Investigating the Impact of Alternative Evolutionary Selection Strategies on Multi-method Global Optimization

Jacomine Grobler
Department of Industrial and
Systems Engineering
University of Pretoria and
Denel Dynamics Pty (Ltd).
Pretoria, South Africa
Email: jacomine.grobler@gmail.com

Andries P. Engelbrecht
Department of Computer Science
University of Pretoria
Pretoria, South Africa

Graham Kendall[1] and
V.S.S. Yadavalli[2]
School of Computer Science
University of Nottingham, UK[1]
Department of Industrial and
Systems Engineering
University of Pretoria
Pretoria, South Africa[2]

***Abstract*—Algorithm selection is an important consideration in multi-method global optimization. This paper investigates the use of various algorithm selection strategies derived from well known evolutionary selection mechanisms. Selection strategy performance is evaluated on a diverse set of floating point benchmark problems and meaningful conclusions are drawn with regard to the impact of selective pressure on algorithm selection in a multi-method environment.**

I. Introduction

Over the last five decades meta-heuristic algorithms have become established as the solution strategies of choice for a large range of optimization problems. The ability of a meta-heuristic algorithm to avoid local optima as well as its robustness and ease of implementation have contributed to the large amount of research carried out in recent years. Unfortunately, it is not always easy, or even possible, to predict which one of the many algorithms already in existence will be most suitable for solving a specific problem.

Within this context, the simultaneous use of more than one algorithm for solving optimization problems has recently become a popular trend in the field of evolutionary computation. Two questions that arise, however, are: “How should we select which algorithms are to be used during different stages of the optimization run?” and “How do we allocate individual solutions to selected algorithms for optimization?”

The answer may lie in the traditional evolutionary algorithm literature. Selection, one of the main evolutionary algorithm operators, directly determines how good solutions are allowed to propagate to subsequent generations. Attempting to utilize the traditional evolutionary selection operators to address the allocation of entities to algorithms in multi-method optimization strategies is a promising future research area, which we continue to explore in this paper.

This paper uses the HMHH algorithm of Grobler *et al.* [8] as basis for further investigation. The algorithm consists of seven common meta-heuristic sub-algorithms which are intelligently used at different stages of the optimization process to optimize different solutions. Five selection strategies, derived from traditional evolutionary optimization, namely random, roulette wheel, rank-based, boltzman and tournament selection are investigated. Performance is evaluated on a set of varied floating point benchmark problems and a tournament selection-based strategy is identified as the most suitable strategy for the set of unimodal optimization problems. Rank-based selection was the best performing strategy for the more multimodal problems. Very low and very high selective pressures are shown to be ineffective for algorithm selection purposes, as seen from the poor performance of roulette wheel and random selection. A comparison of the multi-method hyper-heuristic algorithm with its composing meta-heuristic sub-algorithms, is also shown to be robust and gives promising results.

The rest of the paper is organized as follows: Section II provides an overview of existing literature. Section III provides a brief overview of the HMHH algorithm while Section IV describes selection strategies which were evaluated. The results are documented in Section V before the paper is concluded in Section VI.

II. A brief review of related literature

Evolutionary algorithms, in general, focus on evolving a population of candidate solutions or entities over time in order to converge to a higher quality solution. In most cases, a single algorithm, with its own unique operators, is assigned to all entities and this algorithm is used in isolation to evolve the population.

As soon as multiple algorithms are used simultaneously to optimize a common population of entities, the allocation of entities to algorithms needs to be considered. Two main approaches can be identified from the literature. Static entity-algorithm allocation assigns entities to algorithms at the start of the optimization run and this allocation remains static throughout the rest of the run. Dynamic entity-algorithm allocation continuously updates the allocation of entities to algorithms throughout the optimization run.

The rest of this section provides a closer inspection of multi-method and portfolio algorithms and hyper-heuristics

978-1-4244-7835-4/11/$26.00 ©2011 IEEE

- all fields that have already considered the entity-algorithm allocation problem.

A. Multi-method algorithms

Memetic algorithms can be considered to be the first multi-method techniques applied to the field of computational intelligence [9]. Recently, more complex self-adaptive multi-method algorithms have been developed. The self-adaptive differential evolution algorithm of Qin and Suganthan [13] makes use of different differential evolution (DE) learning strategies which are weighted based on previous algorithm success.

The heterogeneous cooperative algorithm of Olorunda and Engelbrecht [11] makes use of different evolutionary algorithms to update each of the sub-populations in a cooperative algorithm framework, thereby combining the strengths and weaknesses of various optimization strategies within the same algorithm.

Peng *et al.* [12] developed the population based algorithm portfolio. This algorithm is based on the principle of multiple sub-populations each assigned to one algorithm from a portfolio of available algorithms. At pre-specified time intervals, entities are migrated between sub-populations to ensure effective information sharing between the different optimization algorithms.

Vrugt *et al*'s highly successful population-based genetic adaptive method for single objective optimization (AMALGAM-SO) [16] is one of the few examples of an algorithm which continually updates the allocation of algorithms to entities during the optimization run. AMALGAM-SO employs a self-adaptive learning strategy to determine the percentage of candidate solutions in a common population to be allocated to each of three evolutionary algorithms. A restart strategy is used to update the percentages based on algorithm performance.

Another successful adaptive strategy selection mechanism was investigated by Fialho *et al.* [6]. Comparisons of alternative credit assignment methods [7] and strategy selection mechanisms within a differential evolution framework [6], highlighted the superior performance of the fitness-based area-under-curve bandit technique with a rank-based reward scheme.

B. Hyper-heuristics

Burke *et al.* [2] define a hyper-heuristic as "*a search method or learning mechanism for selecting or generating heuristics to solve computational search problems*". The basic idea is not only to obtain an appropriate solution for a specific problem, but rather to focus on automating the development of the method used to obtain an appropriate solution. To achieve this, a search space of low-level heuristics is defined, upon which a high level search strategy operates to decide when to use which low-level heuristic.

Various hyper-heuristics have been developed over the last decade which could prove an effective means of addressing the algorithm selection problem. The tabu-search hyper-heuristic of Burke *et al.* [2] and the simulated annealing based hyper-heuristic of Dowsland *et al.* [4] are notable examples.

III. The Heterogeneous Meta-Hyper-Heuristic Algorithm

The HMHH algorithm of Grobler *et al.* [8] was used as a basic framework for investigating the alternative evolutionary selection strategies. The HMHH algorithm (Figure 1) consists of a common population of entities with each representing a candidate solution which is evolved over time, a set of low-level meta-heuristic sub-algorithms, an acceptance strategy, and a selection strategy.

The idea is that an intelligent algorithm can be evolved which selects the appropriate meta-heuristic at each k^{th} iteration to be applied to each entity within the context of the common parent population, to ensure that the population of entities converge to a high quality solution. The algorithm-assignment is maintained for k iterations, while the common parent population is continuously updated with new information and better solutions. Re-allocation of entities to algorithms is then performed, and the process continues.

