning the use of a human-based objective function to direct the search. An innovative interface was provided to permit users to rate the "*messiness*" of TSP candidate solutions, although unfortunately the experiment showed no benefit for the selected problem-configuration used. Given the architecture, inspiration and implementation of IIDLE (internal activation in response to external stimulation), it is strongly believed that human-interactive search is a natural fit for the technology.

5. **Dynamic objectives**: As was suspected, IIDLE appears to be naturally suited to dynamic problem domains, given a suitable representation and configuration. IIDLE's behaviour in particular on experiment six (dynamic function optimisation) highlighted the systems ability to rapidly respond (after reconfiguration) to changing conditions, and the systems performance on experiment five highlighted the potential of the system on domains with changing individual constraints. The observations, although preliminary demand the further investigation of IIDLE in dynamic settings, perhaps dynamic optimisation domains in particular.

A number of further observations and speculations were taken during the execution of these experiments regarding the suitability and applicability of IIDLE in additional scenarios. The following lists five of these scenarios as potential seeds for further preliminary experimentation on the IIDLE platform to assess viable areas of investigation.

1. The ability of the system to effectively make use of additional dynamically allocated resources, as well as the ability of the system to recover from the sudden lost or failure of resources. In particular the dynamic adding and removing of random (in regard to spatial position and connectivity) localities over the course of a simulation

2. The decentralised and distributed nature of the system thus far has only been exploited on a single digital computer during simulation. It is expected that not only is the architecture amenable to distribution across a computer network, rather the system is capable of such distribution using an equally decentralised

connectivity (network topology). Specifically the effect of various topologies should be investigated, specifically a small world network topology. Further, distributed configurations should be investigated, in particular a peer-to-peer configuration. Both small-world topology and peer-to-peer implementation are decentralised and complementary and are expected to complement the already inherently distributed and decentralised architecture and processes of IIDLE.

3. The system has demonstrated some ability to adapt and learn using a discrete-unit substrate. It may be possible to permit the architecture of the system to learn and adapt for a given problem domain. Such adaptations would consist of changes to the connectivity between localities and or the allocation of stimulation (in this case a controllable external resource) to areas of the structure showing the most promise dynamically over the course of a run. Such dynamic, reallocations of resource are expected to be both novel and beneficial in dynamic and or difficult problem domains.

4. A common theme across all the experiments executed thus far is the uniform nature of stimulation (even in the case of a partitioned system). This is because of the nature of the selected problem domains where there was complete control over the stimulating resource (objective function). It would be useful to investigate the behaviour of the system for a domain that does not have such uniformness of stimulation on IIDLE's spatial structure. Such a scenario may or may not involve a distributed implementation of IIDLE. This non-uniform stimulation of IIDLE is likely best used with the so-called conformer homeostasis manner of decay, providing in both regards (stimulation and resource maintenance) a system that is closer to the biological inspiration, and thus may provide interesting results.

5. A second common theme in the experiments executed thus far is the one-to-one mapping between candidate solution and unit (again distinctly different from inspirational metaphor). Interesting results may be achieved in problem domains in which there is a many-to-one relationship between units and solutions such as classification or function approximation, (where all units in a locality or multiple localities represent a single solution).

IIDLE is an interesting and perhaps exciting inspired machine learning platform, although their remains a lot of work to investigate its utility. Two specific additional areas of work remain to mature the understanding of the platform.

The first is a detailed algorithm description of the processes, as well as instruction and example as to how to embed specific search strategies such as genetic algorithms and ant systems. Such a work is required to both mature the understanding of IIDLE's internal operation, as well as to permit the third-party implementation of the framework and reproduction of experimental results.

The second is a definition and defence of the novelty of the IIDLE platform in the context of existing literature. This must include reference to relevant work from artificial immune systems, and clonal selection algorithms in particular. The strong relationship with evolutionary algorithms requires lengthy discussion, specifically concerning parallel evolutionary algorithms and evolutionary strategies, two approaches that bare some similarity to IIDLE.

Work on the immune inspired, decentralised, spatially distributed learning environment continues.

5 Bibliography

[1] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*, USA, Canada: Addison Wesley Publishing Company, Inc., 1989.

[2] G. Reinelt, TSPLIB - A Traveling Salesman Problem Library *ORSA Journal on Computing*, vol. 3, pp. 376--3841991.

