Late Acceptance Selection Hyper-heuristic for Wind Farm Layout Optimisation Problem

Hiba Abdulaziz*, Areeg Elnahas*, Alaa Daffalla*, Yossra Noureldien*, Ahmed Kheiri†, Ender Özcan‡
*University of Khartoum, Faculty of Engineering
Department of Electrical and Electronic Engineering
Algamaa Street, P.O. Box 321, Khartoum, Sudan
{hiba.abdalaziz87, areegelnahas, alaashibeika, yossramera12}@gmail.com
†Lancaster University, Department of Management Science
Lancaster University Management School
Lancaster, LA1 4YX, United Kingdom
a.kheiri@lancaster.ac.uk
‡University of Nottingham, School of Computer Science
Jubilee Campus, Wollaton Road, Nottingham, NG8 1BB, United Kingdom
ender.ozcan@nottingham.ac.uk

***Abstract*—Wind is a promising source of renewable energy which can be harvested using wind turbines placed on farms. An efficient wind farm layout achieving various engineering and financial objectives is crucial to ensure the sustainability and continuity of energy production. In this study, a high-level search technique, namely late acceptance selection hyper-heuristic is applied to optimise the layout of wind farms. This approach aims to find the best placement of turbines at a given site, maximising the energy output while minimising the cost at the same time. The computational experiments indicate that the late acceptance selection hyper-heuristic improves upon the performance of a previously proposed genetic algorithm across all scenarios and an iterated local search over the majority of scenarios considering the best solutions obtained by each algorithm over the runs.**

***Index Terms*—Optimization, Wind Energy Generation, Renewable Energy Sources, Heuristic Algorithms, Genetic Algorithms**

I. Introduction

Wind turbine technology is a fast growing source of renewable energy that is being studied and explored from different perspectives. The energy consumption from non-renewable sources has reached its peak, while demand for energy has been gradually increasing during the last two centuries. In addition to the inherent variability and uncontrollability of wind power, there are many practical interdisciplinary problems facing the developers of commercial wind farms, such as infrastructure layout, which affects urban development and routes layout, as well as the sizing of the storage system of the generated energy before it enters the power grid [1].

Solutions to the wind farm layout optimisation problem fall into one of two basic solution approaches: *discrete* and *continuous*. In discrete optimisation, the wind farm land is partitioned and binary values are used to indicate whether a turbine is present or not in a given cell location; whereas in continuous optimisation, the Cartesian coordinates of turbine locations are utilised [2]. The placement of turbines in a given wind farm directly impacts the efficiency of the farm as turbines affect each other's performance due to aerodynamic interactions (wakes). The problem can be described as a single or multiple objective problem, and is usually characterised by the following:

- Land: In previous studies, different models of grid spacing (i.e. the distance between two nodes where a turbine can be placed) were proposed [3]. Other models, such as in [4], allow random placement of turbines.
- Turbine type: Different types of turbines have different diameters, heights, power ratings and speed. Previous studies do not usually take this into account, but they are of great importance and highly significant to build the wind farm [5].
- Wake model: This represents a challenge in the farm layout problem as it is very difficult to accurately depict the wake equations in a mathematical model due to their stochastic behaviour.

Wind farm layout optimisation problem is considered an $\mathcal{NP}$-hard problem as it requires computation time that scales exponentially with the problem size [5]. In addition, introducing other dimensions such as different heights of turbines, construction and maintenance costs of each turbine, electrical grid infrastructure and other constraints increases the search space exponentially. Previous studies applied simple heuristic approaches to solve the wind layout optimisation problem. They showed promising results, but due to the complexity of the problem, metaheuristics and hyper-heuristics demonstrated even better results [6]. In this study, we present a high-level hyper-heuristic approach for solving the wind farm layout optimisation problem.

