Exploring Population Geometry and Multi-Agent Systems: A New Approach to Developing Evolutionary Techniques

Camelia Chira
Babes-Bolyai University
Cluj-Napoca
0040-264-405300
cchira@cs.ubbcluj.ro

Anca Gog
Babes-Bolyai University
Cluj-Napoca
0040-264-405300
anca@cs.ubbcluj.ro

D. Dumitrescu
Babes-Bolyai University
Cluj-Napoca
0040-264-405300
ddumitr@cs.ubbcluj.ro

ABSTRACT

Evolutionary algorithms require efficient recombination and selection mechanisms in order to produce high-quality solutions. In order to guide recombination a geometrical structure of the population is introduced. The aim of this paper is to explore connections between population geometry and individual interactions inducing autonomy, communication and reactivity. Each individual in the population acts as an autonomous agent with the goal of optimizing its fitness. In this process, each individual is able to communicate and select a mate for recombination. The introduced paradigm is illustrated by an evolutionary technique relying on a new population model and agent-based selection for recombination strategy. Search operators are asynchronously applied making the proposed approach more realistic. Numerical experiments indicate the potential of the proposed evolutionary agent-driven technique.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *intelligent agents, multiagent systems,* G.2.3 [**Discrete Mathematics**]: Applications

General Terms

Algorithms

Keywords

Evolutionary algorithms, Multi-agent systems, Population topology

1. INTRODUCTION

Reciprocal influences and cross-fertilizations of multi-agent systems (as population of agents) and evolutionary models is a promising approach to developing flexible computing techniques based on a population that can proactively evolve and emerge. Agent properties such as autonomy, communication, pro-activeness, learning and reactivity can potentially be engaged to some extent in the population of individuals used by an evolutionary model. Individuals able to act as an agent (even characterized or semi-characterized by only a few fundamental agent properties) greatly affect the way recombination and selection is carried out and therefore the way that population evolves. The agent-based behavior in population-based evolutionary models can trigger interesting new trends for the field of evolutionary computation.

The aim is to explore connections between population geometry and individual interactions inducing autonomy, communication and reactivity. The introduced model is called the *Geometric Collaborative Evolutionary (GCE)* model. Individuals are arranged according to their fitness using a predefined topological structure. Furthermore, each individual in the population acts as an autonomous agent with the goal of optimizing the individual fitness. In this process, each individual is able to communicate and select a mate for recombination. Basic agent properties can trigger a dynamic evolution emerging from multi-individual interactions similar to those defined in agent-based models.

The proposed geometric agent-based model facilitates an asynchronous search through a gradual propagation of the fittest individuals' genetic material into the population. This is achieved by considering a time dimension and a space dimension for the algorithm. Selection and recombination take place asynchronously allowing an improvement of the individuals during the process of selection and recombination within the same epoch.

Numerical experiments prove the efficiency of the proposed technique, by comparing it with the results obtained by a standard evolutionary algorithm for several difficult unimodal and multimodal real-valued functions with many dimensions.

The paper is organized as follows. The second section describes the proposed population geometry and Section 3 presents one way of involving agent societies in evolutionary optimization processes. Another powerful concept of the proposed technique, namely the asynchronous search, is described in Section 4. The resulting algorithm is given in Section 5, numerical experiments in Section 6 and conclusions and directions for further work in Section 7.

2. POPULATION GEOMETRY

The main idea of this section is to endow the population with a topological structure that can be explored for selection and recombination purposes. A new way of understanding the role of the selection process emerges and it can yield important results. The population topology can be associated to the asynchronous action of the search operators. In this way, a better balance between search intensification and diversification is hopefully obtained. An efficient exploitation of the useful genetic material

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.

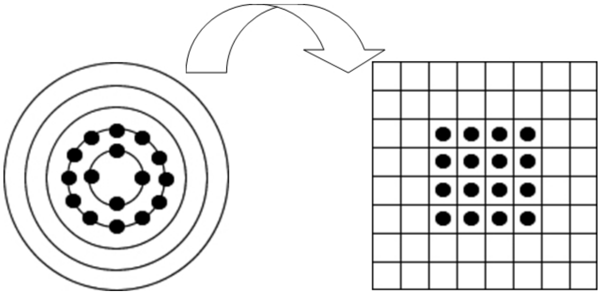

Figure 1. Layer topology of the population and the corresponding two-dimensional grid.

already obtained in the search process is completed by maintaining population diversity.

