AN ADAPTIVE HYBRID GENETIC ALGORITHM FOR PAVEMENT MANAGEMENT

João Santos¹, Adelino Ferreira¹ and Gerardo Flintsch²

¹Road Pavements Laboratory, Research Center for Territory, Transports and Environment, Department of Civil Engineering, University of Coimbra, Rua Luís Reis Santos, 3030-788, Coimbra, Portugal
²Center for Sustainable Transportation Infrastructure, Virginia Tech Transportation Institute, Department of Civil and Environmental Engineering, Virginia Polytechnic Institute and State University
Transportation Research Plaza, Blacksburg, VA 24061, USA

ABSTRACT
Throughout the years, Genetic Algorithms (GA) have been successfully applied to tackle the computational complexity of many real-world global optimization problems, such as those faced in determining the optimal long-term maintenance and rehabilitation (M&R) strategies of road pavement sections. However, it is increasingly recognized that pure GA may not be suitable to fine-tune searches in complex combinatorial spaces due to their limited ability to combine, in an optimal way, the exploration of the search space for promising solutions and the exploitation of the best solutions found during the running time of the algorithm. In order to address this drawback, Local Search (LS) techniques have been incorporated into GA to improve the overall efficiency of the search, either by accelerating the discovery of good solutions, for which evolution alone would take too long to find, or by reaching solutions that would otherwise be unreachable by evolution or a local method alone. In this paper, a novel Adaptive Hybrid Genetic Algorithm (AHGA) is proposed which contains two dynamic learning mechanisms to adaptively guide and combine the exploration and exploitation search processes. The first learning mechanism aims to reactively assess the worthiness of conducting an LS, and to efficiently control the computational resources allocated to the application of this search technique. The second learning mechanism uses instantaneously learned probabilities to select, from a set of pre-defined LS operators which compete against each other for selection, which one is the most appropriate for a particular stage of the search to take over from the evolutionary-based search process. The new AHGA is compared to a non-hybridized version of the GA by applying the algorithms to several case studies with the objective of determining the best pavement M&R strategy that minimizes the present value of the total M&R costs. The results show that the proposed AHGA statistically outperforms the traditional GA in terms of efficiency and effectiveness.

KEYWORDS
Genetic algorithms; adaptive local search; hybridization; pavement management; pavement maintenance and rehabilitation costs.

1. INTRODUCTION
In the current economic paradigm, the need to maintain and preserve existing highway infrastructure assets has been at the forefront of the transportation infrastructure decision-makers’ concerns. This is reflected by highway agency budgets shifting funding from new construction and/or reconstruction to maintenance. This shift in priorities is not likely to change over the coming decades, underlining the importance of establishing consistent procedures and developing innovative and optimization-based engineering management systems to achieve effectiveness in: (1) managing huge investments in maintenance and rehabilitation (M&R) of pavement systems; (2) identifying and implementing proven maintenance, rehabilitation and preservation practices and techniques; and (3) ensuring proper timing and intensity of application of those treatments.

Optimization is a broad concept that can be applied at different levels of evaluation and for different categories of infrastructure assets. Pavement M&R is one of the most critical and costly forms of infrastructure asset management that has benefited from the potentialities of the optimization techniques. However, the growing complexity of large-scale problems required to be solved optimally, such as those that the PMS have to deal with, has imposed great obstacles to the efficiency and effectiveness of the traditional
optimization techniques. For instance, the problem of identifying adequate M&R activities for individual pavement segments is usually formulated using integer variables to represent M&R activities selected for individual pavement segments. This problem is a combinatorial one which, due to a huge solution space, is very difficult to solve optimally using the traditional optimization techniques.

Evolutionary algorithms (EA), which is a subfield of artificial intelligence, have demonstrated their effectiveness over the last few decades as powerful optimizers for difficult, nonlinear, multimodal optimization problems (Eiben and Smith, 2010). Genetic algorithms (GA) are a popular type of EA and have been object of considerable attention in the field of infrastructure management (Ferreira et al., 2002; Mathew and Isaac, 2014).

Notwithstanding the advantages recognized due to the stochastic search mechanisms behind the GA and the remaining bio-inspired and space-based EA, there may also be drawbacks with algorithms as follows: (1) long computing time; (2) premature convergence; and (3) limited capacity to fine-tune solutions. Several research studies have found that a skilled combination of EA with local search (LS) heuristics, named “memetic algorithms” (MA) (Moscato, 1989), can improve the performance of EA in terms of efficiency (i.e. requiring orders of magnitude fewer evaluations to find optimal solutions) and effectiveness (i.e. identifying higher quality solutions), especially when dealing with real-world and large scale problems. In the past few years several GA-based MA have been developed in various engineering fields. However, to the best of our knowledge no study exists in the literature that has applied this concept to the pavement management sector.