II. Background

A. Wind Farm Layout Optimisation Problem

In the literature, several algorithms were developed to optimise the wind farm layout. Mosetti et al. [7] modelled the wind

farm as a ten by ten grid square cells where turbines could only be located at the centre of the cells. A genetic algorithm (GA) was used for optimising an objective function that maximises the energy output and minimises investment costs. Grady et al. [8] modified the settings of the GA. In particular, they showed that better quality solutions could be achieved by letting twenty sub-populations grow for 3,000 iterations. In [9], a multi-population GA is employed to optimise the cost of energy. Huang [10] designed a distributed GA that seeks to find a global near-optimal solution that maximises the profit. Gonzlez et al. [11] employed a hybrid method that combined a GA with a local search method to obtain better results. Emami and Noghreh [12] presented a novel approach using the GA in which the coefficients of the multi-objective functions are adjustable. Their work showed significant improvement in comparison with previous work done in the field. Li et al. [13] developed a novel hyper-heuristic method for solving multi-objective WFLOP. In 2014 and 2015, GECCO (Genetic and Evolutionary Computation Conference) organised a competition with the main subject being the wind farm layout optimisation problem. A detailed review of the challenge and a description of the best performing methods can be found in [5], [14].

B. Hyper-heuristics

Despite the fact that traditional heuristic methods showed remarkable success when designed to solve specific problem domains, it is still difficult to generalise them to include new problems or new instances of similar problems. Hyper-heuristics, on the other hand, aim to solve a range of optimisation problems by operating on the space of low level heuristics (LLHs) rather than the space of solutions directly [15].

There are two fundamental categories of hyper-heuristics which are *selection* and *generation* methodologies. Selection hyper-heuristics are used for controlling and mixing a pool of heuristics, while generation methods are for generating new heuristics from predefined components via a train and test process. This work is based on selection hyper-heuristics.

The main components of selection hyper-heuristics are: (i) *heuristic selection*, which chooses a heuristic from a set of heuristics to generate a new solution and it can be adaptive (as in [16], [17]) or non-adaptive with no learning feedback (e.g. simple random selection method); and (ii) *move acceptance*, that decides whether to accept or reject the newly generated solution.

III. Problem Description

Wind farm layout optimisation problem aims to find the optimal distribution of wind turbines in a 2-dimensional plane so that the cost of energy production is minimised, i.e. maximises the energy production and minimises the layout cost, given by Equation 1, while taking into account multiple design considerations such as wind speed, wake effects, wind turbine power curve, safety constraints (i.e. neighbouring turbines cannot be placed within a given distance), topographic constraints, site characteristics, economic constraints, construction and maintenance cost, cable and road optimisation, turbines features, human constraints (e.g. noise) and existence of obstacles (i.e. wind turbines cannot be installed on some specific areas of the grid) [5]. If we assume that the wind is unidirectional, the number of cells in a given wind farm is determined by considering a square site in which the distance between adjacent grid points in a horizontal or vertical manner equals to 8 times the turbine radius.

The evaluation function used in this study is as follows [5]:

$$\frac{\left(c_t * n + c_s * \lfloor \frac{n}{m} \rfloor\right) * ES + c_{OM} * n}{\left(1 - (1+r)^{-y}\right)/r} * \frac{1}{8760 * P} + \frac{0.1}{n} \quad (1)$$

where

$(c_t * n + c_s * \lfloor \frac{n}{m} \rfloor)$ is the construction cost,
$(ES = \frac{2}{3} + \frac{1}{3} * e^{-0.00174n^2})$ is the economies of scale,
$(c_{OM} * n)$ is the yearly operating costs,
$((1 - (1+r)^{-y})/r)$ is the interests,
$(1/(8760 * P))$ is the annual power output,
$(0.1/n)$ is the farm size coefficient,
$c_t = \$750\text{K}$ is the turbine cost,
$c_s = \$8\text{M}$ is the substation cost,
$m = 30$ is the number of turbines per substation,
$r = 3\%$ is the interest rate,
$y = 20$ years is the lifetime of the farm,
$c_{OM} = \$20\text{K}$ is the annual operation and maintenance costs,
n is the number of turbines in the layout,
P is the energy output of the layout.

IV. Late Acceptance Hyper-heuristic

Late acceptance is a recently proposed metaheuristic by Burke and Bykov [18], [19]. This rather new local search strategy has shown quite promising performance. Unlike simple hill climbing algorithm which compares the newly generated solution at each iteration with the previous solution, this method compares the newly generated solution with a solution generated L iterations ago (see Algorithm 1).