Let *P(t)* be the current population at iteration (epoch) *t*. The size of the population is fixed during all stages of the algorithm and is chosen to be a square number in order to allow a certain topology of the population. Let n^2 be the size of the population, where *n* is an even number.

All individuals from the population are sorted according to their fitness. They are distributed over *n/2* concentric circles (layers) ordered according to their fitness and starting with the fittest individuals on the most inner circle.

The number of individuals placed on layer *i, i=0,.., n/2-2,* is *4(n-2i-1)*. For simplicity reasons, the concentric circle structure is mapped to a two-dimensional grid. Figure 1 describes proposed topology using both concentric circles and a two-dimensional grid.

Let us denote the sorted population by $P(t) = x_1, x_2, ..., x_{n^2}$, where x_1 is the fittest and x_{n^2} is the worst individual in the population. The most inner layer contains the first four individuals (x_1, x_2, x_3, x_4). The next layer holds 12 individuals ($x_5, ..., x_{16}$) having the next best fitness values. The less fit individuals from the population lie on the largest circle. Let us number the layers on which the individuals are placed. The most outer layer is labeled by *0* and the label of the most inner layer has the highest value. For a population size of n^2, *n* even, there are *n/2* layers. The most inner layer is assigned with the label *n/2-1* (see Figure 2). The individuals from the most inner layer (the fittest individuals in the population) are copied in the next population just as they are.

Each individual from the population has the chance of being improved by getting involved in a recombination process (see Section 4). Population diversity is preserved as genetic material of both very fit and less fit individuals is considered in recombination. Decision about the second parent involved in each recombination process is based on agent model (see Section 4), and this is the way the exploitation of the search space is pursued.

Therefore, for each individual except the best four that are copied in the next generation, the selection scheme chooses its mate in the way described in Section 4.

A similar idea of placing population on a grid can be found in [1]. However, the latter approach does not involve sorting of individuals according to the fitness or collaboration between individuals in different agent societies.

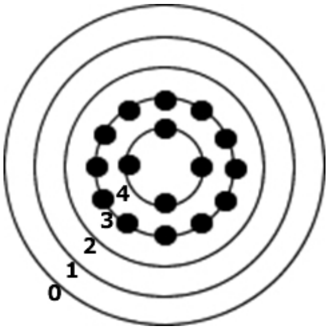

Figure 2. Five concentric circles obtained for a population of 100 individuals (a 10X10 two-dimensional grid).

3. AGENT SOCIETIES

In order to ensure a flexible search process in solving very difficult problems, population individuals can be considered members of a multi-agent system [2, 7, 9].

An agent can autonomously take decisions, acts on behalf of its creator, is situated in an environment and is able to perceive it, has a set of objectives and acts accordingly. The agent properties with a great potential for our approach refer to autonomy, communication, reactivity and learning [3, 5, 6].

Exploring agent fundamental properties is a promising approach to designing evolutionary heuristics. This approach represents a shift in evolutionary paradigm design. The population evolution is not entirely controlled by some fixed *a-priori* known operators. In the proposed agent-based approach, individuals can control to a certain extent their own evolution. This transfer of control towards individuals (that have agent capabilities) can trigger interesting behavioral types emerging from multi-individual interactions.

As an agent, each individual has the objective of optimizing its fitness. This objective is pursued by communicating with other individuals and selecting a mate for recombination based on individual strategies.

In order to ensure a balance between search intensification and diversification specialized agent societies are considered.

The proposed agent evolutionary model implies three agent societies as follows:

1. *Local Correlation (LC) society*
2. *Far Correlation (FC) society*
3. *Global Correlation (GC) society*

Agents from LC society select mates for recombination from their local neighborhood (individuals geometrically situated on the previous layer).

Agents from FC society select mates for recombination from more distanced layers on the population topology.

Agents from GC society select mates for recombination on a global basis from the entire population.

Therefore, LC agent behavior emphasizes local search while FC agents are able to guide the search towards unexplored regions. The GC society focuses on the global exploration of the search space realizing the connection between the LC and FC societies. Mate choice of these societies is depicted in Figure 3.

