

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/266186719>

OPTIMIZATION ALGORITHMS FOR OFFSHORE WIND FARM MICROSITING

Article

CITATIONS

19

READS

311

3 authors, including:



[James Manwell](#)

University of Massachusetts Amherst

143 PUBLICATIONS 7,027 CITATIONS

[SEE PROFILE](#)



[Jon G. McGowan](#)

University of Massachusetts Amherst

183 PUBLICATIONS 6,532 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Wind/Diesel Research at the Renewable Energy Research Lab of U. of Massachusetts, Amherst [View project](#)

OPTIMIZATION ALGORITHMS FOR OFFSHORE WIND FARM MICROSITING

C.N. Elkinton*, J.F. Manwell, J.G. McGowan

Renewable Energy Research Laboratory
Dept. of Mechanical and Industrial Engineering
University of Massachusetts
160 Governors Dr., Amherst, MA 01003
*celkinto@ecs.umass.edu

ABSTRACT

In the United States, offshore wind energy is poised to facilitate substantial growth in wind energy production. Unlike most onshore projects, this growth has the potential to occur in close proximity to large load centers (New York, Boston, Houston, for example). In order for offshore wind to be able to compete with other energy generating technologies, however, further reductions in the cost of energy are required. Making optimal use of current technology is one simple approach to this problem. As part of a larger project focused on offshore wind farm analysis and optimization, this research examines the use of optimization algorithms for wind farm micro-siting.

The paper starts with a discussion of five different types of optimization algorithms: gradient search, heuristic, pattern search, simulated annealing, and evolutionary algorithms. The relevance of each algorithm to wind turbine micro-siting is evaluated by considering two separate objectives: minimization of the levelized production cost and maximization of the energy production. Two algorithms, the genetic and greedy heuristic, are further evaluated for the specific case of offshore wind farm design through the use of design simulations. In these simulations, a full set of site conditions are considered, including as water depth, soil conditions, wind climate, and distance from shore. In addition, comparisons are made with previous studies in the literature. Finally, these algorithms are employed to optimize the layout of a potential, real-world offshore wind farm near Hull, Massachusetts. This process, results, and lessons learned are discussed.

1. OVERVIEW

The problem of optimizing wind farm layouts falls into the class of problems called *combinatorial optimization*. Several papers and books have been written about this class of problem, including multiple papers related to wind farm layout. All of these papers have been concerned with onshore wind farm design, but provide a good starting place and sources of comparison for this investigation of offshore wind farm layouts [1, 2, 3, 4].

The objective of this study is to investigate the use of optimization algorithms for offshore wind farm micro-siting. Five optimization algorithms are considered and the applicability of each to wind farm micro-siting is evaluated. The two most promising algorithms are evaluated further. These two algorithms have been incorporated into the offshore wind farm layout optimization (OWFLO) software [5] and are used to lay out farms. Comparisons are made with optimization results from previous studies in the literature.

This paper represents work at the confluence of two larger projects. The OWFLO project is focused on the modeling of the cost of energy (COE) for offshore wind farms. The optimization techniques discussed here use a COE model to optimize the wind farm layout such that the COE is minimized. The focus of the second project is the design of a small offshore wind farm for the town of Hull, Massachusetts. This offers a real-world case study with which to test the OWFLO optimization and modeling techniques. A description of the Hull project, the preliminary optimization results, and a discussion of the lessons learned are included at the end of this paper.

2. INTRODUCTION TO ALGORITHMS

Five different types of algorithms for wind farm optimization are considered in this paper. A brief description of each is given below and additional information can be found in [6, 7].

Gradient Search Algorithm

Gradient search algorithms (GSA) calculate the gradient of the objective function then search in the direction that optimizes the problem. For example, in a minimization problem, the algorithm would search in the direction of the negative gradient. These algorithms find local optima quickly but, without the introduction of some element of randomness, have a difficult time finding global optima. While the GSA itself is generally fast, it is usually necessary to use a different algorithm to determine the starting point. By itself, the GSA usually produces lower quality solutions.

Greedy Heuristic Algorithm

Heuristic algorithms progress from an initial solution by trying different variations on that solution. For example, in a wind farm layout optimization, turbines are added, removed, and each turbine is moved a short distance from its previous location. Each of these changes comprises a potential layout and each is evaluated. After all the potential layouts have been created, the best layout is selected and the process repeats. This selection of the best layout every iteration is what makes the greedy heuristic algorithm (GHA) greedy. GHAs are generally fast but yield solutions of lower quality.

Genetic Algorithm

Genetic algorithms (GAs) use the concepts of evolution and natural selection to search for the optimal solution. For the case of wind farms, the farm space is converted to a grid of potential turbine locations. For every potential solution, each cell will contain either a '0' (no turbine) or a '1' (turbine). The potential solution is coded into a binary array of 0s and 1s. Many different potential solutions are grouped together into a population. Members of this population are selected to be parents based on their fitness within the population and the next generation is produced by combining genetic information from the parents. The children are also subjected to genetic mutation. Finally, some combination of the parents and children become the next generation and the process repeats. The more individuals in each generation, the better the final solution, but the longer the processing time.

Simulated Annealing Algorithm

Simulated annealing algorithms (SAAs) get their name because they are modeled on the process of metal annealing. They employ a 'temperature' function that controls the accuracy of the

process. In an analogy with the annealing of metals, the temperature is high when the algorithm starts and decreases as time (iterations) goes on. The algorithm works by focusing on better solutions as the temperature decreases. The higher the temperature, the higher the probability that an inferior solution is to be chosen, which allows for a quick but thorough search through the entire solution space. As the temperature cools, superior solutions become more likely to be chosen and the algorithm eventually converges to the optimum. By allowing inferior solutions to be investigated, a SAA may arrive at a better solution than a GHA or GA but, due to the temperature-dependent probability function, the computation time may be longer.

Pattern Search Algorithm

The pattern search algorithm (PSA) evaluates a set of potential solutions following an established pattern. The pattern dictates how the algorithm walks through the solution space. After every iteration, the best solution becomes the starting point for the next pattern. The scale of the pattern (the length of the exploratory step) depends on the success of the previous iteration. If a better solution has been found, the scale remains the same. If no improvement was obtained, the scale decreases and the solutions moves towards convergence. There is the potential for premature convergence with PSAs and some element of randomness may be needed to avoid local optima. These algorithms usually converge relatively quickly.

Each of these algorithms can be placed on a spectrum that describes the trade-off between the time the algorithm takes to arrive at a solution and the quality of that solution. This spectrum is shown in Figure 1.



Figure 1. Speed vs. quality spectrum, after Cagan et al. [7].

3. OBJECTIVE FUNCTION

The choice of an appropriate objective function is critical to solving a difficult optimization problem like wind farm layout. Simple objective functions can be evaluated quickly but may not yield accurate or meaningful results, while complex objective functions can be very time-intensive. The objective function developed for this study is a balance of these two extremes.

