Dynamic Selection of Evolutionary Algorithm Operators Based on Online Learning and Fitness Landscape Metrics

P. Consoli, L. L. Minku, and X. Yao

CERCIA
School of Computer Science
University of Birmingham
Birmingham, West Midlands
B15 2TT, UK
{p.a.consoli,l.l.minku,x.yao}@cs.bham.ac.uk

Abstract. Self-adaptive mechanisms for the identification of the most suitable variation operator in Evolutionary meta-heuristics rely almost exclusively on the measurement of the fitness of the offspring, which may not be sufficient to assess the optimality of an operator (e.g., in a landscape with an high degree of neutrality). This paper proposes a novel Adaptive Operator Selection mechanism which uses a set of four Fitness Landscape Analysis techniques and an online learning algorithm, Dynamic Weighted Majority, to provide more detailed informations about the search space in order to better determine the most suitable crossover operator on a set of Capacitated Arc Routing Problem (CARP) instances. Extensive comparison with a state of the art approach has proved that this technique is able to produce comparable results on the set of benchmark problems.

1 Introduction

Parameter Setting has recently become one important area of research in the Evolutionary Computation field. Since an a-priori identification of the optimal configuration of the parameters is always time-consuming and often not practicable, one must employ a dynamic selection strategy of the optimal configuration which is performed while the search is being executed. In addition, a static set of parameters is not always the optimal choice for a large number of problems where self-adapting techniques have proven to be more effective[8].
The problem of identifying the most suitable variation operator among several, also known as Adaptive Operator Selection (AOS), can be divided into two sub-tasks: the Credit Assignment (CA) mechanism, used to evaluate the performance of the operators, and the Operator Selection (OS) Rule, necessary to determine the most suitable operator using the information provided by the CA mechanism. The majority of the Credit Assignment approaches in literature are based on the evaluation of the fitness of the offspring generated by the operator, which is compared either to the current best solution [6], to the median fitness

[14] or to the parents' fitness[2]. A different strategy evaluating both fitness and diversity of the offspring was proposed for multi-modal optimization in [18]. The reward has been mostly considered as the value assessed during the last evaluation (*Instantaneous* reward), as the average reward over a window of the last N evaluations (*Average* reward), and as the biggest improvement achieved over a window of the last N evaluations (*Extreme* reward)[9]. A different approach for population based meta-heuristics, proposed in [4], assesses the reward as the proportion of solutions generated by each operator which have been selected by the ranking phase of the evolutionary algorithm. Credit Assignment mechanism are coupled with Operator Selection rules such as Probability Matching[10], Adaptive Pursuit[24] or Multi Armed Bandit solvers (MAB)[5].
From the analysis of the existing literature, it is clear how almost all the existing CA strategies rely exclusively on the mere evaluation of the fitness of the offspring. However, the information provided by the fitness may not be sufficient to assess the optimality of an operator (e.g. in a landscape with a high degree of neutrality). The purpose of this work is therefore to develop a new dynamic CA mechanism which considers a suite of measures, and that can be adopted also as an Operator Selection Rule. We consider the Memetic Algorithm with Extended Neighborhood Search (MAENS*)[4] algorithm as a case study and for comparison purposes. More specifically, we aim to answer to the following research questions:

- **RQ1** : What kind of additional information we can provide to the Credit Assignment technique for a more "aware" calculation of the reward and does this information effectively help to improve the prediction ability of the algorithm?
- **RQ2** : What technique would be useful to handle this data and to select the most suitable operator in such a dynamic environment? Would the prediction ability of the technique be better than that of MAENS*? Would the use of this technique improve the optimization ability of MAENS*?

The contributions of this work are:

- An ensamble of four different online Fitness Landscape Analysis techniques, performed during the execution of the MAENS* algorithm in order to give a more accurate description the current population (RQ1);
- A Credit Assignment technique based on the use of a online learning algorithm to predict the reward of the most suitable operator (RQ2).

The results of the experiments carried out show that the proposed approach is able to produce results comparable to a state-of-the-art strategy and reveal how in some cases the presence of a set of measures have a beneficial effect on the optimization ability of the AOS.
The rest of the paper is divided as follows. Section 2 introduces the case scenario and the base MAENS* algorithm. Section 3 describes the ensamble of Fitness Landscape Techniques used in conjunction with the CA mechanism of the MAENS* algorithm. Section 4 describes the online Learning algorithm that

has been used and adapted for the CA system. Section 5 includes a description of the proposed MAENS*-II algorithm. Section 6 describes the experiments that have been carried out to verify the assumptions of this research and their results. Finally, section 8 includes the conclusions and some future work ideas.