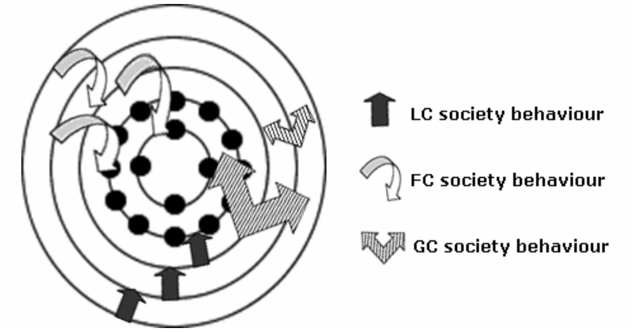

Figure 3. Mate choice of the three agent societies

Similar to the autonomy in agent-based models, each individual invited to be a mate can accept or decline the proposal according to its own strategy. Individuals from LC and FC societies accept only individuals from the same society as mates. Individuals from GC society accept any other individual as mate. Offspring are assigned to a certain society according to a dominance concept. If LC is the dominant agent society then any combination of a GC individual with an LC individual results in an offspring belonging to LC.

Table 1 summarizes these considerations.

Table 1. The society of generated offspring based on the strategies of the parents

Parent 1	Parent 2	Offspring Society
LC	LC	LC
FC	FC	FC
GC	LC	LC if LC is dominant GC otherwise
	FC	FC if FC is dominant GC otherwise
	GC	GC

4. ASYNCHRONOUS SEARCH

The search process in the proposed model takes place asynchronously. This is a distinctive and strong feature of the proposed search scheme.

The individuals from the most inner layer (the fittest individuals of the population) are copied in the next population just as they are.

LC or FC agents from layer *(n/2-2)* initiate a mating procedure by inviting individuals from layer (*n/2-1)* for recombination. As the search process progresses within an epoch the individual behavior is differentiated according to the agent society. Recombination relies on both population topology and agent behavior.

Generally, the recombination scheme can be described as follows:

- LC agents from layer *c* address mating invitations to individuals from layer *(c+1)*, where *c = 0,…,n/2-2*.
- FC agents from layer *c* address mating invitations to individuals from layer *(c+i)*, where *c = 0,…,n/2-3* and $i \geq 2$ is randomly selected using an uniform distribution. FC individuals from layer *(n/2-2)* invite individuals from layer (*n/2-1)*.
- agents from the GC society are more explorative. GC individuals from layer *c* may address mating invitations to individuals from any layer except layer *c*.

Let *x* be an individual from layer *c* and *y* an invited individual for mating. If *x* is from LC or FC and *y* belongs to the same agent society then the invitation is accepted with a given probability *p*. If *x* is from GC then the invitation is accepted by *y* with a given probability *p* (independently from the agent society of *y*).

Two individuals are recombined if and only if the invited individual accepts to be a mate. For each mating pair *(x, y)* the best offspring *z* obtained after recombination is mutated. The best between *z* and *mut(z)* is compared to the first parent *x* and replaces *x* if it has a better quality. The elitist scheme that allows only better individuals to replace the first parents is balanced by the fact that all individuals from the population are involved in recombination.

Future extensions of the proposed technique can allow mutated offspring to replace any parent. If the offspring is not better than the first parent then it is compared with the second one and can possible replace it. Furthermore, if two offspring are generated then both parents can be replaced.

The proposed model uses an asynchronous search scheme. Individuals from a layer are updated through proactive recombination and are involved in forthcoming recombination processes within the same epoch. For instance, improved individuals of layer *(n/2-2)* are targeted as mates by LC individuals belonging to the layer *(n/2-3)*.

The process that results from the described scheme is a process where the useful genetic material collected from the entire population is propagated through the layers until it reaches the less fit individuals from the population. Furthermore, the co-existence of FC and GC individuals in the same population facilitates a more aggressive search space exploration.

The selection of a mate from a layer is done by using one of the existing selection operators such as proportional selection and tournament selection. A tournament selection scheme is considered for all the experiments reported in this paper.

5. PROPOSED GEOMETRIC COLLABORATIVE EVOLUTIONARY ALGORITHM

Previous considerations regarding topological structures of the population and agent-based interactions in the population are summarized in this section by describing the proposed *Geometric Collaborative Evolutionary (GCE) Algorithm*.

The GCE technique relies on an asynchronous model for recombination. This is also an exogenous population model in the sense that each individual from a certain layer considers mates from a different layer.

The GCE algorithm (GCEA) is outlined below.

Geometric Collaborative Evolutionary Algorithm

```
begin
  t:=0
  Initialize P(t)
  For each individual in P(t)
  assign society membership (LC, FC or GC)
  while (not stop-criterion) do
  begin
    Evaluate P(t)
    GeometricSort P(t)
    for each layer c, c=n/2-2, 0
    begin
       for each individual x from c
       begin
         repeat
         begin
             if (x in LC) then
               y=LocalSelection(c+1)
             if (x in FC) then
               y=FarSelection
             if (x in GC) then
               y=GlobalSelection
         end
         until (y accepts invitation)
         z=Recombination(x,y)
         mutz=Mutation(z)
         Replace(x, z, mutz)
       end
    end
    t=t+1
  end
end
```

The GCE algorithm has been implemented and several numerical experiments have been performed to test and validate the proposed approach.