Appropriate candidates for optimization objectives include:

- ▶ maximization of energy production—this approach has been taken by several of the commercially available optimization software packages. This approach delivers the most energy to the grid, but does not account for the cost of that energy.
- ▶ maximization of profit—this approach was used in [2] and yields the greatest benefit to the farm owner and investors. The disadvantage of this approach is that it requires

knowledge of the value of the electricity on the grid and the inclusion of tax incentives, renewable energy credits, and any other economic incentives.

- minimization of the cost of energy—this approach was used in [1, 3] and is a balance between maximum energy and minimum cost. It results in the best solution from the standpoint of the ratepayer.

For this work, the objective that has been selected is the minimization of the cost of energy (COE). Not only does this result in the lowest COE for the ratepayers, but it represents a balance between the costs and benefits of the farm. The two previous studies that used this approach [1, 3] showed that this technique has good potential for wind turbine placement. The COE models used in these previous papers were highly simplified. The first step of the OWFLO project was to develop a more realistic COE estimation (this has been described in earlier papers [5, 8]), allowing the merits of various layouts to be compared using realistic COE values.

The formulation of the COE used here is the levelized production cost (LPC), which is the cost to the wind farm owner/operator of producing electricity. It represents the break-even price for the electricity and is not influenced by tax or other economic incentives, and thus is a good choice for an objective comparison of the costs and benefits of multiple layouts. The LPC has units of \$/kWh and is formulated as

$$LPC = \frac{C_{inv}}{a \cdot E_a} + \frac{C_{O\&M}}{E_a} \quad (1)$$

where C_{inv} is the total investment cost, $C_{O\&M}$ is the annual operation and maintenance (O&M) cost, a is the annuity factor, and E_a is the annual energy production of the farm. The details of the utilization of the LPC as an objective function are given in [9].

The annual energy estimation includes wake losses, electrical transmission losses, and the availability of the wind farm. As was the case in the earlier studies, the wake model derived by Jensen [10] and refined by Katic et al. [11] has been used:

$$Deficit = \left(1 - \frac{U}{U_0}\right) = \frac{2b}{\left(1 + k \frac{x}{R}\right)^2} \quad (2)$$

where U is the wind speed at the location of interest, U_0 is the free-stream wind speed, b is the axial induction factor, k is the wake entrainment and spreading constant, x is the distance down stream, and R is the radius of the rotor.

A previous phase of the OWFLO project focused on the development of economic models for the major cost components of an offshore wind farm: the rotor-nacelle assembly, the support structure, the electrical interconnection, the installation and maintenance, and the O&M costs [5]. These models are now utilized as part of the objective function for each wind farm design that

the optimization algorithm considers. The inputs to the models include site conditions such as water depth, distance from shore, and soil conditions; turbine specifications; and electrical requirements such as collection and transmission voltages.

4. APPLICATION OF ALGORITHMS

From the five algorithms, the GA and GHA have been selected for implementation. These two algorithms have been selected because they are straight-forward to implement and because they can be compared directly with previous work found in the literature, thereby adding to the validation the methods. These two algorithms represent both ends of the quality vs. speed spectrum—the GA is a slow but more accurate algorithm, while the GHA is fast but less accurate. As will be shown, these two algorithms also work well in series combination. This section describes these two algorithms in greater detail and explains some of their implementation options.

One of the keys to success of algorithms that solve problems with complex solution spaces is an element of randomness that helps to avoid premature convergence. The GA, GHA, PSA, and SAA all utilize such random elements. Initial evaluation of the algorithms suggests that the PSA and SAA are also well suited to the complex problem of wind farm micrositeing, but they have yet to be implemented here. The complex nature of the solution space, due to the wake interactions, makes the GSA less well suited to this problem, as it does not contain any random behavior. This lack of randomness often leads to convergence to a local minimum, but not to the global minimum that is sought.

Genetic Algorithm

The principle behind the GA is the Darwinian concept of natural selection. The most fit *individuals* within a *population* of organisms are more likely to reproduce than individuals who are less fit. As the fittest individuals reproduce and pass their genetic information on to the next generations, the “best” genetic traits tend to propagate while less fit traits do not. As this process continues, the better genes dominate and the overall fitness of the population increases.

The first step in applying this process to an engineering optimization problem is to define the genetic structure of an individual. For the case of wind farm layout, an individual is a layout and a genetic sequence is created using 0's and 1's to describe the location of turbines within the entire farm. The farm is first overlaid with an m -by- n grid where the center of each grid cell represents one of the $(m \times n)$ possible locations for a turbine. This grid, or matrix, can be translated into an array that is 1-by- $(m \times n)$ long. Every cell in the farm is represented by one array element. The value of an element can either be 0 if it does not contain a turbine or 1 if it does. An example of a wind farm and its genetic representation are shown in Figure 2.

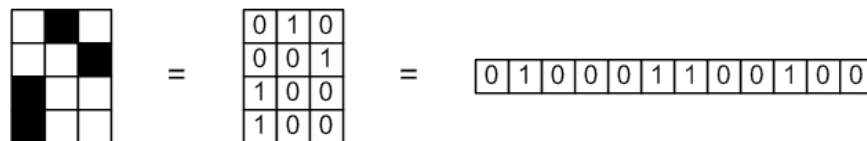


Figure 2. Example of a wind farm and the genetic representation of that wind farm. Black squares and 1's indicate cells occupied by turbines. White squares and 0's indicate empty cells.

An overview of GA is shown in Figure 3. As the figure shows, GA is an iterative process that repeats until the optimization criteria are satisfied. Each step in the genetic manipulation is described in detail below. For an even more thorough explanation of GA, see Sait and Youssef [6] and Pohlheim [12].

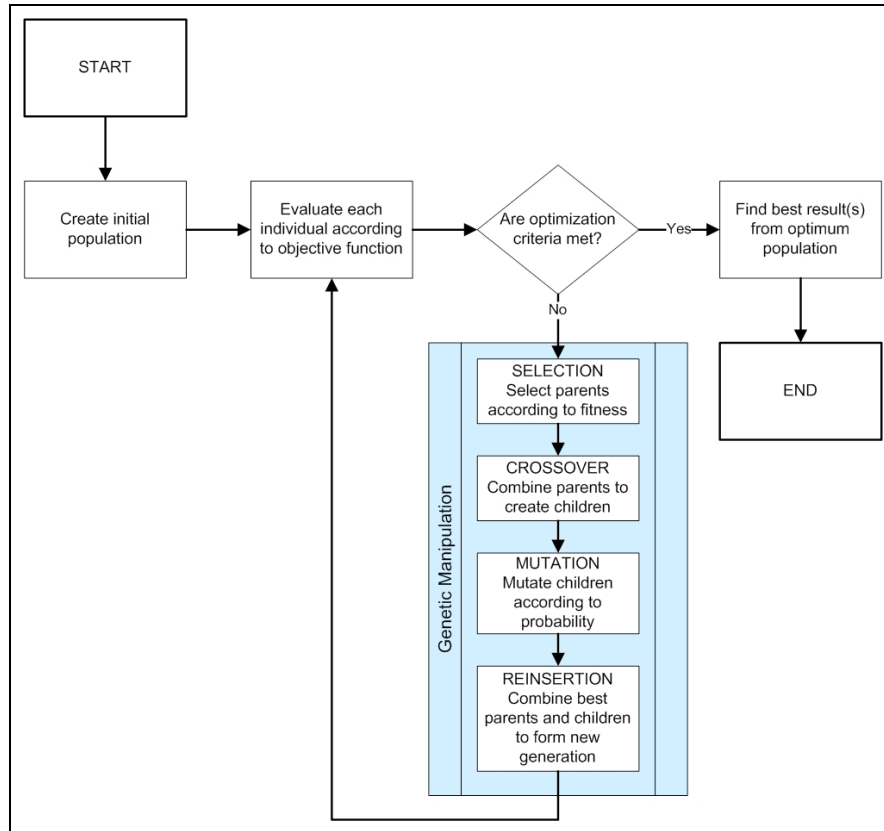


Figure 3. Overview of the steps taken by a genetic algorithm.