This strategy was considered promising since each entity can use unique meta-heuristic operators, helpful for dealing with the specific search space characteristics it is encountering at that specific stage of the optimization process and the role it is playing in the larger algorithm.

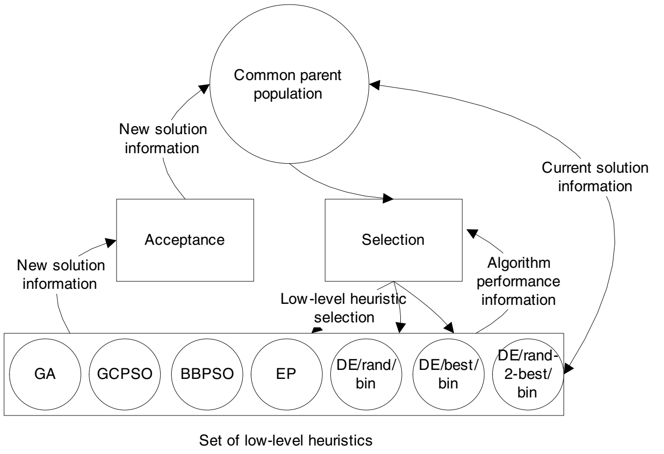

Fig. 1. The heterogeneous meta-hyper-heuristic.

In this paper, seven commonly used meta-heuristic sub-algorithms were used as the set of low-level heuristics:

- Genetic algorithm with floating-point representation, tournament selection, blend crossover [5] [11], and self-adaptive gaussian mutation [8].
- Guaranteed convergence particle swarm optimization algorithm (GCPSO) [15].
- Barebones particle swarm optimization algorithm (BBPSO) [10].
- Three differential evolution algorithms (*DE/rand/bin*, *DE/best/bin*, and *DE/rand-to-best/bin*) [14].
- Covariance matrix adapting evolutionary strategy algorithm (CMAES) [1].

A number of slight modifications had to be made to the sub-algorithms to ensure that they function effectively in the HMHH framework. For example, if the condition number of the covariance matrix exceeds 10^{14}, CMAES terminates even though the other sub-algorithms are still available for selection. In such a case, the entities previously assigned to CMAES are redistributed among the other sub-algorithms as defined by the selection strategy of the algorithm.

Algorithm 1: The heterogeneous meta-hyper-heuristic.

```
Initialize the parent population $\boldsymbol{X}$
for All entities $i \in \boldsymbol{X}$ do
    Randomly select an initial algorithm $A_{im}(1)$ from
    the set of meta-heuristic sub-algorithms to apply to
    entity $i$
end
$t = 1$
$k = 5$
while The stopping conditions are not met do
    for All entities $i$ do
        Apply sub-algorithm $A_{im}(t)$ to entity $i$ for $k$
        iterations
        Calculate $Q_{\delta m}(t)$, the total improvement in
        fitness function value of all entities assigned to
        algorithm $m$ from iteration $t + 1 - k$ to iteration
        $t$.
    end
    for All entities $i$ do
        Use $Q_{\delta m}(t)$ as input to select sub-algorithm
        $A_{im}(t + k)$
    end
    $t = t + k$
end
```

IV. Investigation of alternative evolutionary selection strategies

A number of traditional evolutionary selection strategies have been selected for investigation based on their successful application in traditional evolutionary algorithms.

A. Random selection

The random selection strategy (RAND) randomly assigns entities to algorithms every k^{th} iteration. The probability of selecting an algorithm to be applied to an entity is thus equal to $\frac{1}{n_a}$, where n_a is the number of algorithms available for selection. This strategy has very low selective pressure.

B. Roulette wheel selection

Roulette wheel selection (ROUL) is a proportional selection strategy which biases the selection towards algorithms which performed well during the previous k iterations from the start of the optimization run. This selection strategy has the highest selection pressure of all the investigated strategies. The probability of selection of algorithm m, p_m is given as

$$p_m = \frac{Q_{\delta m}(t)}{\sum_{i=1}^{n_a} Q_{\delta i}(t)}, \tag{1}$$

where $Q_{\delta m}(t)$ is defined as the total improvement in fitness function value of all entities assigned to algorithm m from iteration $t + 1 - k$ to iteration t.

C. Tournament selection

Tournament selection (TOUR), with a tournament size, n_t, of three, has a slightly lower selective pressure than roulette-wheel selection. For each entity in the population, n_t algorithms are randomly selected from the set of available algorithms. These algorithms are then compared according to their performance over the past k iterations based on $Q_{\delta m}(t)$. The entity under consideration is then assigned to the algorithm which showed the best improvement over the last k iterations.

D. Rank based selection

Rank based selection (RANK) functions on the basis that all algorithms are ranked according to their performance during the previous k iterations, where the best performing algorithm has the lowest rank and the worst performing algorithm has the largest rank. This information is then used to determine the probability of selection. Where linear ranking is used, the probability of selecting algorithm m, p_m is given as

$$p_m = \frac{\hat{\lambda} + (r_m(t)/(n_a - 1))(\hat{\lambda} - \tilde{\lambda})}{n_a} \tag{2}$$

where $r_m(t)$ is the rank of algorithm m at iteration t, $1 \leq \hat{\lambda} \geq 2$, and $\tilde{\lambda} = 2 - \hat{\lambda}$.

E. Boltzman selection

Boltzman selection (BOLT) is directly derived from simulated annealing. The selection probability of algorithm m is given as

$$p_m = \frac{1}{1 + e^{Q_{\delta m}(t)/T(t)}} \tag{3}$$

where $T(t)$ is defined as the temperature parameter. The idea is that all algorithms are provided an almost equal opportunity to be selected at the start of the optimization run. However, as the run progresses, more emphasis is placed on the selection of better performing algorithms.

V. Empirical results

The various strategies were evaluated on the first 17 problems of the 2005 IEEE Congress of Evolutionary Computation benchmark problem set, which allows algorithm performance evaluation on both unimodal and multimodal functions and includes various expanded and hybridized problems, some with noisy fitness functions.

The algorithm control parameters values listed in Table I were found to work well for the algorithms under study during previous research by the authors. $m \longrightarrow n$ indicates that the associated parameter is decreased linearly from m to n over 95% of the maximum number of iterations, I_{max}.

The results of the selection strategy comparison is presented in Table II, where the results for each strategy were recorded over 30 independent simulation runs. μ and σ denote the mean and standard deviation associated with the corresponding performance measure and $Iters$ denotes the number of iterations which was needed to reach the global optimum. Where the global optimum could not be found

TABLE II

RESULTS OF ALTERNATIVE SELECTION STRATEGY EVALUATION.