In this paper, a new adaptive hybrid GA (AHGA) combing GA with an LS mechanism is presented for solving the pavement M&R strategy selection problem. The AHGA framework is provided with a pool of LS operators and an Adaptive Local Search Operator Selection (ALSOS) method to decide dynamically and on-the-fly on the relevance of conducting an LS according to a given search strategy. Online learning probabilities are then used to select both the LS operator from the pool and the LS intensity that leads to the best gains of search efficiency and effectiveness.

2. ADAPTIVE HYBRID GENETIC ALGORITHM FRAMEWORK

The framework of the proposed AHGA is illustrated in simple terms in Figure 1. It features the following main components: (1) the encoding of solutions; (2) the initial population generation; (3) the solutions’ fitness evaluation; (4) the parents selection; (5) the reproduction process; (6) the population replacement process; (7) the stagnation prevention methodology; (8) the iterations stopping criteria; and (9) the adaptive LS mechanism.

In the developed AHGA an integer coding is adopted to represent the M&R alternatives. Each individual represents a potential solution (M&R strategy) and consists of a sequence of $S \times T$ genes, where $S$ is the number of pavement sections considered for analysis, $T$ represents the project analysis period (PAP) defined by the decision-maker, and the allele values for each of these genes represent a possible M&R activity. Posteriorly, an initial population with size $N$ is randomly generated, and the best $N \times Elite\_rate$ individuals are copied and stored in an archive pool according to a user-defined rate ($Elite\_rate$). This scheme prevents solutions of the highest relative fitness from being excluded from the next generation through the nondeterministic selection process.

Once the population has been created at each generation, the individual’s fitness has to be evaluated according to the objective function and constraints corresponding to the features of the problem being tackled. For cost minimization problems, the fitter individual is the one with lower present value of the total M&R costs. Given that GA do not have any explicit constraint-handling mechanism, the application of the traditional genetic search operators, which are “blind” with respect to constraints, may produce infeasible solutions. In this research work, a dynamic and parameter-free penalty approach based on the concept of superiority of feasible solutions was developed and incorporated into the AHGS. In order to determine which solutions of the population will be used by the reproduction operators to generate new solutions, called offsprings, a Ranking-based Selection (RS) method (Chuang et al., 2015) was implemented. It was selected after preliminary experiments revealed the superiority of this method over the traditional tournament selection method.
The reproduction process is carried out through two operators: crossover and mutation. Crossover is the process by which one or more new individuals are created through the combination of genetic material selected from two or more parents of the source population, to form the members (offspring chromosomes) of a successor population. In the proposed AHGA, a Direction-based Crossover (DBX) operator was implemented (Chuang et al., 2015). In turn, the mutation operator aims to introduce new genetic material into an existing individual, ensuring that the full range of allele is accessible for each gene. Thereby, it allows the exploration of different areas of the search space by potentially generating solutions that have never been analyzed while it prevents the search from being trapped in a local optima. In the proposed AHGA each pair of parents which do not meet the crossover criterion will automatically undergo mutation according to the Dynamic Random Mutation (DRM) operator (Chuang et al., 2015).

Once the reproduction has been performed, parents, offspring and elite members (if applicable) are subject to a replacement process to determine which solutions are selected to compose the successor population. In the developed AHGA a replacement-with-elitism methodology was adopted. By this process, each offspring chromosome is directly compared with its parent and the better (fitter) chromosome moves to the next generation. The survivor chromosomes are posteriorly joined by the elite chromosomes initially preserved. In this way, the performance of the algorithm is enhanced by ensuring that the good individuals survive to the next generation.

Due to the evolutionary nature of the GA, it may happen that at some given time the population achieves a low diversity level such that the search process stalls around a local optimum. To avoid this situation, a stagnation prevention methodology was implemented in the proposed AHGA. It consists of refreshing all chromosomes of the population, excepting the current best one, whenever a stagnation index (SI), expressed by the standard deviation of the population’s fitness, falls below a pre-defined convergence threshold value (p). Once the stagnation prevention methodology is triggered, the population is regenerated according to two mechanisms, a random regeneration and a biased regeneration, aiming to strike a balance between the exploration of the search space and the exploitation of the best solution. In the random regeneration, 25% of N individuals are randomly generated in order to introduce some diversity into the genetic material available for generating new offspring chromosomes in the upcoming recombination processes. With respect to the biased regeneration, the best individual is used to construct the remaining individuals by using two especially designed operators.