Selection

Individuals are *selected* from the population to become parents so that they can combine their genetic information and create the next generation. The selection process is performed based on the fitness of each individual—the more fit the individual, the more likely it is to be selected. Many different selection methods have been developed and the reader is encouraged to read Pohlheim [12] for further information.

The selection method chosen for use in the OWFLO project is known as the *roulette wheel* selection method [6, 12]. Each individual occupies a pie-piece-shaped section of the wheel. The size of each pie piece is determined by the fitness of the individual. The more fit the individual, the larger the pie piece and, thus, the greater the likelihood that that individual will be selected. Each time the wheel is spun, a parent is chosen. An individual may be selected multiple times.

In order for this method to be useful, a value other than the value of the objective function must be used to determine the relative sizes of the pie pieces. The function used for this purpose is

called the *fitness function*. The purpose of a fitness function is to assign a value to each individual that is based on the objective value for that individual, but is scaled to better match the rest of the population. If the objective value itself were used, highly fit individuals could quickly overpower average individuals and this could lead to premature convergence. By applying a fitness function, this problem can be averted.

There are several different ways to define the fitness function. The linear fitness function used in the OWFLO project was suggested by Pohlheim [12] and is given in Equation (3).

$$Fitness(Rank) = 2 - SP + 2 \cdot (SP - 1) \cdot \frac{(Rank - 1)}{(N_{ind} - 1)} \quad (3)$$

The fitness of an individual is based on its rank within the population (as determined by the objective function) and the selective pressure, SP . The selective pressure is the ratio of the probability that the most desirable individual will be selected to the average probability that any individual will be selected. The allowable values of the selective pressure are between 1 and 2.

The other parameter required for selection is the number or percentage of individuals from one generation that are to survive into the next generation. An individual may be selected to be a parent multiple times, so all of the parents may not be unique.

Crossover

Once the parent individuals have been selected, the *crossover* operation is employed to impart their genetic information to the children. The only parameter needed for crossover is the number of crossover points, which is usually a small integer on the order of 1.

Every pair of parents, paired at random from all of the parents, creates two children, as illustrated in Figure 4. In this example there is only one crossover point and the first child obtains the first portion of its genetic information from the first parent until that crossover point is reached. After the crossover point, the first parent gives its genetic information to the second child. If multiple crossover points have been chosen, the order would flip again for each point. All of the genetic information from the two parents is transferred to the children, but because of the crossover, the children are not exact duplicates of the parents.

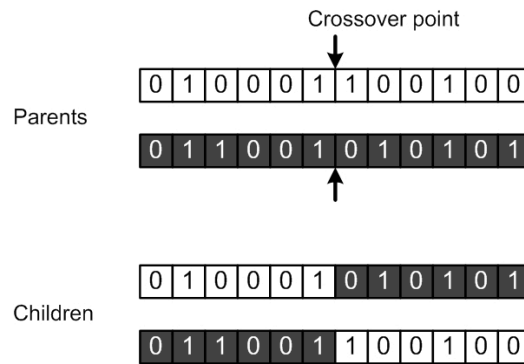


Figure 4. Crossover: the mixing of genetic information to create two new individuals.

Mutation

Genetic *mutation* is the method by which random alterations are made to the population. This is an important part of the GA because without it, no new information would be allowed to enter the population, beyond that which is contained in the initial generation. Mutation is applied to newly created children and typically occurs infrequently—Mosetti et al. [1] suggested that the probability that an individual will undergo mutation should be between 1% and 10%.

If an individual is chosen to undergo mutation, the value of one or more *bits* (each 0 or 1 is called a bit) is inverted. If the value of the bit was 0, it becomes 1 and vice versa. This process is illustrated in Figure 5.

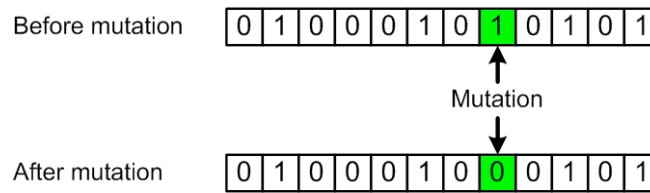


Figure 5. Mutation: the reversal of one or more bits in a genetic code.

Reinsertion

The next generation of individuals will be comprised of parents and children and is created by the process of *reinsertion*. Several reinsertion schemes exist, including [12]:

- Pure reinsertion – Each pair of parents produces two children and only the children survive into the next generation. This scheme is simple to implement, but runs the risk of replacing fit individuals with less fit individuals.
- Uniform reinsertion – Each child replaces, uniformly at random, one parent. Again, this scheme risks replacing a fit parent with a less fit child.
- Elitist reinsertion – Fewer children than parents are produced and are used to replace the least fit parents. This has the advantage of replacing only the least fit parents, but allows less fit children to replace them.
- Fitness-based reinsertion – More children than are required are produced, but only the most fit children are combined with some percentage of the parents to form the next generation. The advantage of this scheme is that only the most fit parents and children survive to the next generation.

Pohlheim [12] recommends a combination of the elitist and fitness-based reinsertion schemes because the most fit individuals survive to create new generations, yet enough new information is allowed to enter that genetic diversity is maintained and the population does not stagnate or converge prematurely. An illustration of fitness-based elitist reinsertion is shown in Figure 6.

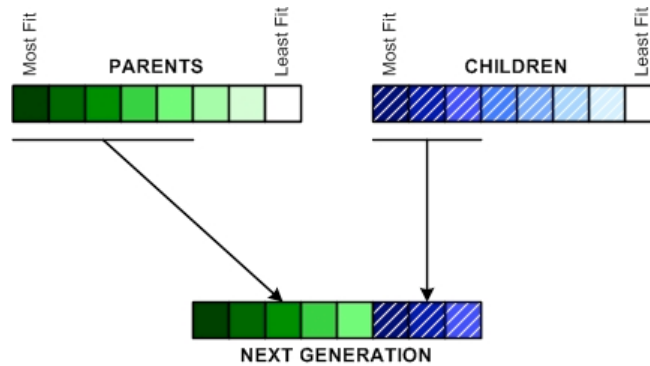


Figure 6. Fitness-based elitist reinsertion: the combination of the most fit parents and children to form the next generation.