2 Background

2.1 Capacitated Arc Routing Problem

The Capacitated Arc Routing Problem (CARP) [11] is the problem of minimizing the total service cost of a routing plan, given a set **T** of tasks (which correspond to a subset of the arcs of a graph) and a fleet of **m** vehicles with capacity **C**. Each task **t** has a *service cost* **sc**, a *demand* **d** (the load of the vehicle necessary to service the task), a unique **id**, a reference to its head and tail vertices, and must be served once and entirely within the same route $\mathbf{R_i}$. Solutions are represented by a permutation of the tasks, divided into several routes, which must start and end in a specific vertex called *depot*. The service cost of a single route is calculated adding the service cost of all the tasks in the route plus the cost of the shortest path **sp** between each task. The problem can be formally defined as follows:

$$\min \mathrm{TC}(S) = \sum_{i=1}^{\mathrm{length}(S)-1} (sc(t_i) + sp(t_i, t_{i+1}))$$

subject to the constraints

$$load(R_i) \leq C \ , app(t_i) = 1 \text{ and } \forall t_i \in T, m <= nveh$$

$$load(R_k) = \sum_{i=1}^{\mathrm{length}(R_k)} d(t_{ik})$$

where $\mathbf{app(t_i)}$ gives the number of appearances of tasks $\mathbf{t_i}$ in the sequence **S** and **nveh** is the number of available vehicles.

2.2 A case study: MAENS*

Among the several approaches for CARP involving Evolutionary Algorithms existing in literature, one of the most competitive is MAENS [23], a memetic algorithm which makes use of a crossover operator, a local search combining three local move operators and a novel long move operator called MergeSplit, and a ranking selection procedure called Stochastic Ranking (SR)[21]. The algorithm was recently refined into the MAENS*[4] algorithm. The major differences with the original algorithm are: (a) the crossover operator is replaced by a set of four operators, namely GSBX, GRX, PBX, SPBX, (b) a dynamic MAB mechanism (dMAB) [9] is adopted as an AOS rule, (c) a novel CA mechanism assigns a reward to the operators which is proportional to the number of solutions generated by each operator that "survived" the ranking phase, (d) the Stochastic

Ranking is improved considering also the diversity of the solutions (dSR) using a (e) novel diversity measure for the CARP search space.
The dMAB [9] approach, adopted in this work, combines the UCB1 algorithm [1] with the Page-Hinckley (PH) statistical test [13] to detect changes in the environment. When the PH test is triggered the MAB system is restarted and the information gathered in the previous generations is discarded. The MAENS* algorithm represents the case study of this research, as the presence of a suite of crossover operators allows the study of several AOS approaches.

3 Online Fitness Landscape Analysis

The existing Fitness Landscape Analysis (FLA) techniques have been analysed with the purpose to identify those that can be used in the CARP context. Such selection has been driven by both the necessity to reduce at most the computational effort, by exploiting some calculations that are already performed by the algorithm and the necessity to identify measures able to "capture" different features of the landscape. We identified a set of four FLA techniques, namely an evolvability measure, two neutrality measure and a fitness distribution measure, as they describe different features of the landscape and do not considerably increase the computational effort. The FLA techniques are then employed during each generation, and their results, in combination with the CA technique of MAENS, are used to create a more accurate and informative "snapshot" of the current population which eventually might lead to a more aware selection of the crossover operator. A final remark is necessary about the constraints handling and how it affects the fitness of the individuals. The landscape in which MAENS* operates is that of solutions which can potentially violate the capacity constraints of the vehicles. Therefore, we consider the following fitness function, adopted from [23]:

$$f(S) = TC(S) + \lambda * TV(S)$$

where λ is an adaptive parameter depending on the cost, on the violation and on the best feasible solution found so far, $TC(S)$ is the total cost of the solution and $TV(S)$ its total violation.
The rest of the section will introduce the four FLA techniques that have been considered in this work and how they have been integrated in the MAENS* algorithm.