The implementation of an efficient stopping criterion is an important aspect for any iterative method. If properly designed it may lead to substantial savings in computational times. The proposed AHGA incorporates the following termination criteria: (1) the number of generations attains the user-specified maximum number (Max_gen); and (2) the number of continuous generations without improvement of the best solution attains the user-specified maximum number (Max_gen_NoImprov). A generation is considered to be a no improvement with regard to its predecessor if the difference of the fitness values of their best individuals is inferior to 0.01%.

In the proposed algorithm, a GA with a classic framework without any kind of LS is hybridized with an adaptive LS mechanism that aims to either accelerate the discovery of good solutions, for which evolution alone would take too long to discover, or to reach solutions that would otherwise be unreachable by evolution or a local method alone (Krasnogor et al., 2006). Specifically, the LS is carried out on the current best solutions of a generation based on a best first improvement strategy. That means that the LS stops when the first better neighbor solution is found, up to a user-specified maximum number of attempts (MaxNumLS_iter). If the LS succeeds, the improved solution replaces the starting solution. In turn, if no better solution has been found by the time the LS process is halted, the solution that underwent LS is kept in the archive of best solutions.

To avoid a waste of algorithm resources due to an improper use of eventually expensive LS, the AHGA incorporate a dynamic approach that controls both the LS frequency, i.e. the number of continuous uninterrupted generations that a GA performs before applying LS (El-Mihoub et al., 2006) and the LS intensity, or, in other words, the maximum number of LS iterations allowed for the LS algorithm to get a successful move (MaxNumLS_iter).

The LS frequency is initially set to 1, but after a given number of unsuccessful LS executions, the decision on whether or not to perform LS is made probabilistically according to a user-defined probability (p_minLS). For that purpose, a sliding time window with a user-defined size W_LS is adopted to record the performance of the last W_LS LS operations. When none of the last W_LS LS operations were successful, the execution of the LS at a given time point i is triggered probabilistically.
With respect to the LS intensity, the value of $MaxNumLS\_iter$ at time point $t$ is initially set to $MaxNumLS\_iter_{max}$ and will be linearly reduced according to the consecutive number of unsuccessful LS operations ($UnsucLS\_iter$) up to a user-defined limit value ($MaxNumLS\_iter_{min}$) (Equation 1). The $MaxNumLS\_iter$ is restored to the initial value whenever a LS operation is successful.

$$MaxNumLS\_iter = \frac{MaxNumLS\_iter_{max} - MaxNumLS\_iter_{min}}{W_{LS}} \times UnsucLS\_iter$$

Given the underlying idea in the previous paragraph, a sensible LS design approach would not be based on a priori choice of one single LS operator that may prove to be unproductive for the problem at hand. Rather, a more efficient design would consider the incorporation of multiple LS operators and the decision of which LS operator to apply on a given search moment would be more rational if made dynamically. This system of adaptive LS process promotes both cooperation and competition among various problem-specific LS operators and favors neighborhood structures containing high quality solutions that may be arrived at by low computational efforts (Ong et al., 2006). To this aim, the AHGA framework is provided with a pool of LS operators and an Adaptive Local Search Operator Selection (ALSOS) method in order to decide dynamically and on-the-fly, based on their recent performances, if it is worthy to perform an LS, and if so, to
select the LS operator, from the several available options, that leads to the best gains in search efficiency. The ALSOS method is divided into two main modules: (1) a credit assignment module, which assigns a reward to each LS operator based on their impacts on the progress of the search; and (2) an operator selection module, which selects the operator to apply to the next LS step, based on the credits previously assigned. In the credit assignment module the assessment of the performance of each LS operator, based on the impact of its application on the progress of the search, is carried out by applying the fitness improvement rate (FIR) method (Equation 2):

$$FIR_{t,i} = \frac{f(X^{\text{neighbor}}) - f(X^{\text{initial}})}{f(X^{\text{initial}})}$$

(2)

where $f(X^{\text{neighbor}})$ is the fitness value of the neighbor solution and $f(X^{\text{initial}})$ is the fitness value of the initial solution.