Greedy Heuristic Algorithm

The GHA begins with an initial guess, or seed. This seed undergoes a series of modifications that result in additional potential solution guesses. These modifications are: add, remove, and move.

- ▶ *Add*: The *Add* operation adds a specified number of new turbines, one by one, to the layout. For each new turbine, the LPC is evaluated. The addition of new turbines may be subject to the constraints of a maximum number of turbines, maximum power rating of the farm, or maximum energy production.
- ▶ *Remove*: The *Remove* operation removes each turbine in the farm, one by one. The LPC is calculated each time. The number of turbines in the farm may again be subject to potential constraints, including a minimum number of turbines, a minimum power rating, or a minimum energy production.
- ▶ *Move*: This operation moves a specified percentage of the turbines some distance, using a specified step size, from their original positions. For each step, the LPC is calculated.

After each modification, the LPC of each potential solution is evaluated. At the end of each iteration, the best (hence “greedy”) of the potential solutions is chosen to become the seed for the next iteration. If the solution stagnates, a perturbation (adjustment of some of the turbine locations) is performed. If, after a specified number of unsuccessful perturbations, the solution remains unchanged, that solution is considered to be the optimal solution. Figure 7 presents a flowchart of the GHA.

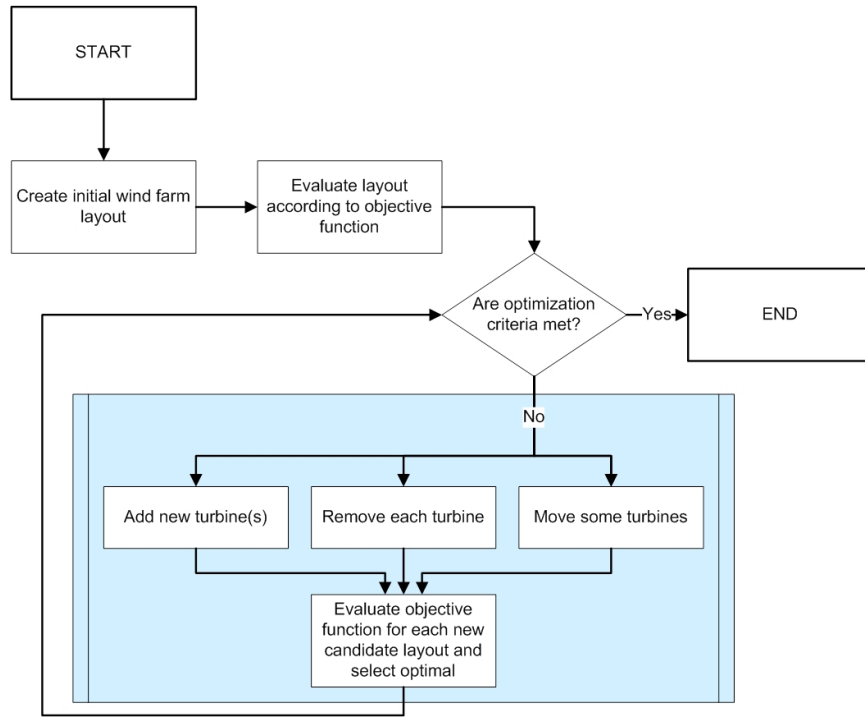


Figure 7. Flowchart of the greedy heuristic algorithm.

5. ALGORITHM TUNING

In order to allow the algorithms to perform most efficiently, it is necessary to tune the various numerical parameters. In general, the objective of tuning an algorithm is to achieve a balance between the error in the result and the required processing time. It is usually difficult to minimize both, but it is often possible to find a satisfactory compromise.

An example of a tunable parameter in the GA is the size of the population. Figure 8 shows the objective function error (left axis) and the processing time (right axis) plotted for increasing numbers of individuals in the population. The balance of small error and short processing time is achieved with populations of 200 to 300 individuals. Nine identical optimization runs were averaged to create these results.

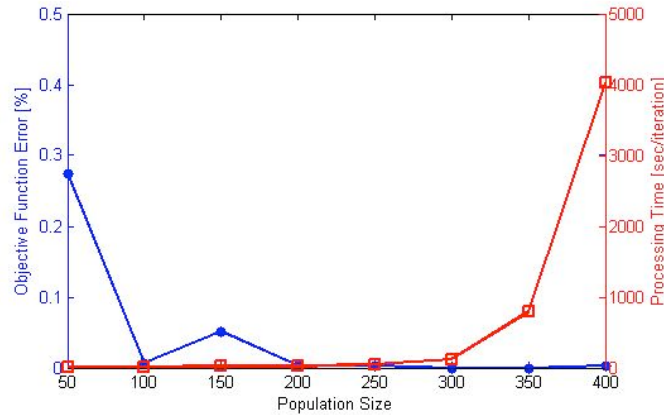


Figure 8. Tuning of population size in genetic algorithm based on the objective function error (left axis, blue line with dots) and the processing time (right axis, red line with squares).

The percentage of a population that is selected to become parents and the percentage of a new generation that is comprised of children are two other tunable parameters. When these two parameters were evaluated with respect to the standard deviation in the objective function (Figure 9) and the mean processing time (Figure 10) for nine identical optimizations, the best results were achieved with 70% of the population being selected as parents and 45-50% of the next generation being comprised of children.

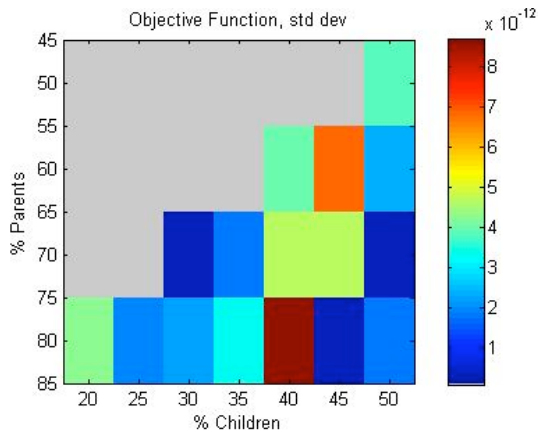


Figure 9. Validation of percentages of parents selected for crossover (y-axis) and children in new generation (x-axis) based on deviations in the objective function.

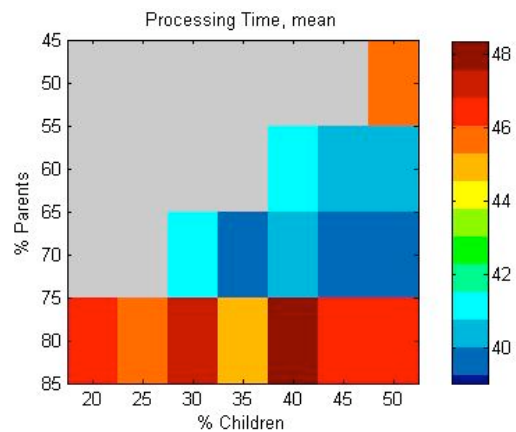


Figure 10. Validation of percentages of parents selected for crossover (y-axis) and children in new generation (x-axis) based on processing time.