Accumulated Escape Probability The Accumulated Escape Probability[16] (aep) is a technique which aims to measure the evolvability, which can be defined as the capacity of the solutions to evolve into better solutions. We obtain the Accumulated Escape Probability (aep) by averaging the mean escape rate[19] (the proportion of solutions with equal or better fitness in the neighbourhood) of each fitness level with the formula:

$$aep = \frac{\sum_{f_i \in F} P_j}{|F|}, \text{ where } F = f_0, f_1, ..., f_L$$

where f_i is a fitness level (subset of all the solutions with fitness equal to f_i), P_j is the average Escape Rate of all samples belonging to the f_j fitness level and L is the number of possible fitness levels. Being the mean value of a set of probabilities, the aep will be equal to 0 when the instance is hard and higher (up to 1) in the opposite case. The calculation of the aep requires the analysis of the neighbourhood of each solution in order to identify how many individuals have a equal or better fitness than the original individual. We analyse therefore the evolvability of the solutions which have been selected (with probability equal to 0.2) for the local search. Since the calculation of the neighbourhood of each solution corresponds to the first step of the local search, no significant additional cost is required to compute the aep.

Dispersion Metric The analysis of the distribution of the solutions within the landscape can be sometimes used to understand more about the difficulty that a "jump" between fitness levels requires and to gain some information on the global structure of the landscape. In this context, the Dispersion Metric (dm) [17] is a technique to obtain information about the global structure of the landscape, by measuring the dispersion of the best solutions. Ideally, if the best solutions are very close we might be in presence a single funnel structure. If, on the contrary, solutions get more distant when their fitness improves, the landscape might be more like a multi funnel structure. The technique can be described as follows:

1. A sample S of solutions is taken from the search space;
2. the best S_{best} solutions are selected from the S (using a threshold value);
3. the average pairwise distances in S ($\overline{d}(S)$) and in $S_{best}(\overline{d}(S_{best}))$ are calculated;
4. the dm is obtained as the difference between $\overline{d}(S_{best})$ and $\overline{d}(S)$.

The calculation of the pairwise distance between all the individuals of the sample is already performed during the dSR and therefore requires no additional cost. Thus, the dm can computed on the set of all the *popsize∗offset* individuals created during each generation of MAENS*. More information about the distance measure can be found in [4]. Finally, it is possible to rely on the ranking performed by the dSR operator and choose these solutions as the subset of the best ones.

Neutrality Measures Neutrality is the study of the width, distribution and frequency of neutral structures within the landscape (e.g. plateaus, ridges). A set of several neutrality measures was defined in [25]. Among these, we selected the following:

1. average neutrality ratio ($\overline{r}$): can be obtained averaging the neutrality ratio (e.g. the number of solutions with equal fitness) of each individual with respect to its neighbourhood;
2. average $\Delta-$fitness of the neutral networks ($\Delta(\overline{f})$): can be defined as the average fitness gain after one mutation step of each individual belonging to a neutral network.

In the same fashion as in the case of the aep, the computation effort of this technique can be absorbed by the generation of the neighbourhood of the initial solution during the local search.

4 Online learning

The AOS model followed in MAENS* is that of the Multi Armed Bandit scenario, where the UCB1 [1] algorithm is used to balance the exploration and the exploitation of the crossover operators and the Page-Hinckley [13] test is used to detect when a different operator has become the most suitable.
In this work, we propose the adoption of a different model. The abrupt and scarcely predictable changes of the most suitable operator which might happen during the search show many similarities to the notion of concept drift [22][20] in machine learning. Thus, in such a context, we might adopt a online learning algorithm capable of (a) predicting a reward for each operator using the online Fitness Landscape Analysis measures and (b) detecting the changes of the environment, relying only on a limited number of training instances. We employ the Dynamic Weighted Majority (DWM) [15] algorithm as our online learning algorithm, which has proved to be one of the most effective techniques in the task of tracking the concept drift. The DWM algorithm can be described as follows. A set of learners (called experts) are used to classify the incoming instances $\{\overrightarrow{x}, y\}$, where $\overrightarrow{x}$ is the vector of the n input features and y is the output feature. Each expert e_j has a own weight w_j, and operates a classification λ of the instance. The global prediction is identified as the prediction with the largest sum of weights. All the experts which have failed to classify correctly the instance have their weights reduced of a β factor. Moreover, every p instances, all the experts with a weight below a certain threshold θ, are deleted and a new expert is created if the global prediction is wrong.