The proposed late acceptance selection hyper-heuristic approach works by invoking the WindFLO platform[1] which encapsulates the problem-specific information, such as the evaluation function computing the cost of energy. Our approach discretises the site into a number of cells, and solutions to the problem are represented as an array of boolean to decide the presence or absence of wind turbines in the cells. The seven low level heuristics implemented in this work are [6]:

- **LLH1** select a cell at random and invert its value
- **LLH2** swap two randomly selected cells
- **LLH3** ruin 10% of cells and reconstruct at random.
- **LLH4** ruin 30% of cells and reconstruct with all zeros or ones
- **LLH5** first improvement local search which searches for the first better quality solution from adjacent solutions

[1] https://github.com/d9w/WindFLO

Algorithm 1: Late acceptance selection hyper-heuristic

Let L be the method parameter;
Let f be a vector of size L;
Let I be the iteration limit;
Let $S_{current}$ be the current solution;
Let S_{new} be the new solution;
Let S_{best} be the best solution;
Let H represent the set of LLHs;
$S_{current} \leftarrow$ InitialSolution();
$S_{best} \leftarrow S_{current}$;
for $i = 0;\ i < L;\ i$++ **do**
    $f(i) \leftarrow Evaluate(S_{current})$;
end
for $i = 0;\ i < I;\ i$++ **do**
    $h \leftarrow$ SelectHeuristicRandomly(H);
    $S_{new} \leftarrow$ ApplyHeuristic($h, S_{current}$);
    if $Evaluate(S_{new}) \leq f(i \bmod L)$ **then**
        $S_{current} \leftarrow S_{new}$;
    end
    if $S_{current} < S_{best}$ **then**
        $S_{best} \leftarrow S_{current}$;
    end
    $f(i \bmod L) \leftarrow Evaluate(S_{current})$;
end
return S_{best};

- **LLH6** select two rows at random and exchange with a crossover rate of 20%
- **LLH7** do the same as LLH6 except that it selects two columns

V. Results

The experiments were run on two Windows OS machines: an i7-7500 Intel processor at 2.7GHz with 8.00GB RAM; and an i5-2410M Intel processor at 2.30GHz with 6.00GB RAM. A visual representation of the five problem scenarios used to measure the performance of the proposed selection hyper-heuristic method is shown in Figure 1[2].

We study the performance of the proposed hyper-heuristic (LA), formed by combining simple random selection method with late acceptance method, and compare its performance with an iterated local search approach (ILS) [6]. Having LLH5 local search method (hill climbing) as one of the LLHs, the implementation of the ILS method is made by combining simple random selection method with improve or equal move acceptance. ILS selects a perturbative low level heuristic (i.e. LLH1-LLH4, LLH6-LLH7) with a probability of 86%, and LLH5 can be performed at a given step with a probability of 14%. The WindFLO repository provides an implementation of a simple genetic algorithm (GA) whose performance is also compared to the LA and ILS approaches.