Complete lists of the tunable parameters are given in Table 1 for the GA and Table 2 for the GHA.

Table 1. Tunable parameters used in GA.

Parameter	Description	Standard Value	Notes
N_{ind}	Number of individuals in a population	200-300	Value based on tuning study
$N_{termination}$	Number of consecutive, unsuccessful iterations before termination	50	
SP	Selective pressure	2	Values between 1 and 2 acceptable
$Pct_{parents}$	Percentage of individuals selected to be parents	70	Value based on tuning study
$Pct_{children}$	Percentage of next generation that is comprised of children	50	Value based on tuning study
$N_{crossover}$	Number of crossover points	1	No improvement with larger values
$Prob_{mut}$	Probability of genetic mutation in a child	30	Tuned, values greater than 40 dramatically increase processing time
N_{mut}	Number of bits to mutate	1	No improvement with larger values


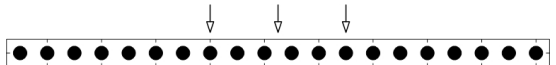
Table 2. Tunable parameters used in GHA.

Parameter	Description	Default Value	Notes
N_{add}	Number of turbines to add	1	
Pct_{move}	Percentage of turbines to move [%]	25	
$L_{move,max}$	Maximum distance to move a turbine [D]	5-10	Larger farm areas require larger move distances
$Step_{move}$	Step size for move [D]	1	
N_{iter}	Number of consecutive, unsuccessful iterations before a perturbation	10	
$N_{perturb}$	Number of consecutive, unsuccessful perturbations before termination	5	
$Pct_{perturb}$	Percentage of turbines to move during perturbation	10-25	Should guarantee movement of at least one turbine

6. VALIDATION

Several means of validating the algorithms and objective functions were used. The simplest test cases to visualize are one-dimensional cases, e.g. a farm that is 1 cell by 20 cells. Table 3 shows the anticipated results—suggested by typical designs of existing offshore farms (for Case A) and intuition (for Case B)—of two simple one-dimensional test cases. Each cell in this analysis is a one rotor diameter by one rotor diameter (1-D by 1-D) square.




Table 3. The anticipated results for simple, one-dimensional farms optimized for maximum energy production.

Case	Description	Anticipated Result
A	constant winds parallel to long axis	
B	constant winds perpendicular to long axis	

The analyses used constant wind speeds of 8 m/s from a single direction, a roughness length of 0.0002 m, and a 2 MW turbine with a rotor diameter of 80 m and a hub height of 70 m. The objective for this validation is maximum energy production. In Case A, the wind blows from left to right, in Case B, the wind is from top to bottom. The uniform spacing expected in Case A (8-D) is the approximate average spacing in the large offshore wind farms (e.g. Horns Rev has 7-D by 7-D spacings and Nysted has 10.5-D by 5.8-D spacings).

Table 4 shows the analysis results for these cases.

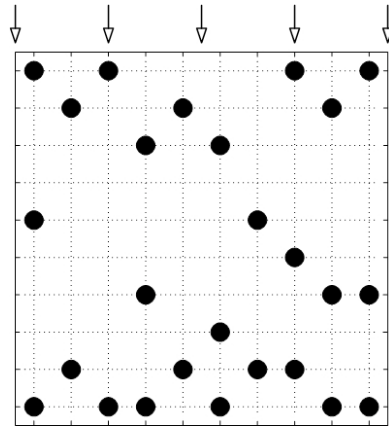
Table 4. Results from one-dimensional farm analysis.

Case	Result	Notes
A.1		$U_0 < U_{rated}$
A.2		$U_0 > U_{rated}$
B		

The most interesting result of this analysis is the effect of the wind speed on the spacing. For the case where the wind speed is less than the rated speed (case A.1, $U_0 = 8$ m/s) the turbine spacing is not uniform as was expected. This is presumably due to the significant impact of wind speed deficits on the turbine power, due to wake losses at slow to moderate wind speeds. When the wind speed is significantly greater than the rated speed, (case A.2, $U_0 = 18$ m/s), the wake effect is greatly reduced, and the turbine spacing was almost uniform at 6-7 rotor diameters, close to the expected value. This highlights the fact that the wind climate must be well understood in order to lay out a wind farm in an optimal manner.

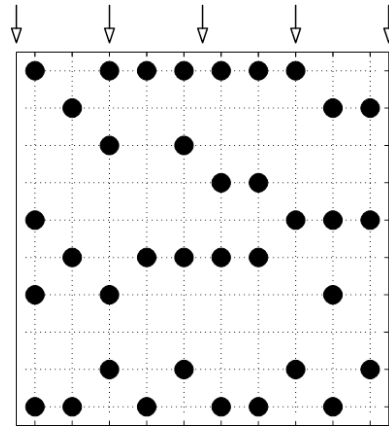
Another way to validate the algorithms themselves is by direct comparison with results in the literature. The papers of Mosetti et al. [1] and Grady et al. [3] give good descriptions of the test cases they considered. Those cases have been reproduced here, using the same configurations and objective functions where possible. The turbine used in this study had a diameter of 40 m, a rated power of 630 kW, a constant thrust coefficient of 0.88, and a hub height of 60 m. The site is characterized by a roughness length of 0.3 m, and the winds are a constant 12 m/s from the north. The results from the OWFLO optimization tool have been compared with those of the previous studies and an example of this comparison is shown in Figure 11 and Figure 12. Each cell in these figures represents a 5-D by 5-D area.

A slight discrepancy was observed between the values of total power reported here and those reported in Grady et al. [3]. The discrepancy is small and likely due to a slight difference in the wake estimation.



Number of turbines: 26
Total rated power: 12.8 MW
LPC: 4.20 ¢/kWh

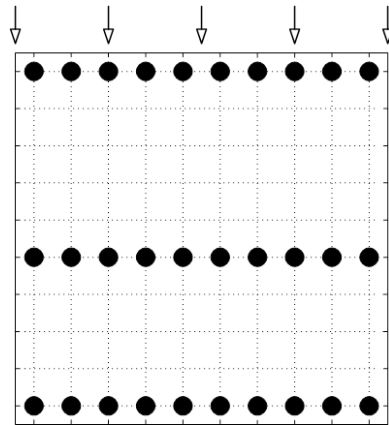
(a)



Number of turbines: 36
Total rated power: 16.8 MW
LPC: 3.98 ¢/kWh

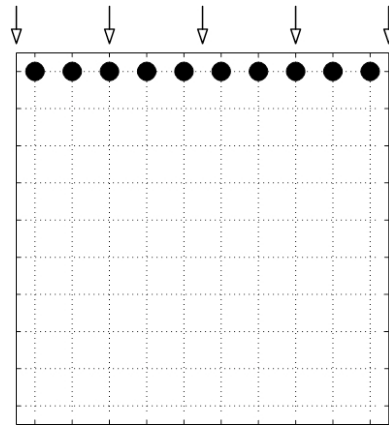
(b)

Figure 11. Comparison with results from the GA literature for a 50-D by 50-D farm with constant winds of 12 m/s from the north. (a) shows results from Mosetti et al. [1] in which the objective was a balance of minimum COE and maximum energy, using a simple COE model. (b) shows the results of the OWFLO optimization tool using the same configuration, including the same simple COE model. In both cases, the LPC values were estimated using the OWFLO optimization tool.