6. NUMERICAL EXPERIMENTS

Numerical experiments are performed on several benchmark unimodal and multimodal difficult functions [8] with different dimensions (D=10 and D=50):

1. Shifted Sphere Function

$$f_1(x)=\sum_{i=1}^{D} z_i^{\,2}+f\_bias_1,$$

$$x\in[-100,100]^D, f_1(x^*)=f\_bias_1=-450.$$

2. Schwefel's Problem 2.21

$$f_2(x)=\max_i\{|z_i|,1\le i\le D\}+f\_bias_2,$$

$$x\in[-100,100]^D, f_2(x^*)=f\_bias_2=-450.$$

3. Shifted Rosenbrock's Function

$$f_3(x)=\sum_{i=1}^{D-1}(100(z_i^{\,2}-z_{i-1})^2+(z_i-1)^2)+f\_bias_3,$$

$$x\in[-100,100]^D, f_3(x^*)=f\_bias_3=390.$$

4. Shifted Rastrigin's Function

$$f_4(x)=\sum_{i=1}^{D}(z_i^{\,2}-10\cos(2\pi z_i)+10)+f\_bias_4,$$

$$x\in[-5,5]^D, f_4(x^*)=f\_bias_4=-330.$$

5. Shifted Griewank's Function

$$f_5(x)=\sum_{i=1}^{D}\frac{z_i^{\,2}}{4000}-\prod_{i=1}^{D}\cos(\frac{z_i}{\sqrt{i}})+1+f\_bias_5,$$

$$x\in[-600,600]^D, f_5(x^*)=f\_bias_5=-180.$$

6. Shifted Ackley's Function

$$f_6(x)=-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}z_i^{\,2}})-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi z_i))+$$
$$+20+e+f\_bias_6$$
$$x\in[-32,32]^D, f_6(x^*)=f\_bias_6=-140.$$

For all considered functions z is given by:

$$z=x-x^*, x=[x_1,...,x_D],$$

where x^* is the shifted global optimum.

In the current implementation we consider the dominance relationship

$$GC\prec FC\prec LC,$$

meaning that LC is the dominant society. Several dominance schemes are possible. The considered one favors local search.

The results obtained using GCEA for all 6 functions are compared with the results obtained using a standard evolutionary algorithm (SEA) with the same parameters. SEA uses proportional selection and the best offspring obtained after recombination and possible mutation replaces the worst parent. The populations size for both algorithms is 100 and the number of iterations is 100*D, where D is the number of dimensions. The recombination and mutation processes are also the same for both algorithms.

The results refer to the best solution, the average solution and the standard deviation obtained at several iterations after 25 runs of each algorithm, for all considered problems (see Tables 2-5).

One can see that GCEA outperforms SEA for all considered problems, at all considered iterations, regarding both the best solution and the average solution obtained in 25 runs of the algorithms. We have considered different dimensions for the

Table 2. GCEA results for functions $f_1 - f_3$ after 25 runs, for D=10, after 100, 500, 1000, iterations

		f_1		f_2		f_3	
		GCEA	SEA	GCEA	SEA	GCEA	SEA
Iter 100	**Best**	-448.704	-137.475	-442.763	-435.259	1289.918	322784.5
	Mean	-437.229	65.93771	-438.664	-428.308	16512.04	2859401
	Std	8.800344	151.9066	1.800564	3.652689	45249.74	2147391
Iter 500	**Best**	-449.985	-449.387	-448.281	-446.194	459.4922	889.8491
	Mean	-449.828	-447.826	-447.06	-444.219	1022.646	2034.761
	Std	0.155485	0.968246	0.643879	1.132136	490.0473	855.8799
Iter 1000	**Best**	-450	-449.988	-448.943	-448.03	408.0458	575.1024
	Mean	-449.965	-449.849	-448.223	-447.056	874.2837	1162.038
	Std	0.059671	0.126279	0.394464	0.502179	463.5571	326.7314