DWM for the regression task As the DWM algorithm was originally conceived for a classification purpose it is necessary to adapt and modify some of its mechanism for the regression task of predicting the reward of a given operator based on the FLA techniques. A pseudocode of the revised DWM algorithm for the regression task (rDWM) is available in table 1, where the grey lines indicate the novelties introduced with respect the original algorithm previously described. The modifications introduced are:

1. The global prediction σ_i is obtained calculating the weighted average of all predictions (line 10);
2. we consider correct the predictions if the difference with the output feature is less than a threshold τ (lines 5-6);
3. a new expert is created if the difference between the global prediction and the output feature is less than a t factor (lines 17-18);
4. we introduce a window containing the last n instances wTS, which is used to train the new experts upon creation (line 2).

Table 1: Dynamic Weighted Majority algorithm for the regression task

```
for (each instance {x⃗_i, y_i}) do
    update wTS(x⃗_i);
    for (each expert e^j) do
        λ^j =classify(e^j, x⃗_i);
        if ( |λ_i^j − y_i| > tau ) then
            w_j = β ∗ w_j;
        end
    end
    normalize weights;
    σ_i= weighted average of the prediction of all the experts;
    if (p mod i = 0) then
        for (each expert e^j) do
            if (w_j < θ) then
                delete expert;
            end
        end
        if ( |σ_i − y_i| > t ) then
            create new expert and train using wTS;
        end
    end
    for (each expert e^j) do
        train(e^j, x⃗_i);
    end
end
```

5 MAENS*-II

The revised version of the algorithm adopting the rDWM as an AOS mechanism, named MAENS*-II, is shown in the pseudocode included in table 2, where the grey lines highlight the modifications over the MAENS* algorithm previously introduced in section 2.2. Further information about MAENS* can be found in [4]. A set of four (one for each crossover operator) rDWM instances are created upon initialization of the algorithm (line 2). During each generation, one new training example is created for each rDWM instance by using the current set of FLA metrics as input features, and the reward associated to the operator as output feature (lines 10, 13-14). The set of four rDWM instances are then used predict the reward of each operator (line 4). The algorithm adopts an Instantaneous Reward mechanism to choose among the options, to limit the bias constituted by the performances of the operator in the previous generations and facilitate, in this way, the tracking of the concept drift. All the experiments were performed using the *weka* [12] implementation of REPTrees as base learners.

6 Experimental Studies

A set of experiments was designed to verify the behaviour of MAENS*-II. As a first step, an oracle was implemented with the purpose of analysing a set of CARP instances in order to obtain optimal crossover operator selection rates and to exclude those instances where all the crossover operators achieve the same results. The oracle can be briefly described as follows. Four different populations are obtained during each generation by using each crossover operator.

All the individuals of the four generations are merged into a single population which is sorted using the MAENS* ranking operator. The Credit Assignment mechanism is therefore used to assess the best operator. The results achieved by the oracle show that the predictions operated by the dMAB are not optimal, as better results can be achieved. Besides, these results should be considered "optimal" only when the MAENS* reward measure is considered, while they might not be anymore when in presence of a set of multiple measures, as in the case of MAENS*-II. The experiments were performed on instances taken from the known benchmark test sets proposed in [7] and [3], named *egl* and *C,D,E,F*. The analysis of the results achieved by the oracle allowed to identify a subset of 42 instances. The set of parameters adopted in the MAENS*-II algorithm, included in table 3, was identified with a series of test-and-trial attempts and might not correspond to the most optimal one. All the values were obtained by averaging the results of 30 independent runs and all the experiments are performed on the instances selected from the two different datasets.

Effectiveness of the FLA measures (RQ1) A first experiment was designed to understand whether the use of the online FLA techniques has a beneficial effect on both the optimization ability and the prediction capacity of the algorithm. Therefore, the performances of the MAENS*-II were compared to that of a version of the algorithm which only makes use of the original reward of MAENS* as an input feature of the learning algorithm, named MAENS*-rw. In this context, we are not interested in the results achieved by the algorithm but rather we want to verify that the results are significantly different and prove, as a consequence, a certain sensibility of the rDWM algorithm to the presence of the FLA measures. The results are included in table 4 in columns MAENS*-rw and MAENS*-II. The results have been tested for significance using the Wilcoxon signed-rank test across the problem instances, which confirmed that the two algorithms produce significantly different results (respectively $W = 26$ with $p < 0.05$ and $W = 54.5$