Once the performance is assessed, the reward of each operator is determined according to the Extreme Value approach (Fialho et al., 2008). For that purpose, a sliding window with fixed size $W_{\text{CredAssig}}$ was adopted for each LS operator to store the FIR values resulting from the last $W_{\text{CredAssig}}$ applications of the LS operators. The sliding windows work as a first-in, first-out (FIFO) queue, to the extent that the most recent FIR values are added at the tail of the sliding window, while the oldest ones are removed to preserve the window size. The reward of each LS operator at the point time $t$ is then calculated as the greatest FIR value among the current values stored in the sliding window (Equation 3).

$$rew_k(t) = \max\{FIR_{t,1}, FIR_{t,2}, ..., FIR_{t,W_{\text{CredAssig}}})$$

(3)

where $t$ is the current time point; $FIR_{t}(t)$ is the fitness improvement rate observed at search time $t$; and $rew(t)$ is the expected reward for LS operator $k$ at search time $t$.

As far as the LS operator selection is concerned, it is performed by implementing the Adaptive Pursuit (AP) method (Thierens, 2005). The AP method, originally proposed for learning automata, adopts a winner-takes-all strategy to increase the chance of selecting the best LS operator $k^*$ up to $p_{\text{max}}$ while the remaining probabilities are decreased to $p_{\text{min}}$.

In the proposed AHGA, a set of LS operators were considered to generate several neighbors’ structures, and consequently neighborhood solutions for the problem being tackled. It should be mentioned that the choice of LS operators for the pool is to some extent subjective, as there is no conventional procedure in the literature for guiding the selection of the best set of LS operators for a given problem. Thus, the choice of the LS operators presented in the list below was based on the authors’ knowledge of the problem and on empirical evidence, and resulted from narrowing down an initial extended list. The LS operators are the following: (1) swap mutation (SWM); (2) forward shift mutation (FSM); (3) backward shift mutation (BSM); (4) Cauchy distribution-based mutation (CaDM); (5) chaotic dynamic-based mutation (ChDM); and (6) delete mutation (DM).

3. PARAMETERS SETTING FOR THE PROPOSED AHGA

Before measuring and comparing the performances of the AHGA and traditional GA when applied to different case studies, a parameter tuning campaign based on the Taguchi approach (Roy, 2010) was undertaken to determine the set of parameter values that yields the best algorithm performance. Extensively used for engineering process optimization, Taguchi designs experiments using orthogonal arrays (OAs) to systematically vary and test the different levels of the control factors (i.e. parameter settings). The appropriate levels for those factors are those that make the system more “robust”, or in other words, less sensitive to variations in uncontrollable (noise) factors. To conclude on the robustness of a given process or system, a criterion entitled Signal-to-Noise (S/N) ratio ($\eta_{\text{SN}}$) is adopted, where factor levels that maximize the S/N ratio are optimal. Once all of the S/N ratios have been computed for each run of an experiment, the Taguchi method suggests a graphical approach to analyze the data. According to this approach, the S/N ratios and average responses are plotted for each factor against each of its levels. The graphs are then examined to determine the optimal factor level, i.e. to select the factor level which (1) best maximizes the mean of the S/N rations and (2) minimizes the mean of the average responses.
To concretize the parameter tuning process, the AHGA was applied to a case study consisting of determining the best M&R strategy that minimizes the total discounted M&R costs of a one-way road pavement section of an Interstate highway in Virginia, USA. The parameters (control factors) that were calibrated through the Taguchi method are the following: (1) maximum number of LS iterations \((\text{MaxNumLS}_{\text{itermax}})\); (2) minimum number of LS iterations \((\text{MaxNumLS}_{\text{itermin}})\); (3) probability of performing LS \((p_{\text{minLS}})\); (4) size of the sliding time window used to store the status of the LS operations \((\text{W}_{\text{LS}})\); (5) size of the sliding time window used to store the performance of the LS operators, expressed in terms of FIR \((\text{W}_{\text{CredAssig}})\); (6) adaptation rate \((\alpha)\) considered by the credit assignment module of the ALSOS method; (7) minimum probability of selecting a given operator \((p_{\text{min}})\) considered by the AP method; and (8) learning rate \((\beta)\) also considered by the AP method. Overall, for each parameter three alternative values (levels) were considered based on preliminary tests (Table 1). According to the Taguchi method, for a calibration process with such features, i.e. eight parameters with three alternative values, the L18 OA is recommended for the matrix experiment. The optimal parameter values were then identified by applying the Taguchi’s parameter design approach, according to which the optimal parameter values are the ones that best maximize the mean of S/N ratios and minimize the mean of the average responses, expressed in terms of computational running time (seconds). Table 1 summarizes the three alternative levels considered and the optimal level of the parameters. These parameter values will be used by the AHGA when comparing its performance against that of the non-hybridized version of the GA.