Prob(Dims)	RAND				TOUR				BOLT				RANK				ROUL			
	FF		Iters		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1(10)	−450.00	0.00	121.33	3.94	−450.00	0.00	120.00	17.84	−450.00	0.00	117.73	4.53	−450.00	0.00	115.73	5.30	−450.00	0.00	**110.13**	23.72
2(10)	−450.00	0.00	226.77	11.80	−450.00	0.00	195.73	28.41	−450.00	0.00	191.80	11.43	−450.00	0.00	**184.17**	8.34	−449.99	0.03	222.63	267.41
3(10)	$2.35E+4$	$2.28E+4$	1000.00	0.00	**−450.00**	0.00	514.27	108.23	$2.96E+3$	$4.25E+3$	1000.00	0.00	$5.34E+3$	$8.41E+3$	1000.00	0.00	$7.29E+4$	$2.39E+5$	**327.53**	342.48
4(10)	−450.00	0.00	248.47	19.28	−450.00	0.00	**199.03**	22.30	−450.00	0.00	203.37	9.14	−450.00	0.00	200.50	11.18	−448.04	10.70	200.90	225.05
5(10)	−310.00	0.00	147.67	13.73	−310.00	0.00	**118.77**	18.77	−310.00	0.00	150.87	8.23	−310.00	0.00	155.43	8.52	−309.98	0.13	255.90	329.50
6(10)	390.01	0.00	522.33	147.87	390.01	0.00	**345.43**	133.38	390.01	0.00	399.73	132.60	390.14	0.73	428.57	191.56	396.18	14.90	676.13	387.60
7(10)	1087.00	0.00	1000.00	0.00	1087.00	0.00	1000.00	0.00	1087.00	0.00	1000.00	0.00	1087.00	0.00	1000.00	0.00	1087.00	0.05	1000.00	0.00
8(10)	−119.92	0.11	1000.00	0.00	−119.92	0.11	1000.00	0.00	−119.91	0.11	1000.00	0.00	−119.92	0.13	1000.00	0.00	**−119.80**	0.13	1000.00	0.00
9(10)	−329.60	0.76	705.80	249.18	−329.90	0.30	536.57	258.27	−329.53	0.89	736.27	241.73	−329.87	0.34	637.60	254.55	−329.07	1.47	**512.03**	407.26
10(10)	−316.08	5.37	1000.00	0.00	**−311.51**	8.12	1000.00	0.00	−317.11	8.51	1000.00	0.00	−319.20	5.16	1000.00	0.00	−316.11	9.09	1000.00	0.00
11(10)	94.25	2.20	1000.00	0.00	94.45	2.08	1000.00	0.00	94.13	2.13	1000.00	0.00	**93.70**	2.20	**950.73**	187.51	96.59	1.48	1000.00	0.00
12(10)	−295.94	488.11	627.47	330.19	−130.45	624.25	747.47	322.42	**−333.39**	383.64	**614.40**	372.87	−16.88	687.88	686.73	354.70	299.04	962.31	870.37	297.33
13(10)	−129.56	0.20	1000.00	0.00	−129.53	0.20	1000.00	0.00	**−129.65**	0.18	999.77	1.28	−129.59	0.19	1000.00	0.00	−129.46	0.23	1000.00	0.00
14(10)	−296.57	0.32	1000.00	0.00	−296.35	0.30	1000.00	0.00	−296.78	0.47	1000.00	0.00	**−296.90**	0.45	1000.00	0.00	−296.47	0.38	1000.00	0.00
15(10)	**258.34**	171.46	**777.70**	311.90	376.97	192.70	981.80	67.04	403.32	176.43	936.10	205.60	357.63	202.61	885.53	219.44	415.60	175.46	949.40	192.62
16(10)	247.17	21.95	1000.00	0.00	245.91	21.01	1000.00	0.00	243.20	22.53	1000.00	0.00	**234.95**	12.75	1000.00	0.00	251.45	36.34	1000.00	0.00
17(10)	249.91	20.16	1000.00	0.00	261.54	24.31	1000.00	0.00	241.54	15.19	1000.00	0.00	**237.09**	12.77	1000.00	0.00	249.55	22.63	1000.00	0.00
1(30)	−450.00	0.00	556.70	21.97	−450.00	0.00	**325.30**	48.32	−450.00	0.00	558.60	24.50	−450.00	0.00	546.47	24.76	−450.00	0.00	396.63	167.40
2(30)	−450.00	0.00	2090.50	93.90	−450.00	0.00	**1096.40**	106.07	−450.00	0.00	1669.10	77.50	−450.00	0.00	1622.90	85.97	−360.89	164.55	2430.10	711.26
3(30)	$3.36E+5$	$1.52E+5$	3000.00	0.00	**1220**	$4.08E+3$	2999.10	4.93	$4.42E+4$	$2.72E+4$	3000.00	0.00	$1.54E+5$	$8.71E+4$	3000.00	0.00	$2.05E+6$	$1.67E+6$	3000.00	0.00
4(30)	−449.99	0.01	3000.00	0.00	−450.00	0.00	**1653.60**	126.26	−450.00	0.00	2556.10	295.81	−450.00	0.00	2479.70	307.44	8885.90	7293.70	3000.00	0.00
5(30)	1616.70	950.27	3000.00	0.00	**75.52**	485.25	3000.00	0.00	293.57	526.04	3000.00	0.00	106.17	485.42	3000.00	0.00	3167.90	1590.80	3000.00	0.00
6(30)	391.27	2.85	2599.20	447.59	390.67	1.51	**2026.00**	589.31	390.28	1.01	2297.40	455.88	**390.01**	0.00	2192.20	393.09	405.68	26.58	2861.20	482.69
7(30)	4516.30	0.00	3000.00	0.00	4516.30	0.00	3000.00	0.00	4516.30	0.00	3000.00	0.00	4516.30	0.00	3000.00	0.00	4516.30	0.00	3000.00	0.00
8(30)	−119.69	0.29	3000.00	0.00	**−119.83**	0.11	3000.00	0.00	−119.57	0.34	3000.00	0.00	−119.48	0.34	3000.00	0.00	−119.28	0.23	3000.00	0.00