Table 2: MAENS*-II pseudocode

initialize a population *pop* of *popsize* individuals;
initialize a set of four rDWM$_i$ instances and a set of rewards rw_i (one for each crossover operator)
while *(termination condition not met)* **do**
    choose the crossover operator op_i with largest rw_i
    generate a population $pop_x$of *popsize∗offset* individuals, choosing the parents from $pop \cup pop_x$;
    generate $pop_{ls}(i)$ for each individual $pop_x(i)$ with probability = 0.2;
    if *($pop_{ls}(i)$ is better than $pop_x(i)$)* **then**
        overwrite $pop_x i$;
    end
    calculate aep, $\overline{r}$, $\Delta(\overline{f})$ and the dm measures
    use d-Stochastic Ranking and overwrite *pop*;
    use the MAENS* CA approach to calculate the output feature out_i for each op_i;
    for *each crossover operator op_i* **do**
        rw_i = rDWM$_i$([aep, $\overline{r}$, $\Delta(\overline{f})$, dm,], out_i)
    end
end

sample size: 42). The comparison of the average fitness shows that MAENS*-rw produced slightly better results in only 6 instances out of 42 and considerably worse ones in all the rest. This can be interpreted as a signal that the rDWM is concretely affected by the FLA measures, which influence (in a beneficial way) the decisions made by the algorithm.

MAENS*-II vs MAENS* (RQ2) The second research question focuses on the performance of the proposed approach with respect to the existing one. Therefore, the MAENS*-II was tested against the MAENS* algorithm and the oracle. A Wilcoxon signed-rank test performed on the dataset proved that the differences between the results achieved by the two algorithms are not statistically significant ($W = 375$ with $p > 0.05$ and sample size: 40). The results are similar also in terms of mean average fitness over all the instances, standard deviation and best solution. The online learning system is therefore able to achieve results comparable to those achieved by the bandit solver. Despite this result, it is possible to notice some significant differences between the results in some of the instances. A Wilcoxon rank-sum test was performed on each couple of results and 6 instances (highlighted in boldface in table 4) showed statistically significant results. A comparison of the selection rates of such instances is included in figure 1. The ordinates axis in the figure refers to the selection rate of each crossover operators, while the abscissas corresponds to the average fitness of the population discretised into 50 intervals. We study, therefore, how the selection rate of the four operator changes while the search is carried on and the average fitness of the population decreases. In the first instance, *egl-s1-B*, it is possible to notice three phases in the oracle prediction. A first phase where the GRX operator is preferred over the others, an intermediate phase where the GRX and GSBX operators have nearly equal selection rates and a last phase characterized by a rise of the selection rate of the GRX operator which reaches 1 in the last moments of the search. Both MAENS* and MAENS*-II award the GSBX operator with the highest selection rate for the whole search, missing the prediction of the oracle. It is possible to see, however, how MAENS*-II increases the selection rate of GSBX more rapidly than MAENS*. In the second instance, the oracle clearly identifies a change in the environment halfway through the search. The concept drift is not detected by both MAENS* and MAENS*-II, which, however shows an higher exploitation of the GSBX operator. The performance of MAENS*-II instances suggests the hypothesis that despite the not enhanced prediction ability, the availability of more than one measures has led

Table 3: Parameters of the FLA-MAENS* algorithm

Name	Description	Value	Name	Description	Value
psize	population size	30	p	expert removal period	5
ubtrial	maximum attempts to generate a solution	50	β	decrease factor for expert weights	0.75
opsize	size of the offspring during each generation	6*psize	τ	expert weight reduction threshold	0.05
P_{ls}	probability of performing the local search	0.2	θ	threshold for expert removal	0.05
pMS	routes selected during MergeSplit	2	t	threshold for expert creation	0.10
G_max	maximum generations	500			
SR_{r1}	probability of sorting solutions using diversity	0.25			
SR_{r2}	probability of sorting solutions using fitness	0.70			

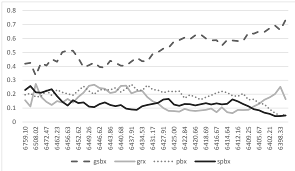
Table 4: Experimental results. The first two columns show the instance name (*inst*) and the best known result (*BK*). Further columns show the average fitness of the best solution (*avg*), the standard deviation (*std*), the best solution (*best*) achieved by the four different versions of the MAENS* algorithm. Instances in boldface show results statistically significant between MAENS* and MAENS*-II with $p < 0.05$ according to the Wilcoxon rank sum test. The avg row shows the average value of each column. Bottom row shows the number of comparisons won (W), drawn (D), and lost (L) to MAENS*-II in terms of average fitness of the best solution