[3] G. Reinelt. *The traveling salesman : computational solutions for TSP applications*, Berlin ; New York: Springer-Verlag, 1994.

[4] H. P. Schwefel. *Numerical optimization of computer models*, New York, NY, USA: John Wiley & Sons, Inc., 1981.

[5] Jason Brownlee, "Introduction to IIDLE - The Immunological Inspired Distributed Learning Environment," Centre for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies (ICT), Swinburne University of Technology, Victoria, Australia, Technical Report ID: 8-01, Sep 2005.

[6] Leandro N. de Castro and Jon Timmis. *Artificial Immune Systems: A new computational intelligence approach*, Springer-Verlag, 2002.

[7] Marco Dorigo and Thomas Stützle. *Ant Colony Optimization*, USA: The MIT Press, 2004.

6 Appendix A - Configurations

6.1 Experiment 1 Configuration

Variable	**Configuration**
Total Runs	100
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	200
Total cells	50
Initial units	Random [1, 10]
Initial energy	Random [0, 1]
Stimulation amplitude	5 (1/10), 2(1/8 per partition)
Stimulation type	Tour length, intersections, nearest neighbour, (all three)

Stimulation partitions	one, three
Selection	All
Proliferation	Discrete history ACO (pheromone exponent 1.0, heuristic exponent 1.5, multiplication, progeny 3)
Stop Condition	Iterations (100 steps)

6.2 Experiment 2 Configuration

Variable	**Configuration**
Total Runs	100
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	200
Total cells	50
Initial units	Random [1, 10]
Initial energy	Random [0, 1]
Stimulation amplitude	5 (1/10), 3(3/25 per partition)
Stimulation type	Tour Length
Stimulation partitions	one, two
Selection	All, Greedy
Proliferation	Discrete history ACO (pheromone exponent 1.0, heuristic exponent 1.5, multiplication, progeny 3) Local Mutation search (mutation approx 3/52 (0.06), progeny 3)
Stop Condition	Iterations (100 steps)

6.3 Experiment 3 Configuration

Variable	**Configuration**
Total Runs	10
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	50
Total cells	10
Initial units	Random [1, 10]
Initial energy	Random [0, 1]
Stimulation amplitude	User Feedback
Stimulation type	User Feedback (9 per screen, for 2 seconds)
Stimulation partitions	1
Selection	All, Greedy
Proliferation	Discrete history ACO (pheromone exponent 1.0, heuristic exponent 1.5, multiplication, progeny 3)
Stop Condition	Iterations (500 steps)

6.4 Experiment 4 Configuration

Variable	Configuration
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	200
Total cells	50
Initial units	Random [1, 10]
Initial energy	Random [0, 1]
Stimulation amplitude	5 (1/5 per partition)
Stimulation type	Schwefel's function (dimensions 2, bits per input 32)
Stimulation partitions	2 (partition 1 – x input, partition 2 – y input)
Selection	Tournament selection (group size 3, selected 2)
Proliferation	GA (progeny 2, crossover 95%, mutation 1/32)
Stop Condition	Iterations (500 steps)

6.5 Experiment 5 Configuration

Variable	Configuration
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	200
Total cells	50
Initial units	Random [1, 10]
Initial energy	Random [0, 1]
Stimulation amplitude	5 (1/5 per partition)
Stimulation type	Schwefel's function (dimensions 2, bits per input 32)
Stimulation cycle length	100 steps
Stimulation partitions	2 (partition 1 – x input, partition 2 – y input)
Selection	Tournament selection (group size 3, selected 2)
Proliferation	GA (progeny 2, crossover 95%, mutation 1/32)
Stop Condition	Iterations (500 steps)

6.6 Experiment 6 Configuration

Variable	Configuration
Movement amplitude	1
Movement probability	1
Decay amplitude	1
Decay mode	Regulatory homeostasis (energy)
Ideal energy	200
Total cells	50
Initial units	Random [1, 10]
Initial energy	Random [0, 1]

Stimulation amplitude	20 (2/5)
Stimulation type	Schwefel's function (dimensions 2, bits per input 32)
Stimulation cycle length	1000 steps
Stimulation partitions	1 (Normal)
Selection	Tournament selection (group size 3, selected 2)
Proliferation	GA (progeny 2, crossover 95%, mutation 1/32)
Stop Condition	Iterations (5000 steps)