9(30)	−327.42	1.06	3000.00	0.00	−324.97	2.30	3000.00	0.00	**−327.75**	1.17	3000.00	0.00	−327.68	1.39	2951.20	189.69	−317.40	18.63	**1968.10**	818.69
10(30)	−269.82	30.42	3000.00	0.00	−266.26	33.21	3000.00	0.00	−281.56	23.21	3000.00	0.00	**−285.66**	19.00	3000.00	0.00	−235.76	21.93	3000.00	0.00
11(30)	**114.53**	5.90	3000.00	0.00	118.48	5.24	3000.00	0.00	115.70	6.27	3000.00	0.00	116.47	6.06	3000.00	0.00	121.56	5.32	3000.00	0.00
12(30)	1480.70	2299.60	3000.00	0.00	1836.70	3671.80	3000.00	0.00	**1284.70**	2693.70	3000.00	0.00	1436.80	2739.80	**2980.80**	104.98	12488.00	9977.60	3000.00	0.00
13(30)	−128.67	0.62	3000.00	0.00	−128.20	0.56	3000.00	0.00	−128.50	0.51	3000.00	0.00	**−128.72**	0.49	3000.00	0.00	−127.11	1.09	3000.00	0.00
14(30)	−287.16	0.55	3000.00	0.00	−286.96	0.45	3000.00	0.00	−287.24	0.40	3000.00	0.00	**−287.29**	0.55	3000.00	0.00	−287.01	0.45	3000.00	0.00
15(30)	443.44	100.71	3000.00	0.00	470.38	111.02	3000.00	0.00	**412.01**	129.29	3000.00	0.00	464.54	124.61	3000.00	0.00	438.93	127.35	**2879.70**	457.96
16(30)	**292.73**	139.15	3000.00	0.00	329.86	177.77	3000.00	0.00	328.17	166.01	3000.00	0.00	334.05	182.23	3000.00	0.00	374.87	162.86	3000.00	0.00
17(30)	311.12	171.16	3000.00	0.00	327.27	168.39	3000.00	0.00	305.50	160.91	3000.00	0.00	**303.13**	176.64	3000.00	0.00	361.12	166.85	3000.00	0.00
1(50)	−450.00	0.00	1255.10	45.51	−450.00	0.00	**492.57**	41.38	−450.00	0.00	1317.20	81.21	−450.00	0.00	1279.10	74.66	−450.00	0.00	769.67	713.61
2(50)	−450.00	0.00	5000.00	0.00	−450.00	0.00	**2748.30**	174.20	−450.00	0.00	4807.30	161.40	−450.00	0.00	4688.30	235.01	921.14	2146.10	4882.10	160.33
3(50)	$7.34E+5$	$2.46E+5$	5000.00	0.00	$\mathbf{1.01E+5}$	$7.28E+4$	5000.00	0.00	$1.23E+5$	$4.05E+4$	5000.00	0.00	$4.03E+5$	$2.04E+5$	5000.00	0.00	$2.77E+6$	$2.93E+6$	5000.00	0.00
4(50)	−244	$1.55E+2$	5000.00	0.00	**−449.96**	0.15	5000.00	0.00	−434.32	14.74	5000.00	0.00	−427.99	22.72	5000.00	0.00	26209.00	9608.30	5000.00	0.00
5(50)	5191.00	1277.50	5000.00	0.00	4080.90	1152.20	5000.00	0.00	**3198.90**	737.17	5000.00	0.00	3551.90	898.84	5000.00	0.00	9539.40	3088.70	5000.00	0.00
6(50)	394.71	5.36	4720.30	462.25	**391.52**	2.04	**4233.80**	876.75	397.38	17.99	4622.80	648.40	396.97	17.67	4516.20	589.50	641.17	1172.00	4892.60	588.25
7(50)	6015.30	0.00	5000.00	0.00	6015.30	0.00	5000.00	0.00	6015.30	0.00	5000.00	0.00	6015.30	0.00	5000.00	0.00	6015.40	0.58	5000.00	0.00
8(50)	−119.19	0.41	5000.00	0.00	−119.21	0.40	5000.00	0.00	−119.11	0.26	5000.00	0.00	**−119.23**	0.37	5000.00	0.00	−119.03	0.17	5000.00	0.00
9(50)	−319.26	3.70	5000.00	0.00	−304.01	11.04	5000.00	0.00	−319.79	2.84	5000.00	0.00	**−320.52**	3.05	5000.00	0.00	−287.28	34.94	5000.00	0.00
10(50)	−246.45	45.39	5000.00	0.00	−219.89	49.51	5000.00	0.00	−252.91	45.77	5000.00	0.00	**−261.30**	28.40	5000.00	0.00	−126.19	59.84	5000.00	0.00
11(50)	142.25	8.10	5000.00	0.00	140.90	7.21	5000.00	0.00	**140.34**	12.52	5000.00	0.00	145.26	6.28	5000.00	0.00	147.49	4.40	5000.00	0.00
12(50)	$1.13E+4$	$9.23E+3$	5000.00	0.00	$1.32E+4$	$1.04E+4$	5000.00	0.00	$\mathbf{1.10E+4}$	$1.35E+4$	5000.00	0.00	$1.15E+4$	$9.61E+3$	5000.00	0.00	$8.09E+4$	$5.02E+4$	5000.00	0.00
13(50)	**−126.94**	0.99	5000.00	0.00	−124.63	2.49	5000.00	0.00	−126.83	1.22	5000.00	0.00	−126.77	1.33	5000.00	0.00	−122.44	3.29	5000.00	0.00
14(50)	−277.63	0.62	5000.00	0.00	−277.50	0.62	5000.00	0.00	−277.59	0.44	5000.00	0.00	**−277.64**	0.56	5000.00	0.00	−277.46	0.43	5000.00	0.00
15(50)	411.44	109.94	5000.00	0.00	**396.98**	116.05	5000.00	0.00	407.02	90.41	5000.00	0.00	437.74	99.69	5000.00	0.00	453.88	85.56	5000.00	0.00
16(50)	281.44	137.75	5000.00	0.00	303.94	117.44	5000.00	0.00	**261.82**	125.80	5000.00	0.00	274.40	131.47	5000.00	0.00	329.01	113.18	5000.00	0.00
17(50)	274.00	140.37	5000.00	0.00	254.76	109.44	5000.00	0.00	**221.46**	94.85	5000.00	0.00	283.57	148.27	5000.00	0.00	313.70	100.78	5000.00	0.00