		MAENS*-rw			MAENS*-II			MAENS*			oracle		
inst	BK	avg	std	best	avg	std	best	avg	std	best	avg	std	best
C01	1590	1668.67	13.16	1660	1670.00	17.75	1660	1671.67	19.38	1660	1665.33	13.03	1660
C05	2410	2483.33	18.36	2470	2474.00	5.83	2470	2471.00	2.00	2470	2470.00	0.00	2470
C06	855	905.17	3.98	895	901.00	4.90	895	902.00	4.58	895	896.67	3.73	895
C09	1775	1840.33	20.41	1820	1824.00	10.12	1820	1830.00	16.73	1820	1829.00	15.08	1820
C10	2190	2277.33	11.53	2270	2275.17	9.53	2270	2272.17	6.54	2270	2270.67	3.59	2270
C11	1725	1832.00	27.37	1815	1817.33	2.49	1815	1816.33	3.14	1805	1815.17	2.41	1805
C18	2315	2407.17	6.91	2390	2402.76	9.27	2385	2403.67	7.74	2385	2401.17	7.82	2390
D01	725	734.83	8.99	725	742.17	5.11	725	742.83	4.02	725	739.50	6.87	725
D07	735	836.33	3.40	835	835.00	0.00	835	835.00	0.00	835	835.00	0.00	835
D08	615	692.00	4.58	685	687.67	4.42	685	685.67	2.49	685	685.67	2.49	685
D11	920	937.67	6.42	920	936.72	3.48	930	936.50	3.91	935	934.67	4.99	920
D21	695	818.67	11.47	810	814.00	4.16	805	814.83	5.24	805	810.17	3.98	805
D23	715	772.83	12.23	745	**767.67**	7.39	755	**769.83**	12.28	740	758.17	8.51	740
E01	1855	1941.00	6.11	1935	1936.50	2.93	1935	1936.17	2.11	1935	1935.50	1.50	1935
E09	2160	2266.33	25.26	2230	2249.17	21.64	2225	2252.00	21.16	2230	2250.83	21.26	2230
E11	1810	1878.00	25.68	1850	1858.00	15.03	1840	1857.00	13.52	1835	1853.83	7.71	1845
E12	1580	1741.00	17.63	1710	1722.50	14.59	1695	1717.33	13.15	1695	1719.50	11.50	1705
E15	1555	1608.67	5.91	1595	**1604.33**	5.59	1595	**1602.50**	6.68	1590	1599.50	6.24	1590
E19	1400	1444.67	1.80	1435	**1442.00**	4.58	1435	**1442.67**	4.23	1435	1438.33	4.71	1435
E21	1700	1707.67	2.49	1705	1708.10	2.39	1705	1708.00	2.45	1705	1705.67	1.70	1705
E23	1395	1440.50	7.34	1435	1435.50	1.98	1430	1435.50	1.50	1435	1434.00	2.00	1430
F01	1065	1071.43	2.54	1065	1072.59	3.32	1065	1071.83	2.73	1065	1069.50	3.73	1065
F04	930	954.67	5.31	940	954.00	4.16	940	953.67	3.64	940	951.17	3.80	940
F09	1145	1165.54	12.34	1145	1163.45	8.48	1145	1161.00	11.79	1145	1157.00	8.12	1145
F11	1015	1026.96	13.56	1015	1027.07	12.35	1015	1030.00	11.11	1015	1021.00	6.88	1015
F12	900	940.71	32.32	910	931.83	26.94	910	925.00	23.42	910	917.33	13.09	910
F14	1025	1035.83	13.17	1025	1034.50	11.86	1025	1037.33	12.23	1025	1033.00	13.52	1025
F19	685	737.67	8.73	725	**732.50**	9.64	725	**735.17**	9.35	725	726.67	3.73	725
F24	975	997.00	9.36	980	998.83	10.38	975	999.33	8.63	980	990.50	11.28	975
e1-B	4498	4509.17	11.68	4498	4504.79	10.42	4498	4501.20	8.33	4498	4502.60	8.50	4498
e2-B	6305	6329.83	13.35	6317	6323.86	9.41	6317	6323.67	9.58	6317	6320.37	6.36	6317
e4-A	6408	6464.07	5.39	6446	6463.83	5.07	6446	6462.50	3.04	6450	6462.77	2.58	6456
e4-B	8884	9023.47	16.23	8992	9021.10	17.84	8990	9022.50	16.39	8988	9011.20	11.79	8993
s1-B	6384	6407.30	19.35	6388	**6397.59**	12.70	6388	**6399.90**	16.38	6388	6399.70	14.50	6388
s2-A	9824	9943.43	32.78	9889	9934.80	29.49	9881	9931.63	26.62	9889	9928.37	27.01	9885
s2-B	12968	13217.13	44.41	13159	**13171.41**	29.10	13123	**13179.07**	26.11	13124	13179.20	29.61	13124
s2-C	16353	16516.03	46.02	16430	16505.83	51.89	16434	16510.10	43.05	16430	16498.00	41.64	16433
s3-A	10143	10293.87	29.07	10242	10290.67	25.78	10251	10282.63	29.41	10221	10276.50	26.39	10221
s3-B	13616	13874.37	59.29	13736	13821.50	47.04	13747	13820.13	57.75	13736	13823.37	60.51	13750
s3-C	17100	17325.90	46.56	17237	17309.87	37.46	17221	17289.73	42.75	17220	17296.10	33.42	17249
s4-A	12143	12403.37	47.36	12316	12388.59	41.42	12316	12400.87	47.91	12283	12382.93	41.71	12304
s4-B	16093	16454.30	42.73	16351	16437.60	54.52	16281	16421.17	50.46	16325	16414.67	47.18	16344
avg	4266.16	4355.38	17.91	4327.16	4347.37	14.58	4323.88	4346.69	14.60	4322.95	-	-	-
		W	D	L				W	D	L			
		6	0	36				18	2	20			