Table 1. Parameters levels initially considered in the calibration procedure and optimal level obtained after applying Taguchi’s parameter design approach

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
<th>Optimal level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(\text{MaxNumLS}_{\text{itermax}})</td>
<td>50</td>
<td>150</td>
<td></td>
<td>L2</td>
</tr>
<tr>
<td>2</td>
<td>(\text{MaxNumLS}_{\text{itermin}})</td>
<td>5</td>
<td>25</td>
<td>45</td>
<td>L3</td>
</tr>
<tr>
<td>3</td>
<td>(p_{\text{minLS}})</td>
<td>5%</td>
<td>20%</td>
<td>50%</td>
<td>L3</td>
</tr>
<tr>
<td>4</td>
<td>(\text{W}_{\text{LS}})</td>
<td>5</td>
<td>15</td>
<td>25</td>
<td>L3</td>
</tr>
<tr>
<td>5</td>
<td>(\text{W}_{\text{CredAssig}})</td>
<td>10</td>
<td>50</td>
<td>100</td>
<td>L3</td>
</tr>
<tr>
<td>6</td>
<td>(\alpha)</td>
<td>0.1</td>
<td>0.5</td>
<td>0.9</td>
<td>L1</td>
</tr>
<tr>
<td>7</td>
<td>(p_{\text{min}})</td>
<td>5%</td>
<td>10%</td>
<td>15%</td>
<td>L3</td>
</tr>
<tr>
<td>8</td>
<td>(\beta)</td>
<td>0.2</td>
<td>0.5</td>
<td>0.8</td>
<td>L2</td>
</tr>
</tbody>
</table>

4. COMPARISON OF THE ALGORITHMS PERFORMANCE

The AHGA is compared with the non-hybrid version of the GA with respect to: (1) its ability to consistently reach the fittest solution achieved by employing the two algorithms; and (2) the convergence behavior during the search process. The fittest solution obtained from the usage of both algorithms is considered the point of comparison of the effectiveness of the algorithms because the problem size (i.e. number of M&R considered and PAP length), the complexity of the pavement performance prediction models (PPPM) and the way they relate to each other make it difficult or even impossible for the analytical or exhaustive optimization approaches to reach the global optimum solution within the span of a human lifetime.

Both algorithms were applied to several case studies with the objective of determining the best pavement M&R strategy that minimizes the total discounted M&R costs of a one-way road pavement section of an Interstate highway in Virginia, USA. With respect to pavement conditions, 16 different scenarios were considered, and for each scenario ten independent computational runs were performed. They differ from each other in the following features: PAP length; initial pavement age; initial pavement critical condition index (CCI); and CCI warning level. To provide a fair basis for the comparison of the performance of both algorithms, they were run considering the same values for the parameters common to both algorithm. In turn, the parameter setting displayed in Table 1 was adopted when specifically applying the AHGA.

In order to examine the statistical difference between the algorithms, two non-parametric tests were carried out: the Wilcoxon signed-rank test (Hollander et al., 2014); and Page’s trend test (Page, 1963). Specifically, the non-parametric Wilcoxon signed-rank test was conducted to compare the algorithms’ final results with the significance level \((\alpha)\) of 5%. In turn, the Page’s trend test was adopted to assess the algorithms’ convergence performance, considering intermediate results instead of just the final results in each case study. This test is applied under the assumption that an algorithm with a good convergence performance
will advance towards the optimum faster than another algorithm with a worse performance (Derrac et al., 2014).