TABLE I
HMHH ALGORITHM PARAMETERS.

Parameter	Value used
Number of entities in common population (n_s)	100
Number of iterations between re-allocation (k)	5
PSO parameters	
Acceleration constant (c_1)	$2.0 \longrightarrow 0.7$
Acceleration constant (c_2)	$0.7 \longrightarrow 2.0$
Inertia weight (w)	$0.9 \longrightarrow 0.4$
DE parameters	
Probability of reproduction (p_r)	$0.75 \longrightarrow 0.25$
Scaling factor (F)	$0.75 \longrightarrow 0.125$
GA parameters	
Probability of crossover (p_c)	$0.6 \longrightarrow 0.4$
Probability of mutation (p_m)	0.1
Blend crossover parameter (α)	0.5
GA tournament size (N_t)	13
CMAES parameters	As specified in [1].
Tournament selection parameters	
Algorithm selection tournament size (n_t)	3
Rank-based selection parameters	
Number of offspring of best entity ($\hat{\lambda}$)	3
Boltzman selection parameters	
Temperature parameter (T)	$1 \times 10^5 \longrightarrow 0$

within the maximum number of iterations, the final solution at I_{max}, denoted by FF, was recorded.

Mann-whitney U tests were used to evaluate the various strategies according to the number of iterations required to obtain the final fitness function value, as well as the quality of the actual fitness function value. The results in Table III are provided in the form: "Number of wins, number of draws, number of losses" which were obtained from a pairwise comparison of each strategy versus all of the other strategies. To illustrate, (36-13-9) in row 1 column 2, indicates that random selection outperformed roulette wheel selection 36 times over the benchmark problem set.

TABLE III
HYPOTHESES ANALYSIS OF ALTERNATIVE SELECTION STRATEGIES.

	RAND	ROUL	TOUR	RANK	BOLT	TOTAL
RAND	NA	36 − 13 − 9	11 − 25 − 22	2 − 35 − 21	3 − 38 − 17	52 − 111 − 69
ROUL	9 − 13 − 36	NA	4 − 18 − 36	5 − 13 − 40	7 − 13 − 38	25 − 57 − 150
TOUR	22 − 25 − 11	36 − 18 − 4	NA	14 − 33 − 11	17 − 29 − 12	89 − 105 − 38
RANK	21 − 35 − 2	40 − 13 − 5	11 − 33 − 14	NA	4 − 51 − 3	76 − 84 − 24
BOLT	17 − 38 − 3	38 − 13 − 7	12 − 29 − 17	3 − 51 − 4	NA	70 − 131 − 31

It is clear that random selection performs very poorly. This confirms our suspicions that simply using a number of different algorithms interchangeably throughout the optimization run, is insufficient to obtain good multi-method algorithm performance and that a more intelligent selection strategy promotes better performance. The poor performance of the roulette wheel selection strategy is slightly surprising, and can probably be attributed to the high selective pressure and associated quick convergence of fitness-based algorithms.

The better performing strategies are those with less selective pressure which allows the algorithm more time to explore and converge slowly to better performing strategies. Rank-based selection performed the best of all selected strategies over the entire problem set, with boltzman selection second and tournament selection third. A closer inspection of the results in Table II, showed that the good performance of tournament selection could be traced to its performance on unimodal problems, while rank based selection performed well when solving more multimodal problems.

In an attempt to verify the actual performance of the selection strategies, the two best performing hyper-heuristic algorithms from the previous analysis were compared under similar conditions to their composing algorithms. The actual results are recorded in Tables IV and V whereas a statistical analysis is provided in Table VI. Mann whitney U tests were again used to compare the performance of each composing algorithm to the tournament-based HH (unimodal problems) and rank-based HH (multimodal problems). The same "number of wins, draws and losses" format of Table III was also used.

TABLE VI
HYPOTHESES ANALYSIS OF HMHH ALGORITHM VERSUS ITS COMPOSING ALGORITHMS.

Problem type	Unimodal	Multimodal
Algorithm	TOUR	RANK
GCPSO	17 − 6 − 0	28 − 21 − 3
BBPSO	17 − 5 − 1	32 − 18 − 2
DE/rand/bin	17 − 5 − 1	27 − 17 − 8
DE/best/bin	18 − 5 − 0	20 − 24 − 8
DE/rand-to-best/bin	17 − 5 − 1	20 − 23 − 9
GA	22 − 1 − 0	22 − 23 − 7
CMAES	1 − 4 − 18	12 − 21 − 19
TOTAL	109 − 31 − 21	161 − 147 − 56

From the results it can be seen that the HH algorithms perform statistically significantly better a large number of times when compared to six of its seven composing algorithms. CMAES performs better than the HH when solving unimodal problems, but the HH performance starts to improve in comparison with CMAES as problem size and complexity increases. A closer inspection of the probability of selection of the various strategies, showed that the HH was, however, able to identify CMAES as the best performing algorithm and bias the allocation of entities accordingly. The data also indicates that convergence to CMAES deteriorated as problem size increased. Due to space constraints, only the results on problem CEC10 is provided as an example in Figures 2, 3, and 4.

TABLE IV
THE HMHH ALGORITHM VERSUS ITS COMPOSING ALGORITHMS.

Prob	Dims	TOUR				GCPSO				BBPSO				*DE/rand/bin*			
		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1	10	−450	0	120	17.84	−450	0	264.8	7.99	−450	0	123.07	4.7484	−450	0	324.07	5.29
1	30	−450	0	325.3	48.32	−450	0	694.23	15.40	−429	84.339	2265.4	1053.7	−450	0	1368.5	8.34
1	50	−450	0	492.57	41.38	−450	0	1129.9	37.72	−450	0	4239	350.33	−450	0	2603.2	9.82
2	10	−450	0	195.73	28.41	−450	0	385.17	11.31	−450	0	422.57	117.22	−450	0	981.57	20.60
2	30	−450	0	1096.4	106.07	−450	0	1997.7	111.39	1969.8	2467.6	3000	0	1105.9	471.86	3000	0.00
2	50	−450	0	2748.3	174.20	−450	0	4934.4	92.71	24813	14732	5000	0	46219	7302.6	5000	0.00
3	10	**−450**	0	514.27	108.23	64539	60668	1000	0.00	$2.28E+05$	$1.54E+05$	1000	0	$6.53E+05$	$2.26E+05$	1000	0.00
3	30	1224.3	4081.6	2999.1	4.93	$6.68E+05$	$3.13E+05$	3000	0.00	$9.73E+06$	$7.13E+06$	3000	0	$7.98E+07$	$1.22E+07$	3000	0.00
3	50	$1.01E+05$	72787	5000	0.00	$1.16E+06$	$5.83E+05$	5000	0.00	$4.21E+07$	$3.05E+07$	5000	0	$2.97E+08$	$5.15E+07$	5000	0.00
4	10	−450	0	199.03	22.30	−450	0	432.8	37.01	−450	0	491.37	66.165	−450	0	1000	0.00
4	30	−450	0	1653.6	126.26	1898.2	1403.9	3000	0.00	3438.4	4021	3000	0	4055.1	1001.7	3000	0.00
4	50	−449.96	0.14731	5000	0.00	$2.85E+04$	8503	5000	0.00	$4.79E+04$	17583	5000	0	$7.06E+04$	11798	5000	0.00
5	10	−310	0	118.77	18.77	−310	0	411.23	20.53	−310	0	**107.03**	11.737	−310	0	427.63	6.36
5	30	75.517	485.25	3000	0.00	4369.3	974.88	3000	0.00	5375.4	1850.6	3000	0	31.104	198.63	3000	0.00
5	50	4080.9	1152.2	5000	0.00	10520	2614.5	5000	0.00	13229	2917.4	5000	0	2576.8	1011.8	5000	0.00