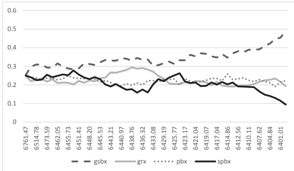
to better results in some instances, outperforming even the oracle, based only on the use of the Credit Assignment system of MAENS*.

7 Conclusions and future work

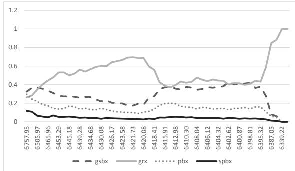
In this work we proposed the adoption of a novel Adaptive Operator Selection scheme to identify the optimal crossover operator. The AOS is tested against the Multi Armed Bandit approach employed in the MAENS* algorithm for the Capacitated Arc Routing Problem. The AOS proposed combines a set of four Fitness Landscape Analysis measures in conjunction with the existing Credit Assignment measure of MAENS* and an online learning algorithm, to predict the most suitable crossover operator. The results achieved by MAENS*-II show that this technique is able to compete with the state-of-the-art techniques and can, in some cases, exploit the multiple measures to outperform the alternative



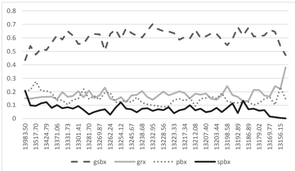
(a) egl-s1-B: MAENS*-II



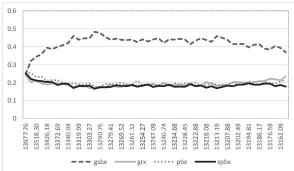
(b) egl-s1-B: MAENS*



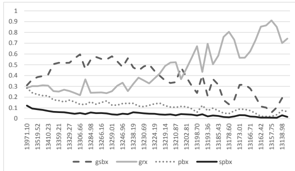
(c) egl-s1-B: oracle



(d) egl-s2-B: MAENS*-II



(e) egl-s2-B: MAENS*



(f) egl-s2-B: oracle

Fig. 1: Crossover operator selection rates on two CARP instances of MAENS* (first column), MAENS*-II (central column) and the oracle (right column)

strategy. This work leaves space for interesting directions that can be explored, such as the adoption of an Average or Extreme Reward strategy, the use of different base learners or the combined use of this Credit Assignment strategy with existing Operator Selection Rules and vice versa.

Acknowledgment

This work was supported by EPSRC (Grant Nos. EP/I010297/1 and EP/J017515/1). Xin Yao was supported by a Royal Society Wolfson Research Merit Award.