Number of turbines: 30
Total rated power: 14.8 MW
LPC: 4.18 ¢/kWh

(a)



Number of turbines: 10
Total rated power: 5.2 MW
LPC: 3.98 ¢/kWh

(b)

Figure 12. Comparison with results from the GA literature for a 50-D by 50-D farm with constant winds of 12 m/s from the north. (a) shows the results from Grady et al. [3] in which a simple COE estimate was minimized. (b) shows the results of the OWFLO optimization tool using a more rigorous LPC model. In both cases, the LPC values were estimated using the OWFLO optimization tool.

The comparison with the previous studies shows that, when the simple COE model used in the previous studies was employed, the results obtained here are between those of Mosetti et al. and Grady et al.—the layout did not show the neat rows of the farm of Grady et al. but was more organized than the farm of Mosetti et al. The differences here are explained by the particular implementation of the genetic algorithm. Results obtained using the more rigorous cost models were compared with the results in Grady et al. The more realistic LPC model revealed that the energy that would be produced by additional turbines did not compensate for the added cost.

7. COMPARISON OF ALGORITHMS

Four test cases have been used to compare the performance of the genetic and greedy heuristic algorithms. The turbine used in these tests has a diameter of 80 m, a rated power of 2 MW, and a hub height of 70 m. The wave climate is defined by a 50-year reduced wave height of 6.7 m and a wave length of 91 m. The soil at the site is sand. The number of turbines allowed for all four cases were constrained to be between 3 and 10. Table 5 describes the differences between each of the cases.

Table 5. Descriptions of test cases.

	Case 1	Case 2	Case 3	Case 4
General description	linear, flat, constant winds from north	flat site, constant winds from all directions	deep with small shallow areas, constant winds from all directions	constant slope, constant winds from all directions
Farm dimensions	20-D x 1-D	20-D x 20-D	20-D x 20-D	20-D x 20-D
Water depth	10 m	10 m	5 m, 99 m	4-42 m,
Wind regime	8 m/s from the north, no fetch dependence	8 m/s from 16 direction sectors, no fetch dependence	8 m/s from 16 direction sectors, no fetch dependence	8 m/s from 16 direction sectors, increases with fetch

For each case, 10 identical optimization runs have been performed using the GA, GHA, and the combination of the GA and GHA. After each run, the LPC value and processing time were recorded. The mean LPC values for each case, normalized by the mean LPC value for the GA, is shown in Figure 13, and the standard deviations of the LPC is shown in Figure 14.

From Figure 13 it can be seen that, on average, the GHA results in the solution with the highest LPC in all four test cases, while the solutions from the GA-GHA combination are either equivalent to or better than the other two algorithms. This is not unexpected, given that, in the combination, the GHA only improves upon the GA solution. The standard deviations in Figure 14 show a similar result. With the exception of Case 1, the LPC standard deviation for the GA-GHA combination is less than the other two. Lower standard deviation values indicate greater repeatability. These results suggest that the performance of the GA-GHA combination is, on average, superior to either the GA or GHA when used alone.

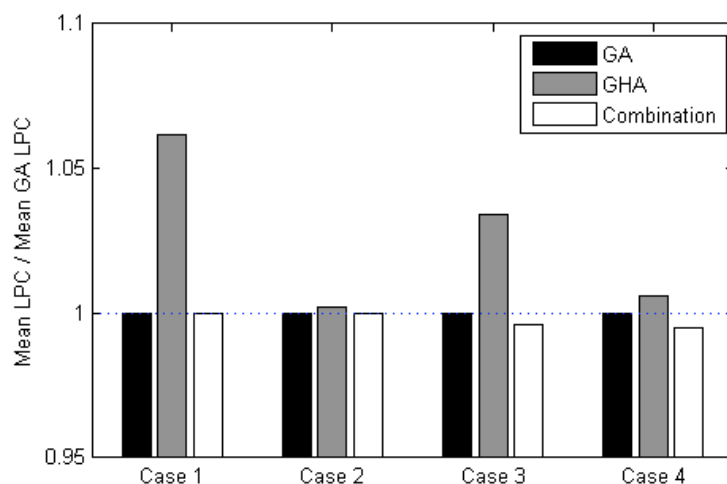


Figure 13. Mean LPC values for the GA, GHA, and GA-GHA combination, normalized by the LPC value for the GA, for the four test cases.

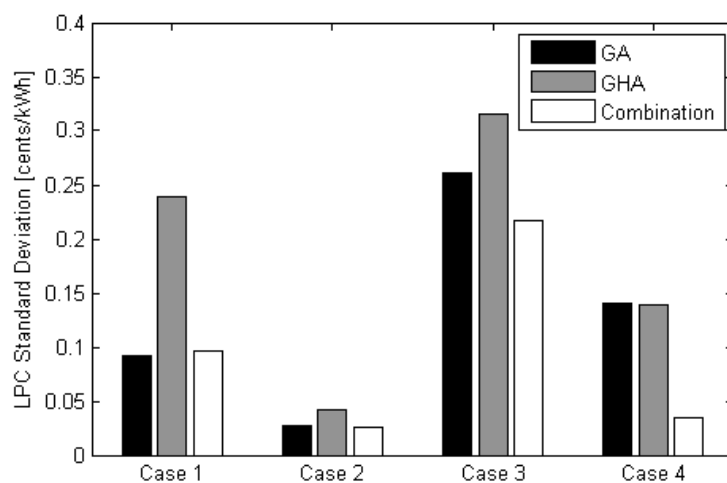


Figure 14. LPC standard deviations for the GA, GHA, and GA-GHA combination, for the four test cases.

It is also instructive to look at the processing time required by the three optimization methods for the four test cases. The mean processing time is shown in Figure 15.

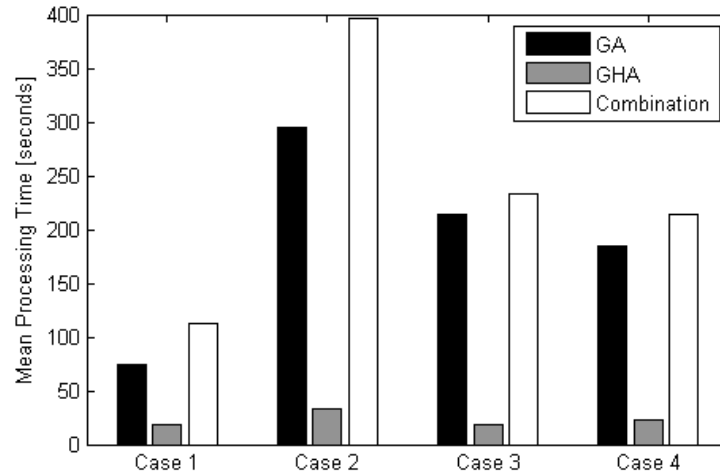


Figure 15. Mean processing time required by each optimization method for each test case.