Prob	Dims	*DE/best/bin*				*DE/rand-to-best/bin*				GA				CMAES			
		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1	10	−450	0	138.5	3.54	−450	0	255.4	3.43	−450	0	185.57	15.82	−450	0	**85.27**	3.03
1	30	−450	0	465.1	13.24	−450	0	875.6	4.68	−450	0	992.97	48.60	−450	0	**191.10**	4.47
1	50	−363.58	235.56	1529.4	1579.30	−450	0	1534	5.56	−450	0	4110.1	205.54	−450	0	**269.30**	7.26
2	10	−441.41	29.553	370.93	213.49	−450	0	514.73	7.55	−448.91	1.54	1000	0	−450	0	**91.57**	2.86
2	30	−437	38.328	3000	0.00	−437.93	11.04	3000	0.00	−3.3338	228.69	3000	0	−450	0	**267.83**	7.39
2	50	−133.04	978.05	5000	0.00	975.99	517.38	5000	0.00	5692	2003.1	5000	0	−450	0	**529.03**	8.69
3	10	$7.09E+04$	77215	1000	0.00	$9.69E+04$	$5.42E+04$	1000	0.00	$9.76E+05$	$1.05E+06$	1000	0	**−450**	0	**133.20**	3.79
3	30	$6.38E+06$	$3.53E+06$	3000	0.00	$9.23E+06$	$3.69E+06$	3000	0.00	$5.93E+06$	$2.65E+06$	3000	0	**−450**	0	**611.73**	13.87
3	50	$1.74E+07$	$6.47E+06$	5000	0.00	$8.17E+06$	$2.56E+06$	5000	0.00	$1.47E+07$	$4.65E+06$	5000	0	**−450**	0	**1566.00**	22.44
4	10	−450	0	328.1	9.45	−450	0	538.23	9.61	−69.812	510.25	1000	0	−450	0	**95.90**	2.83
4	30	−365.52	173.68	3000	0.00	−394.4	42.73	3000	0.00	15496	7051.1	3000	0	−450	0	**293.57**	5.70
4	50	9244.5	6281.6	5000	0.00	3058.7	1174.8	5000	0.00	49369	$1.19E+04$	5000	0	**−450**	0	**596.07**	9.98
5	10	−310	0	131.3	5.34	−310	0	345.13	7.70	559.56	1241.9	1000	0	−310	0	174.33	5.46
5	30	983.77	1499.7	3000	0.00	7.37	187.56	3000	0.00	12577	3070	3000	0	**−310**	0	**1146.50**	39.60
5	50	4753.2	1920.6	5000	0.00	1579	473.88	5000	0.00	22732	3062.5	5000	0	**−310**	0	**3414.60**	295.88

Prob	Dims	TOUR				GCPSO				BBPSO				*DE/rand/bin*			
		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
6	10	390.14	0.73	428.57	191.56	396.5	21.17	997.3	14.79	446.61	208.49	980.13	108.81	394.72	0.37	1000	0
7	10	1087	0.00	1000	0	1087	0.06	1000	0	1087	0.07	1000	0	1087	0.00	1000	0
8	10	**−119.92**	0.13	1000	0	−119.75	0.10	1000	0	−119.66	0.07	1000	0	−119.64	0.07	1000	0
9	10	−329.87	0.34	637.6	254.55	−327.55	1.45	967	125.85	−327.81	1.21	989.67	56.598	−329.99	0.00	870.7	27.476
10	10	−319.2	5.16	1000	0	−311.6	6.81	1000	0	−307.45	7.62	1000	0	−310.24	3.45	1000	0
11	10	93.701	2.20	950.73	187.51	94.096	1.59	1000	0	95.383	1.50	1000	0	97.857	0.58	1000	0
12	10	−16.884	687.88	**686.73**	354.7	518.45	1413.60	991.77	45.10	492.12	1500.60	1000	0	**−444.09**	17.15	1000	0
13	10	**−129.59**	0.19	1000	0	−129.31	0.21	1000	0	−129.39	0.21	1000	0	−129.05	0.14	1000	0
14	10	−296.9	0.45	1000	0	−297.14	0.28	1000	0	−296.81	0.35	1000	0	−296.38	0.13	1000	0
15	10	357.63	202.61	885.53	219.44	422.16	183.58	1000	0	437.82	148.26	1000	0	471.99	110.62	1000	0
16	10	234.95	12.75	1000	0	253.95	23.69	1000	0	276.05	27.40	1000	0	254.35	9.58	1000	0
17	10	237.09	12.77	1000	0	253.29	23.46	1000	0	276.37	31.53	1000	0	270.29	13.25	1000	0

TABLE V
THE HMHH ALGORITHM VERSUS ITS COMPOSING ALGORITHMS — CONTINUED.