References

1. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine learning 47(2-3), 235–256 (2002)
2. Barbosa, H.J., Sá, A.: On adaptive operator probabilities in real coded genetic algorithms. In: Workshop on Advances and Trends in Artificial Intelligence for Problem Solving (SCCC'00) (2000)
3. Beullens, P., Muyldermans, L., Cattrysse, D., Van Oudheusden, D.: A guided local search heuristic for the capacitated arc routing problem. European Journal of Operational Research 147(3), 629–643 (2003)
4. Consoli, P., Yao, X.: Diversity-driven selection of multiple crossover operators for the capacitated arc routing problem. In: Blum, C., Ochoa, G. (eds.) Evolutionary Computation in Combinatorial Optimisation - 14th European Conference, EvoCOP 2014, Granada, Spain, April 23-25, 2014, Revised Selected Papers. pp. 97–108. No. 12 in Lecture Notes in Computer Science, Springer (2014)
5. DaCosta, L., Fialho, A., Schoenauer, M., Sebag, M.: Adaptive operator selection with dynamic multi-armed bandits. In: Proceedings of the 10th annual conference on Genetic and evolutionary computation. pp. 913–920. ACM (2008)

6. Davis, L.: Adapting operator probabilities in genetic algorithms. In: International Conference on Genetic Algorithms '89. pp. 61–69 (1989)
7. Eglese, R.W.: Routeing winter gritting vehicles. Discrete applied mathematics 48(3), 231–244 (1994)
8. Eiben, A.E., Hinterding, R., Michalewicz, Z.: Parameter control in evolutionary algorithms. Evolutionary Computation, IEEE Transactions on 3(2), 124–141 (1999)
9. Fialho, Á., Da Costa, L., Schoenauer, M., Sebag, M.: Dynamic multi-armed bandits and extreme value-based rewards for adaptive operator selection in evolutionary algorithms. In: Learning and Intelligent Optimization, pp. 176–190. Springer (2009)
10. Goldberg, D.E.: Probability matching, the magnitude of reinforcement, and classifier system bidding. Machine Learning 5(4), 407–425 (1990)
11. Golden, B.L., Wong, R.T.: Capacitated arc routing problems. Networks 11(3), 305–315 (1981)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD explorations newsletter 11(1), 10–18 (2009)
13. Hinkley, D.V.: Inference about the change-point from cumulative sum tests. Biometrika 58(3), 509–523 (1971)
14. Julstrom, B.A.: What have you done for me lately? adapting operator probabilities in a steady-state genetic algorithm. proceeding of: Proceedings of the 6th International Conference on Genetic Algorithms, Pittsburgh, PA, USA, July 15-19, 1995 (1995)
15. Kolter, J.Z., Maloof, M.: Dynamic weighted majority: A new ensemble method for tracking concept drift. In: Data Mining, 2003. ICDM 2003. Third IEEE International Conference on. pp. 123–130. IEEE (2003)
16. Lu, G., Li, J., Yao, X.: Fitness-probability cloud and a measure of problem hardness for evolutionary algorithms. In: Evolutionary Computation in Combinatorial Optimization, pp. 108–117. Springer (2011)
17. Lunacek, M., Whitley, D.: The dispersion metric and the cma evolution strategy. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation. pp. 477–484. ACM (2006)
18. Maturana, J., Saubion, F.: A compass to guide genetic algorithms. In: Parallel Problem Solving from Nature–PPSN X, pp. 256–265. Springer (2008)
19. Merz, P.: Advanced fitness landscape analysis and the performance of memetic algorithms. Evolutionary Computation 12(3), 303–325 (2004)
20. Minku, L.L., White, A.P., Yao, X.: The impact of diversity on online ensemble learning in the presence of concept drift. Knowledge and Data Engineering, IEEE Transactions on 22(5), 730–742 (2010)
21. Runarsson, T.P., Yao, X.: Stochastic ranking for constrained evolutionary optimization. Evolutionary Computation, IEEE Transactions on 4(3), 284–294 (2000)
22. Schlimmer, J.C., Granger, R.H.: Beyond incremental processing: Tracking concept drift. In: AAAI. pp. 502–507 (1986)
23. Tang, K., Mei, Y., Yao, X.: Memetic algorithm with extended neighborhood search for capacitated arc routing problems. Evolutionary Computation, IEEE Transactions on 13(5), 1151–1166 (2009)
24. Thierens, D.: An adaptive pursuit strategy for allocating operator probabilities. In: Proceedings of the 2005 conference on Genetic and evolutionary computation. pp. 1539–1546. ACM (2005)
25. Vanneschi, L., Pirola, Y., Collard, P.: A quantitative study of neutrality in gp boolean landscapes. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation. pp. 895–902. ACM (2006)