Table 3. GCEA results for functions $f_4 - f_6$ after 25 runs, for D=10, after 100, 500, 1000, iterations

		f_4		f_5		f_6	
		GCEA	SEA	GCEA	SEA	GCEA	SEA
Iter 100	**Best**	-324.751	-313.048	-179.31	-177.244	-138.408	-134.318
	Mean	-320.363	-303.118	-178.921	-175.027	-136.899	-132.867
	Std	2.657532	6.958635	0.188076	1.209354	0.669115	0.87211
Iter 500	**Best**	-330	-326.76	-179.972	-179.784	-139.956	-139.912
	Mean	-329.725	-324.696	-179.822	-179.417	-139.805	-138.396
	Std	0.297313	1.275885	0.086546	0.209644	0.083603	0.860075
Iter 1000	**Best**	-330	-329.978	-179.972	-179.897	-139.989	-139.954
	Mean	-329.973	-329.436	-179.899	-179.778	-139.935	-139.824
	Std	0.040607	0.744477	0.050965	0.065104	0.03796	0.105457

functions because it is well known that the complexity of the problem increases as the number of dimensions increases. GCEA algorithm outperforms SEA for high dimensional problems as well. Using GCEA, the convergence process is accelerated and the search is capable of exploring more promising regions of the search space and finding better solutions. Both GCEA and SEA algorithms have the same complexity computed as number of fitness evaluations and number of performed recombinations/ mutations.

A statistical analysis is performed using the expected utility approach [4] to determine the most accurate algorithm. Let x be the percentage deviation of the solution given by the algorithm used and the best known solution on a given function:

$$x = \left| \frac{asolution - bestknownsolution}{bestknownsolution} \right| \times 100 \cdot$$

The expected utility function can be:

$$euf = \gamma - \beta(1 - \bar{b}t)^{-\bar{c}},$$

where $\gamma = 500$, $\beta = 100$ and $t = 0.000005$.

The estimated parameters $\bar{b}$ and $\bar{c}$ are calculated using the following formulae that take into account the number of test functions (i.e. six) used in our numerical experiments:

$$\bar{b} = \frac{s^2}{\bar{x}}, \bar{c} = \frac{(\bar{x})^2}{s^2}, where$$

$$\bar{x} = \frac{1}{6}\sum_{j=1}^{6} x_j, s^2 = \frac{1}{6}\sum_{j=1}^{6}(x_j - \bar{x})^2.$$

Table 6 presents the results of the statistical analysis performed on the best results of the two compared algorithms for the six functions with 10 dimensions. The last column in Table 6

Table 4. GCEA results for functions $f_1 - f_3$ after 25 runs, for D=50, after 50, 500, 1000, 5000 iterations

		f_1		f_2		f_3	
		GCEA	SEA	GCEA	SEA	GCEA	SEA
Iter 50	**Best**	16524.53	77122.09	-380.666	-377.348	9.8E+08	9.61E+09
	Mean	27025.03	91112.06	-372.359	-371.393	2.07E+09	1.63E+10
	Std	3999.502	6470.013	3.43586	3.31062	7.89E+08	2.57E+09
Iter 500	**Best**	-396.516	3986.054	-416.125	-402.789	37667.78	49507620
	Mean	-354.614	5214.711	-409.435	-396.609	139895.9	78037314
	Std	41.97308	577.7886	4.687308	3.500514	310449.1	17878481
Iter 1000	**Best**	-446.67	130.9709	-429.664	-416.903	3367.43	1512343
	Mean	-439.492	330.1781	-424.722	-412.028	7907.553	2564382
	Std	5.712628	118.1188	3.726679	2.53101	2641.899	583705.3
Iter 5000	**Best**	-449.964	-449.288	-444.196	-441.942	652.6909	1239.097
	Mean	-449.88	-448.649	-442.723	-440.539	928.8355	1869.286
	Std	0.070764	0.425314	0.967819	0.830601	137.5611	410.0351

Table 5. GCEA results for functions $f_4 - f_6$ after 25 runs, for D=50, after 50, 500, 1000, 5000 iterations