Prob	Dims	TOUR				GCPSO				BBPSO				*DE/rand/bin*			
		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
6	30	390.01	0.00	2192.2	393.09	415.67	46.22	3000	0	$4.03E+07$	$1.53E+08$	3000	0	415.52	0.38	3000	0
7	30	4516.3	0.00	3000	0	4518.1	10.08	3000	0	4516.9	2.18	3000	0	4516.3	0.00	3000	0
8	30	−119.48	0.34	3000	0	−119.19	0.12	3000	0	−119.07	0.05	3000	0	−119.04	0.05	3000	0
9	30	−327.68	1.39	2951.2	189.69	−299.06	11.33	3000	0	−275.64	13.95	3000	0	−329.96	0.12	2984.6	18.515
10	30	−285.66	19.00	3000	0	−218.19	29.49	3000	0	−216.2	38.53	3000	0	−156.39	9.59	3000	0
11	30	116.47	6.06	3000	0	117	3.02	3000	0	116.54	3.54	3000	0	127.96	1.45	3000	0
12	30	**1436.8**	2739.80	**2980.8**	104.98	10093	8705.20	3000	0	68467	$3.96E+04$	3000	0	40981	$1.93E+04$	3000	0
13	30	**−128.72**	0.49	3000	0	−124.55	1.54	3000	0	−126.33	0.94	3000	0	−121.27	0.55	3000	0
14	30	−287.29	0.55	3000	0	−287.64	0.52	3000	0	−287.37	0.44	3000	0	−286.52	0.17	3000	0
15	30	464.54	124.61	3000	0	485.94	107.54	3000	0	488.01	113.09	3000	0	**335.3**	51.35	3000	0
16	30	334.05	182.23	3000	0	371.64	137.44	3000	0	372.13	149.91	3000	0	324.71	27.43	3000	0
17	30	303.13	176.64	3000	0	437.38	212.99	3000	0	355.61	135.46	3000	0	345.44	30.91	3000	0
6	50	396.97	17.67	4516.2	589.5	430.95	25.80	5000	0	$1.33E+08$	$1.57E+08$	5000	0	435.94	0.33	5000	0
7	50	6015.3	0.00	5000	0	6015.3	0.00	5000	0	6017	3.57	5000	0	6015.3	0.00	5000	0
8	50	−119.23	0.37	5000	0	−119.15	0.18	5000	0	−118.86	0.04	5000	0	−118.87	0.03	5000	0
9	50	−320.52	3.05	5000	0	−246.42	18.11	5000	0	−180.1	28.95	5000	0	**−329.03**	0.78	5000	0.18257
10	50	−261.3	28.40	5000	0	−93.806	50.37	5000	0	−80.598	64.84	5000	0	13.598	15.60	5000	0
11	50	145.26	6.28	5000	0	145.87	4.66	5000	0	144.43	4.69	5000	0	160.47	1.69	5000	0
12	50	**11465**	9612.80	5000	0	91606	$8.08E+04$	5000	0	$4.62E+05$	$1.82E+05$	5000	0	$1.42E+05$	$1.18E+05$	5000	0
13	50	**−126.77**	1.33	5000	0	−116.36	3.31	5000	0	1664.5	6530.20	5000	0	−109.92	0.95	5000	0
14	50	−277.64	0.56	5000	0	−278.19	0.51	5000	0	−277.61	0.55	5000	0	−276.71	0.16	5000	0
15	50	437.74	99.69	5000	0	515.42	69.75	5000	0	530.23	58.73	5000	0	**320**	0.00	5000	0
16	50	274.4	131.47	5000	0	390.07	105.33	5000	0	362.55	110.60	5000	0	371.54	9.36	5000	0
17	50	**283.57**	148.27	5000	0	350.94	130.51	5000	0	397.84	110.49	5000	0	401.62	16.43	5000	0
Prob	Dims	*DE/best/bin*				*DE/rand-to-best/bin*				GA				CMAES			
		FF		Iters		FF		Iters		FF		Iters		FF		Iters	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
6	10	428.11	207.87	451.23	226.79	391.98	0.89	1000	0	736.92	1274.00	1000	0	**390.01**	0.00	**189.5**	7.4452
7	10	1087.2	0.08	1000	0	1087	0.00	1000	0	1087	0.00	1000	0	1087	0.00	1000	0
8	10	−119.66	0.09	1000	0	−119.65	0.06	1000	0	−119.79	0.12	1000	0	−119.68	0.11	1000	0
9	10	−326.22	2.71	956.8	165.98	−329.99	0.00	612.57	28.919	**−329.99**	0.00	**189.83**	60.203	−328.04	1.51	882.03	306.01
10	10	−320.81	4.20	1000	0	−323	2.45	1000	0	−298.82	12.99	1000	0	**−328.34**	1.18	**909.47**	276.25
11	10	93.046	1.82	1000	0	96.785	0.71	1000	0	97.598	1.19	1000	0	**91.308**	1.37	**303.1**	104.96
12	10	1413.3	3709.90	788.77	307.33	−428.66	57.58	985.53	55.906	413.71	1566.80	1000	0	1086.1	2735.50	700.53	430.9
13	10	−129.48	0.17	1000	0	−129.28	0.11	1000	0	−129.55	0.17	1000	0	−129.09	0.25	1000	0
14	10	−297.12	0.48	1000	0	−296.83	0.28	1000	0	−296.3	0.31	1000	0	**−297.41**	0.53	1000	0
15	10	379.08	183.34	1000	0	509.61	58.09	1000	0	**350.48**	215.32	**664.87**	391.89	463.33	97.14	1000	0
16	10	231.51	10.57	1000	0	223.95	6.01	1000	0	282.31	23.79	1000	0	**220.73**	11.40	1000	0
17	10	236.98	13.27	1000	0	236.32	10.97	1000	0	280.48	28.05	1000	0	**219.59**	10.51	1000	0
6	30	50031	$2.37E+05$	3000	0	412.14	1.29	3000	0	1671.6	2540.40	3000	0	390.14	0.73	**1201.8**	378.7
7	30	4516.3	0.00	3000	0	4516.3	0.00	3000	0	4516.3	0.00	3000	0	4516.3	0.00	3000	0
8	30	−119.06	0.06	3000	0	−119.04	0.05	3000	0	**−119.62**	0.09	3000	0	−119.1	0.17	3000	0
9	30	−292.87	11.48	3000	0	−320.45	2.46	3000	0	**−329.99**	0.01	**1568**	399.13	−290.43	6.45	3000	0
10	30	−261.25	22.08	3000	0	−272.18	15.39	3000	0	−207.05	32.78	3000	0	**−320.6**	3.28	3000	0
11	30	108.81	4.72	3000	0	124.53	2.04	3000	0	120.67	3.45	3000	0	**99.036**	3.05	3000	0
12	30	40985	$2.82E+04$	3000	0	18815	$1.06E+04$	3000	0	14754	11587.00	3000	0	19864	19261.00	3000	0
13	30	−127.72	0.61	3000	0	−126.91	1.09	3000	0	−128.33	0.47	3000	0	−126.81	0.56	3000	0
14	30	−287.24	0.39	3000	0	−286.75	0.20	3000	0	−286.9	0.32	3000	0	**−289.6**	0.81	3000	0
15	30	455.43	108.01	3000	0	353.33	75.81	3000	0	450.54	176.73	**2821.3**	545.44	386.28	64.91	3000	0
16	30	297.01	139.39	3000	0	**228.09**	70.93	3000	0	330.76	100.30	3000	0	250.5	150.98	3000	0
17	30	350.38	150.20	3000	0	**264.87**	49.84	3000	0	376.63	162.06	3000	0	295.67	179.47	3000	0
6	50	$3.38E+07$	$9.98E+07$	5000	0	434.06	1.99	5000	0	2746.8	4657.40	5000	0	**390.14**	0.73	**2890.2**	518.3
7	50	6018.7	17.58	5000	0	6015.3	0.00	5000	0	6015.3	0.00	5000	0	6015.3	0.00	5000	0
8	50	−118.87	0.03	5000	0	−118.86	0.04	5000	0	**−119.53**	0.09	5000	0	−118.86	0.03	5000	0
9	50	−259.98	14.79	5000	0	−304.29	4.79	5000	0	−327.2	1.46	**4983.8**	88.731	−267.65	7.33	5000	0
10	50	−180.11	33.29	5000	0	−216.94	30.05	5000	0	−119.98	48.10	5000	0	**−305.44**	7.60	5000	0
11	50	134.42	8.02	5000	0	156.88	3.86	5000	0	144.82	6.02	5000	0	**109.89**	5.29	5000	0
12	50	$1.70E+05$	$1.46E+05$	5000	0	39219	$2.09E+04$	5000	0	95961	45347.00	5000	0	65366	$6.95E+04$	5000	0
13	50	−123.76	1.57	5000	0	−125.56	0.65	5000	0	−125.25	0.93	5000	0	−124.63	0.83	5000	0
14	50	−277.33	0.31	5000	0	−276.98	0.25	5000	0	−277.29	0.34	5000	0	**−280.54**	1.10	5000	0
15	50	497.02	65.80	5000	0	**320**	0.00	5000	0	383.57	76.50	5000	0	369.12	63.01	5000	0
16	50	288.48	109.76	5000	0	220.87	49.21	5000	0	311.19	84.78	5000	0	**206.87**	112.93	5000	0
17	50	389.85	126.60	5000	0	291.2	43.48	5000	0	357.15	111.19	5000	0	302.29	166.26	5000	0