Of the four cases, Case 2 is the least constrained—the other three have some element of physical constraint on the solution or, in Case 1, on the search space. As Figure 15 shows, the GA, GHA, and the combination of the two all took longer to arrive at a solution for Case 2. This suggests one way to reduce the processing time is to add constraints to the problem.

8. APPLICATION: HULL, MASSACHUSETTS

The town of Hull, Massachusetts, is located 13 km (8 miles) east-southeast of Boston at the southern end of the entrance to Boston Harbor (see Figure 16). The town currently owns two onshore wind turbines and has started the process of developing a small offshore wind farm within town waters. This project provides an opportunity to test the optimization algorithms and modeling software in a real-world setting. When the Hull Municipal Light Board voted to pursue the installation of an offshore wind farm, they decided that the farm should contain 4 turbines. Therefore, all of the energy and economic estimations presented here are based on the assumption that four 3 MW turbines are to be installed.

Site Description

Wind data for this analysis were obtained through a measure-correlate-predict (MCP) process [13]. Short-term data were measured at a height of 61 m above the ground, using cup anemometers on a radio tower near the coast of Hull. These data were combined with 4 years of data measured at a met tower located on Thompson Island, located in Boston Harbor, approximately 12 km (7.5 miles) west-northwest of the radio tower.

Based on the MCP analysis, the long-term mean wind speed at the radio tower is 6.7 m/s at 61 m. Using the 70 m data obtained from the New England Wind Map from AWS TrueWind [14], the wind data were translated from the radio tower to the coast by multiplying the data by the ratio of the mean speeds at the two locations (ratio = 1.01). The data were also extrapolated to the 80 m proposed hub height using the power law exponent derived from the 50 m and 70 m wind map data. This exponent, $\alpha = 0.174$, has been chosen over the exponent measured at the radio tower ($\alpha = 0.182$) in order to obtain a conservative energy estimate. Table 6 summarizes the wind

speeds at the different locations and heights. The estimated mean wind speed at 80 m at the coastal location is 7.1 m/s. These translated and extrapolated wind data, $U_{K\&L,80}$, have been used in the analyses detailed below.

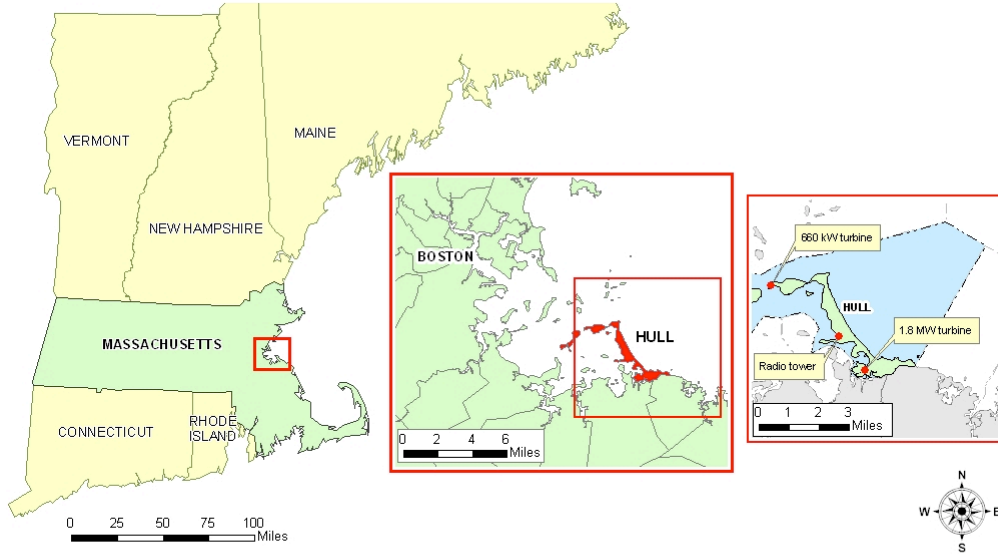


Figure 16. Map of Hull, Massachusetts, showing Hull's proximity to Boston, the offshore town boundaries, the locations of the two onshore turbines in Hull, and the location of the radio tower where wind data were measured.

Table 6. Mean wind speeds in Hull, MA.

Location	Parameter	Mean wind speed	Derivation
Radio tower, 61 m	$U_{WBZ,61}$	6.7 m/s	From MCP data
Radio tower, 80 m	$U_{WBZ,80}$	7.0 m/s	$U_{WBZ,80} = U_{WBZ,61} \cdot (80/61)^{0.174}$
Coast, 80m	$U_{K\&L,80}$	7.1 m/s	$U_{K\&L,80} = U_{WBZ,80} \cdot 1.01$

The wind at this site is characterized by the wind speed histogram and wind rose shown in Figure 17. As the wind rose shows, the large majority of the wind comes from the west and southwest. The overall Weibull c and k parameters at 80 m at the site are 8.0 m/s and 2.2, respectively.

The wind map was used again to determine a functional relationship between the wind speed and the distance from shore, x , measured in kilometers:

$$U(x) = U_{x=0} \left[-0.17 \exp\left(\frac{-x}{3.21}\right) + 1.17 \right] \quad (4)$$

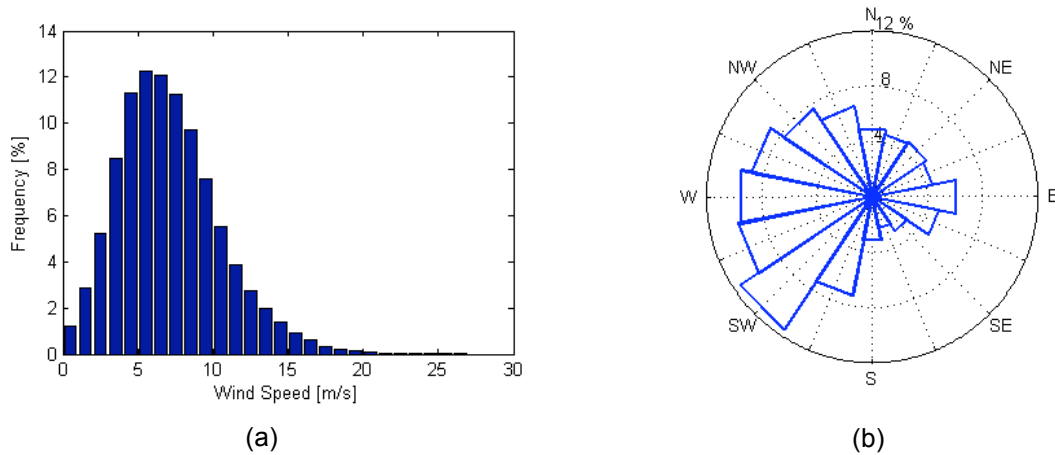


Figure 17. (a) Wind speed histogram and (b) wind rose for the 80 m data used in the analysis at Hull.