[2]These scenarios are available at the WindFLO repository

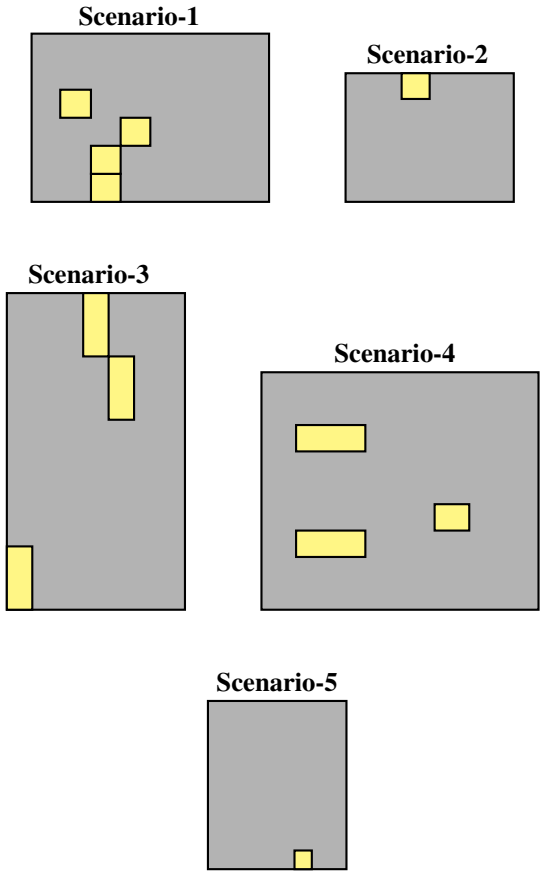

Fig. 1. The five problem scenarios used in this study. The figure shows the terrain sizes; and the boxes in yellow represent obstacles where turbines cannot be placed

LA, ILS and GA are applied to the five problem scenarios each for five runs, and the search terminates whenever 2000 layout evaluations is reached. The performance of each method is measured using the cost of energy described in Section III. In our experiments, we choose a relatively small value of $L = 3$ in LA method. This value is chosen after light experimentation. Table I outlines the results. It shows that the proposed late acceptance selection hyper-heuristic outperforms GA in all problem scenarios. The results also indicate that LA performs slightly better than ILS in three scenarios.

Further analysis showed that LA tends to provide better quality solution than the best solution obtained by GA in just 39 iterations (see Table II).

Figure 2 depicts the change of the cost of energy over time. It can be seen that there is a continuous improvement in the cost of energy until the 500th iteration of the search, after which there is slight improvement. This indicates the ability of the proposed late acceptance method in finding better solutions with longer run times.

Figure 3 shows the utilisation rate of each LLH considering the invocations that generate better than the best solution in hand. All LLHs contribute in improving the best solution in hand with LLH6 achieving most of the improvements.

TABLE I
SUMMARY OF RESULTS. BEST VALUES ARE SHOWN IN BOLD

Scenario	LA			ILS			GA		
	Best	Avg	Std	Best	Avg	Std	Best	Avg	Std
1	0.00123754	0.00123783	1.99E-07	**0.00123743**	**0.00123773**	3.23E-07	0.00126400	0.00126535	8.05E-07
2	**0.00110051**	**0.00110342**	14.74E-07	0.00110361	0.00110383	1.73E-07	0.00115669	0.00115866	10.74E-07
3	**0.00066554**	**0.00066566**	1.07E-07	0.00066566	0.00066576	0.86E-07	0.00069128	0.00069461	25.41E-07
4	**0.00069484**	0.00069505	1.77E-07	0.00069488	**0.00069500**	0.81E-07	0.00071219	0.00071375	9.40E-07
5	**0.00121396**	**0.00121427**	2.65E-07	0.00121402	0.00121443	5.42E-07	0.00124626	0.00126395	89.11E-07

TABLE II
AVERAGE NUMBER OF ITERATIONS REQUIRED BY LA TO ENHANCE THE VALUE OF THE BEST SOLUTION OBTAINED BY GA OVER 5 RUNS

Scenario	GA	LA
	Cost of energy	Average number of iterations
1	0.00126400	**20**
2	0.00115669	**24**
3	0.00069128	**27**
4	0.00071219	**19**
5	0.00124626	**39**