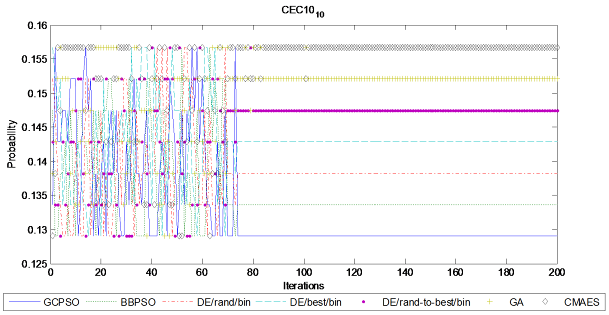

Fig. 2. Probability of selection of the RANKHH subalgorithms on the 10 dimensional CEC10 benchmark function.

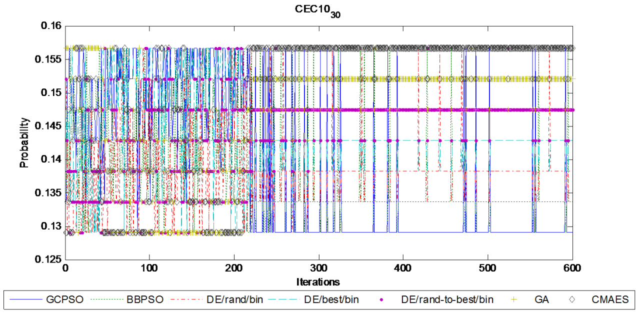

Fig. 3. Probability of selection of the RANKHH subalgorithms on the 30 dimensional CEC10 benchmark function.

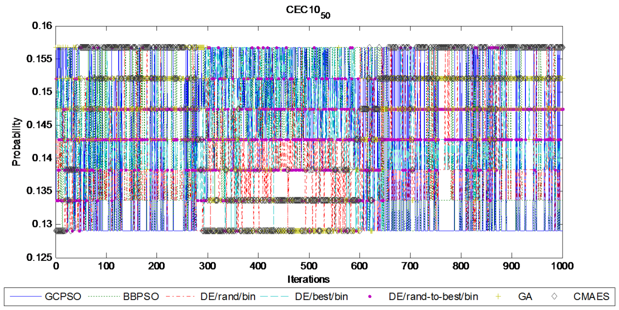

Fig. 4. Probability of selection of the RANKHH subalgorithms on the 50 dimensional CEC10 benchmark function.

VI. Conclusion

This paper investigated the impact of different evolutionary selection strategies on multi-method optimization algorithm performance. Experimental results indicated that tournament selection was the most suitable strategy for unimodal problems, whereas rank-based selection was more appropriate for multimodal problems. It was also shown that the HMHH algorithm compared favourably to its composing algorithms.

Significant future research opportunities exist in improving the performance of the HMHH algorithm through more detailed analyses of various algorithmic aspects such as, for example, the information exchange mechanisms between sub-algorithms. Comparisons against other well known techniques for operator selection, such as probability matching, adaptive pursuit and bandit-based approaches could also prove insightful.

References

[1] A. Auger, and N. Hansen, “A Restart CMA evolution strategy With increasing population size,” *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, pp. 1769-1776, 2005.

[2] E. K. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Ozcan, and J. R. Woodward, “A Classification of Hyper-heuristic Approaches,” *International Series in Operations Research and Management Science*, In M. Gendreau and J-Y Potvin (Eds.), Springer (in press).

[3] E. K. Burke, G. Kendall, and E. Soubeiga, “A Tabu-Search Hyperheuristic for Timetabling and Rostering,” *Journal of Heuristics*, vol. 9, no. 6, pp. 451–470, 2003.

[4] K. A. Dowsland, E. Soubeiga, and E. K. Burke, “A simulated annealing based hyperheuristic for determining shipper sizes for storage and transportation,” *European Journal of Operational Research*, vol. 179, pp. 759–774, 2007.

[5] L. J. Eshelman and J. D. Schaffer, “Real-coded genetic algorithms and interval schemata,” In D. Whitley, editor, *Foundations of Genetic Algorithms*, vol. 2, pp. 187–202, 1993.

[6] A. Fialho, M. Schoenauer, and M. Sebag, “Fitness-AUC bandit adaptive strategy selection vs. the probability matching one within differential evolution: an empirical comparison on the BBOB-2010 noiseless testbed,” *Proceedings of the GECCO 2010 Workshop on Black-Box Optimization Benchmarking*, 2010.

[7] W. Gong, A. Fialho, and Z. Cai, “Adaptive strategy selection in differential evolution,” *Proceedings of the 2010 Genetic and Evolutionary Computation Conference*, 2010.

[8] J. Grobler, A. P. Engelbrecht, G. Kendall, and V. S. S. Yadavalli, “Alternative hyper-heuristic strategies for multi-method global optimization,” *Proceedings of the 2010 IEEE World Congress on Computational Intelligence*, pp. 826–833, 2010.

[9] W. E. Hart, N. Krasnogor, and J. E. Smith, “Recent Advances in Memetic Algorithms,” *Springer-Verlag*, 2005.

[10] J. Kennedy and R. Mendes, “Population structure and particle performance,” *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1671–1676, 2002.

[11] O. Olorunda and A. P. Engelbrecht, “An Analysis of Heterogeneous Cooperative Algorithms,” *Proceedings of the 2009 IEEE Congress on Evolutionary Computation*, pp. 1562–1569, 2009.

[12] F. Peng, K. Tang, G. Chen, and X. Yao,“Population-Based Algorithm Portfolios for Numerical Optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 782–800, 2010.

[13] A. K. Qin and P. N. Suganthan,“Self-adaptive differential evolution algorithm for numerical optimization,” *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, pp. 1785–1791, 2005.

[14] R. Storn and K. Price, “Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces,” *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

[15] F. Van den Bergh and A. P. Engelbrecht, “A new locally convergent particle swarm optimiser,” *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 6–12, 2002.

[16] J. A. Vrugt, B. A. Robinson, and J. M. Hyman, “Self-Adaptive Multimethod Search for Global Optimization in Real-Parameter Spaces,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 243–259, 2009.