		f_4		f_5		f_6	
		GCEA	SEA	GCEA	SEA	GCEA	SEA
Iter 50	**Best**	-22.0364	206.0064	6.932391	462.9203	-123.326	-121.025
	Mean	69.56419	320.3557	67.24673	569.9548	-121.588	-120.073
	Std	37.03681	43.52046	36.8724	50.20058	0.665469	0.336128
Iter 500	**Best**	-284.272	-136.02	-178.624	-146.988	-136.794	-130.47
	Mean	-268.014	-78.0663	-178.192	-136.712	-136.236	-127.17
	Std	8.669642	25.29544	0.326283	5.133106	0.41684	1.705756
Iter 1000	**Best**	-309.603	-212.444	-178.966	-174.728	-138.996	-134.725
	Mean	-303.773	-188.552	-178.895	-173.281	-138.128	-132.822
	Std	3.445336	11.94627	0.06896	0.824102	0.388368	0.90976
Iter 5000	**Best**	-329.956	-321.584	-179.954	-179.696	-139.956	-139.889
	Mean	-329.874	-318.127	-179.839	-179.289	-139.921	-139.733
	Std	0.063486	2.040392	0.086006	0.2053	0.024758	0.085427

provides the rank of the algorithm. It can be noted that the GCEA approach obtains Rank 1 for all iterations considered.

Similar statistical results have been obtained for functions with 50 dimensions resulting that GCEA is the most accurate algorithm for the considered testing set.

7. CONCLUSIONS AND FURTHER WORK

A new evolutionary technique combining a geometric population model and a multi-agent model is proposed. Individuals are considered as agents able to communicate and to make autonomous decisions concerning a mating invitation. Three agent societies are considered and a dominance concept on these societies is introduced.

Numerical experiments are encouraging. For some well known test functions the proposed technique outperforms a standard evolutionary technique.

The proposed GCE algorithm can be furthermore extended by allowing individuals to dynamically change their agent society based on learning mechanisms that depend on the problem being solved. Moreover, an individual can act as an agent with a set of rules that are applied depending on the context. The rules of an agent can be modified through a learning process. This strategy

Table 6. Statistical analysis for GCEA and SEA on the best results for the set of six functions with D=10

Iterations	Algorithm	$\bar{x}$	s^2	$\bar{b}$	$\bar{c}$	euf	Rank
100	**GCEA**	39.2926	7331.329	186.5829	0.21059	**399.9803**	1
	SEA	13791.45	9.5E+08	68790.5	0.20048	391.1818	2
500	**GCEA**	3.0418	43.6881	14.3622	0.21179	**399.9985**	1
	SEA	21.7188	2266.348	104.3492	0.20813	399.9891	2
1000	**GCEA**	0.8142	2.9146	3.5794	0.22748	**399.9996**	1
	SEA	7.9999	311.4768	38.93498	0.20546	399.9960	2

would generalize the proposed model and enhance the autonomy of each individual having a great potential particularly for dynamic complex problems.

At any stage, a shaking mechanism changing the membership of individuals to a certain society can be engaged if the solution is not significantly improved based on the current configuration.

8. REFERENCES

[1] Alba E., Giacobini M., Tomassini M., Romero S. 2002, Comparing Synchronous and Asynchronous Cellular Genetic Algorithms. J.J. Merelo et al. (eds.), Proceedings of the Parallel Problem Solving from Nature VII, Granada (SP), LNCS 2439, p. 601-610.

[2] Bradshow, J.M. 1997. An Introduction to Software Agents, in Software Agents, J.M. Bradshow, MIT Press.

[3] Chira, O., Chira, C., Tormey, D., Brennan, A., Roche, T. 2006. An Agent-Based Approach to Knowledge Management in Distributed Design, Special issue on E-Manufacturing and web-based technology for intelligent manufacturing and networked enterprise interoperability, Journal of Intelligent Manufacturing, Vol. 17, No. 6, Springer Verlag, pp. 737-750.

[4] Golden, B.L., Assad, A.A. 1984. A decision-theoretic framework for comparing heuristics, European J. of Oper. Res., Vol. 18, pp. 167-171.

[5] Jennings, N.R. 2000. On Agent-Based Software Engineering. Artificial Intelligence Journal, 117 (2), pp. 277-296.

[6] Nwana, H., Lee, L., Jennings, N. 1996. Coordination in Software Agent Systems, BT Technology Journal, Vol. 14, No. 4, pp. 79-88.

[7] Russel, S., Norvig, P. 2002. Artificial Intelligence: A Modern Approach, Prentice Hall, 2nd edition.

[8] Tang, K., Yao, X., Suganthan, P.N., MacNish, C., Chen, Y.P., Chen, C.M., Yang, Z. 2007. Benchmark Functions for the CEC'2008 Special Session and Competition on Large Scale Global Optimization. Technical Report, Nature Inspired Computation and Applications Laboratory, USTC, China, http://nical.ustc.edu.cn/cec08ss.php.

[9] Wooldrige, M. 2002. An Introduction to Multiagent Systems, Wiley & Sons.