Regarding the performance of each algorithm with respect to their ability to consistently reach the fittest solution obtained by employing the two algorithms, the AHGA was always able to reach the best known solution regardless of the case study considered, whereas the GA did not present this general capacity by failing to converge to the best solution in one of the case studies. With respect to the ability of the algorithms to more consistently achieve the best known solutions, the AHGA was found to almost always converge to the best known solution in 10 out of the 10 computational runs, with the exception of two case studies. By contrast, GA was only able to converge to the best known solution in all of the ten computational runs when it was applied to 6 out of the 16 case studies. In terms of the Wilcoxon signed-rank test results, it was observed that the AGHA does not present an overwhelming superiority over the GA with respect to its capacity to consistently achieve fitter solutions. Indeed, the null hypothesis was rejected in only 4 of the 10 case studies in which differences were observed in the fitness of the best solutions obtained by the algorithms in at least 1 of the 10 computational runs. Therefore, the overall conclusion that can be extracted from the Wilcoxon signed-rank test results, if only the fitness of the best solutions produced by the algorithms were to be analyzed, would be that the two algorithms exhibit a similar behavior. However, this conclusion should not be overemphasized since it strongly depends on the stopping criterion adopted. In fact, if enough time is given to the GA (it seems to be what happened in the case studies analyzed) it will reach a comparable solution to the AHGA, but the point of using the AHGA is not only to obtain high-quality solutions but also to do it in a shorter time. In the next section, the issue of how quick the algorithms are in achieving the best solution to the AHGA, but the point of using the AHGA is not only to obtain high-quality solutions but also to do it in a shorter time. In the next section, the issue of how quick the algorithms are in achieving the best know solutions will be addressed.

In order to assess the statistical significance of the difference of the algorithm’s convergence performance throughout the search process, Page’s trend test was conducted. Table 2 displays the cut-point rankings (r_j) computed for the absolute difference in the objective function value of the best solutions produced by the two algorithms and the summation of all r_j per cut-point (R_j). From the results presented in this table, a Page’s L statistic value of 3182 was computed, which corresponds to a p-value inferior to 0.0001 at a significance level of α=0.05. Thus, given the low p-value the null hypothesis can be strongly rejected. This fact allows us to conclude that the increasing trends in the rankings observed in the last row of the Table 2 are backed up statistically, or, in other words, that the AHGA converges faster than the GA. Therefore, the overall conclusion that can be extracted from the Page’s trend test results is that is the computational running time is a limiting factor, the AHGA may achieve better results in less time through the definition of stopping criteria that terminates the optimization process either when a determined amount of improvement is not achieved after a given number of iterations or when a predefined number of iterations is reached.

<table>
<thead>
<tr>
<th>Case study ID</th>
<th>Cutting-point ID (generation ID)</th>
<th>1 (12)</th>
<th>2 (24)</th>
<th>3 (36)</th>
<th>4 (48)</th>
<th>5 (60)</th>
<th>6 (72)</th>
<th>7 (84)</th>
<th>8 (96)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3.5</td>
<td>3.5</td>
<td>5.5</td>
<td>5.5</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>(R_j)</td>
<td>23</td>
<td>40</td>
<td>48</td>
<td>63</td>
<td>79.5</td>
<td>93.5</td>
<td>107.5</td>
<td>121.5</td>
<td></td>
</tr>
<tr>
<td>(R_j \times 1)</td>
<td>23</td>
<td>80</td>
<td>144</td>
<td>252</td>
<td>397.5</td>
<td>561</td>
<td>752.5</td>
<td>972</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Computation of ranks for Page’s trend test
5. CONCLUSION

This paper presents the development of an AHGA intended to help decision makers in the field of pavement management tackle the optimization problem consisting in minimizing the life cycle M&R costs of a given pavement section throughout its PAP, while keeping the pavement condition above a predefined threshold value, meeting technical constraints and considering deterministic and non-linear PPPM. The proposed algorithm maintains the exploring ability of a traditional GA and improves its exploiting aptitude through the execution of LS operations. Its main novelty lies on the inclusion of a pool of LS operators and the use of an adaptive LS operator selection approach within the framework of a traditional GA. Specifically, a dynamic-based learning mechanism was developed to decide on the worthiness of performing an LS and to automatically select which LS operator should be applied at each instant of the search, while solving the problem, according to how well each of the LS operators included in the pool have recently performed in the same optimization process. After the algorithm parameters had been calibrated using the Taguchi method, its efficiency and effectiveness were compared with those of a traditional GA through its application to several case studies designed to replicate Virginia Department of Transportation’s real-pavement management problems for a pavement section. The outcomes of the comparative experiments undertaken and accordingly supported by statistical tests proved the superiority of the proposed algorithm in consistently converging to the optimum solution while requiring a lower computational running time.

ACKNOWLEDGEMENT

João Santos wishes to thank the Portuguese Foundation of Science and Technology for a personal research grant (SFRH/BD/79982/2011). The authors also thank the ACIV for the financial support for the presentation of this research work in the 5th International Conference on Theory and Practice in Modern Computing.

REFERENCES