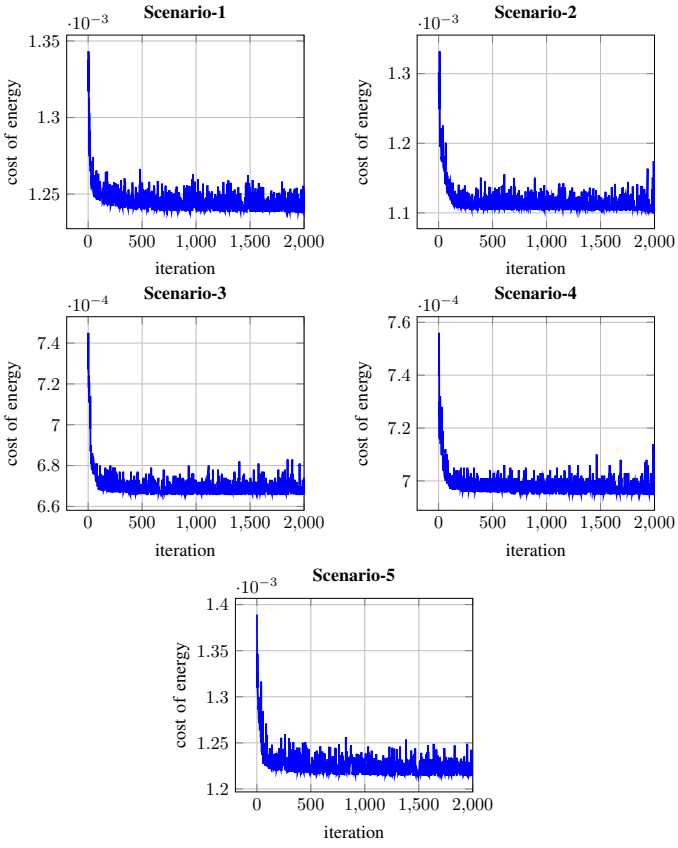

Fig. 2. Cost of energy changes over time

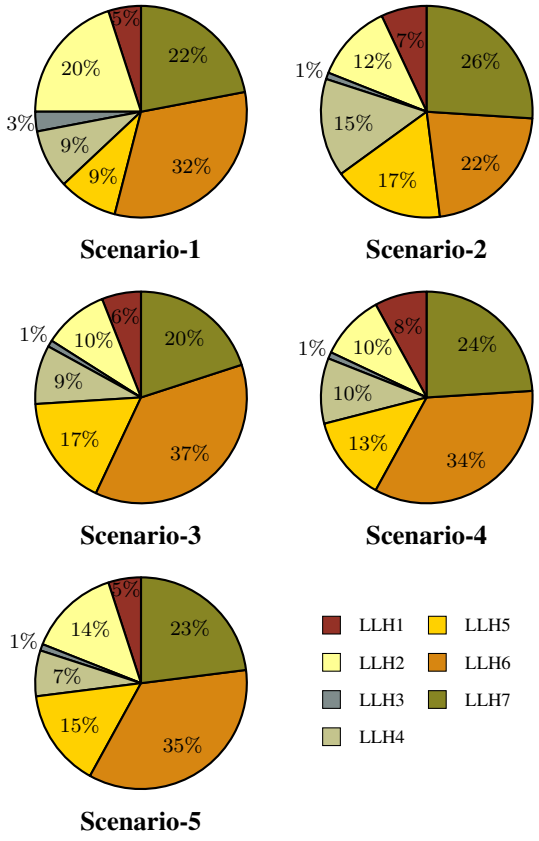

Fig. 3. Utilisation rate of each low level heuristics

VI. CONCLUSION

In this study, we have introduced a hyper-heuristic approach to solve the problem of wind farm turbine placement, combining simple random heuristic selection with late acceptance. This move acceptance method ameliorates the results as it accepts worsening moves time to time, preventing it to get stuck on local minima, hence producing better layouts in the long run. We have compared the performance of our approach to two previously proposed metaheuristics, GA and ILS using a well-known benchmark of five scenarios. The proposed method significantly outperformed the GA in all test scenarios. It was also shown that the method generally performs better than an ILS approach, achieving better costs in three scenarios. To get a better perspective of where this approach stands, comparing it to the wining algorithms of the second edition of the

Wind Farm Layout Optimisation Competition [5], it comes in the fifth place (out of 6). As a future work, we will investigate other hyper-heuristic components, including heuristic selection and move acceptance along with an additional set of new LLHs for further performance improvement.

REFERENCES

[1] S. Dutta and R. Sharma, "Optimal storage sizing for integrating wind and load forecast uncertainties," in *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*. IEEE, 2012, pp. 1–7.

[2] H. Zhang, H. Li, and C. Tam, "Particle swarm optimization for resource-constrained project scheduling," *International Journal of Project Management*, vol. 24, no. 1, pp. 83–92, 2006.

[3] A. Mittal, "Optimization of the layout of large wind farms using a genetic algorithm," Ph.D. dissertation, Case Western Reserve University, 2010.

[4] A. Kusiak and Z. Song, "Design of wind farm layout for maximum wind energy capture," *Renewable energy*, vol. 35, no. 3, pp. 685–694, 2010.

[5] D. Wilson, S. Rodrigues, C. Segura, I. Loshchilov, F. Hutter, G. L. Buenfil, A. Kheiri, E. Keedwell, M. Ocampo-Pineda, E. Özcan, S. I. V. Pena, B. Goldman, S. B. Rionda, A. Hernandez-Aguirre, K. Veeramachaneni, and S. Cussat-Blanc, "Evolutionary computation for wind farm layout optimization," *Renewable Energy*, vol. 126, pp. 681–691, 2018.

[6] A. Kheiri, A. Daffalla, Y. Noureldien, and E. Özcan, "Selection hyper-heuristics for solving the wind farm layout optimisation problem," in *Proceedings of the 8th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA2017)*, 2017, pp. 393–395.

[7] G. Mosetti, C. Poloni, and B. Diviacco, "Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 51, no. 1, pp. 105–116, 1994.

[8] S. Grady, M. Hussaini, and M. Abdullah, "Placement of wind turbines using genetic algorithms," *Renewable Energy*, vol. 30, no. 2, pp. 259–270, 2005.

[9] X. Gao, H. Yang, and L. Lu, "Study on offshore wind power potential and wind farm optimization in hong kong," *Applied Energy*, vol. 130, pp. 519–531, 2014.

[10] H. S. Huang, "Distributed genetic algorithm for optimization of wind farm annual profits," in *2007 International Conference on Intelligent Systems Applications to Power Systems*, 2007, pp. 1–6.

[11] J. S. Gonzalez, A. G. G. Rodriguez, J. C. Mora, J. R. Santos, and M. B. Payan, "Optimization of wind farm turbines layout using an evolutive algorithm," *Renewable Energy*, vol. 35, no. 8, pp. 1671–1681, 2010.

[12] A. Emami and P. Noghreh, "New approach on optimization in placement of wind turbines within wind farm by genetic algorithms," *Renewable Energy*, vol. 35, no. 7, pp. 1559–1564, 2010, special Section: IST National Conference 2009.

[13] W. Li, E. Özcan, and R. John, "Multi-objective evolutionary algorithms and hyper-heuristics for wind farm layout optimisation," *Renewable Energy*, vol. 105, pp. 473–482, 2017.

[14] D. G. Wilson, S. Rodrigues, C. Segura, I. Loshchilov, F. Hutter, G. L. Buenfil, A. Kheiri, E. Keedwell, M. Ocampo-Pineda, E. Özcan, S. I. V. Pea, B. Goldman, S. B. Rionda, A. Hernndez-Aguirre, K. Veeramachaneni, and S. Cussat-Blanc, "Summary of evolutionary computation for wind farm layout optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '18. New York, NY, USA: ACM, 2018, pp. 31–32.

[15] E. K. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and R. Qu, "Hyper-heuristics: a survey of the state of the art," *Journal of the Operational Research Society*, vol. 64, no. 12, pp. 1695–1724, Dec 2013.

[16] A. Kheiri and E. Keedwell, "A hidden markov model approach to the problem of heuristic selection in hyper-heuristics with a case study in high school timetabling problems," *Evolutionary Computation*, vol. 25, no. 3, pp. 473–501, 2017.

[17] A. Kheiri and E. Özcan, "An iterated multi-stage selection hyper-heuristic," *European Journal of Operational Research*, vol. 250, no. 1, pp. 77–90, 2016.

[18] E. K. Burke and Y. Bykov, "A Late Acceptance Strategy in Hill-Climbing for Exam Timetabling Problems," in *PATAT '08 Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling*, 2008.

[19] ——, "The late acceptance hill-climbing heuristic," *European Journal of Operational Research*, vol. 258, no. 1, pp. 70 – 78, 2017.