The seafloor off of Hull have been characterized as mostly boulder and cobble, with a well-defined corridor of sand [15]. Sub-bottom profiling is underway to determine the depth of the bedrock at the site.

An investigation of the wave conditions at the site is currently underway, but an initial estimate suggested that the 50-year reduced wave height (extreme wave height with a return period of 50 years) is 6.8 m and the characteristic wave period is 9.9 s [16].

Results and Discussion

Initial analysis of Hull's total offshore area showed that the greatest energy production would be achieved farthest from shore. At 7.4 km (4 nautical miles) from the coast, the estimated mean wind speed is 8.1 m/s at 80 m, compared to 7.6 m/s at 1.9 km (1 nautical mile) from the coast. For four 3 MW turbines, the estimated energy yield at 7.4 km is 36 GWh, compared with 32 GWh 1.9 km from the coast. The wind, energy, and cost estimations are summarized in Table 7. The three layouts summarized in Table 7 are depicted in Figures 18-20.

Table 7. Summary of analysis results for the proposed Hull offshore wind farm.

Location	1.9 km from shore		7.4 km from shore		Optimal solution	
Mean hub-height wind speed [m/s]	7.6		8.1		7.7	
Annual energy production [GWh]	32		36		33	
Capacity factor [%]	30		34		31	
Support structure	Monopiles	Gravity bases	Monopiles	Gravity bases	Monopiles	Gravity bases
System cost [\$ million]	22	23	26	28	21	22
Levelized production cost [¢/kWh]	7.6	7.9	7.6	8.1	7.0	7.3

The wind speed off of Hull increases with distance from shore as expected, indicating that the highest energy production is obtained at the far edge of the town boundary. The results also show, however, that the capital cost increases farther from shore. The optimization algorithms have been used to balance these two competing factors. The optimal layout shows that, for the

case of Hull, the increased cost of a farm farther from shore outweighs the benefits of higher energy production.

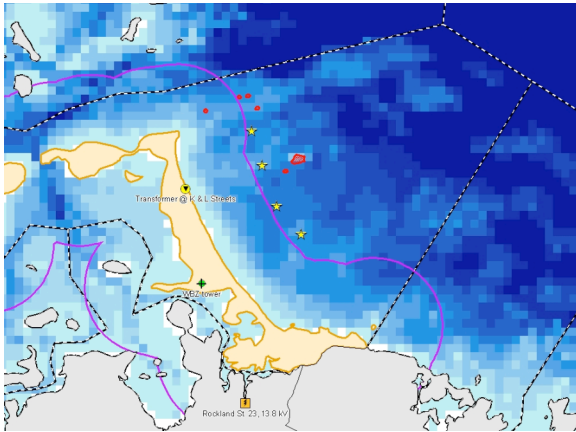


Figure 18. Map of Hull showing turbines (yellow stars) approximately 1.9 km from shore.

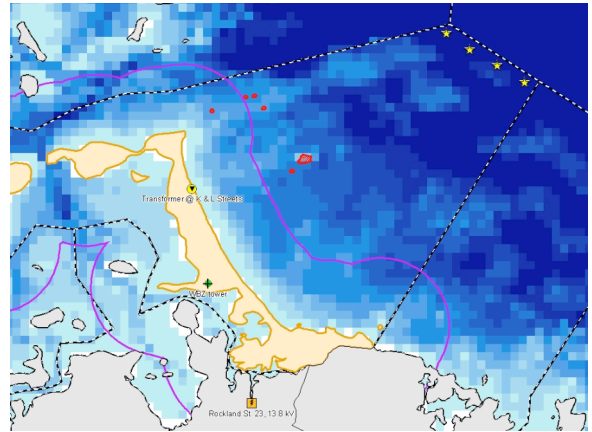


Figure 19. Map of Hull showing turbines approximately 7.4 km from shore.

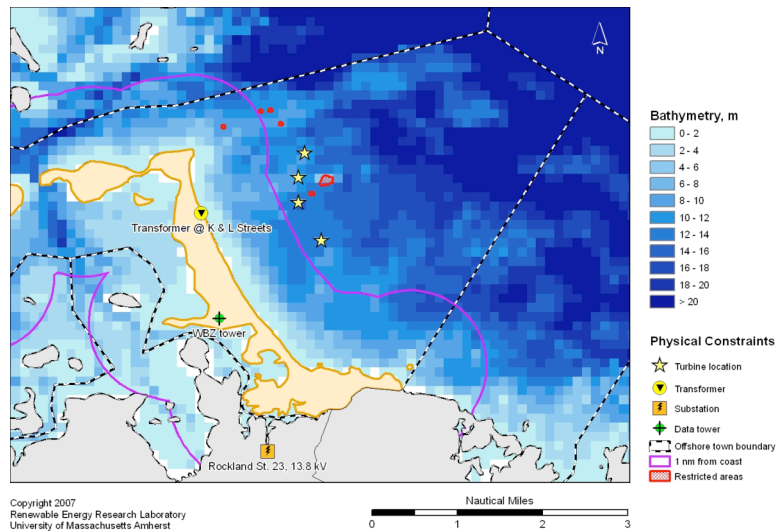


Figure 20. Map of Hull showing results of layout optimization.

The locations with the minimum LPC are close to, but not up against, the 1 nautical mile setback, and take advantage of the shallow parts of the area. The turbines are aligned perpendicularly to the prevailing southwest and west wind directions, thereby making the best use of the energy available at the site. They are separated by 6- to 10-D, which is consistent with the 5- to 10-D separation utilized in the existing farms. All of these results are reasonable and, in many cases, conform to expectation. They suggest that the cost and energy models have combined to give a reasonable representation of a wind farm off of Hull.

An important lesson was learned during the preliminary optimization of the Hull site. The total area with Hull's boundaries was too large for the optimization tool to give consistent results.

The most efficient methodology was to run the optimization tool several times over the total area to identify a smaller area in which to search further. The search was then restricted to this smaller area and the process was repeated. The solution described above was the result of one such iteration.

9. CONCLUSIONS AND FUTURE WORK

Five optimization algorithms were considered for utilization in offshore wind farm micrositeing. Four of the algorithms appeared to be well suited to this purpose. Of these, the greedy heuristic and genetic algorithms have been implemented, tuned, and validated. From the analysis of these algorithms, the following conclusions can be drawn:

- ▶ The combination of an optimization algorithm, such as one of those described in this paper, with an appropriately rigorous objective function, has the ability to optimize offshore wind farm layouts such that the cost of energy is minimized.
- ▶ Simple, one-dimensional test cases have been analyzed and the results have been compared with expected values. When the wind regime consisted of constant wind speeds greater than the rated speed of the turbine, uniform 6-7 diameter turbine spacings were observed. For wind speeds less than the rated speed, the turbine spacing were greater at the leading edge of the farm, and decreased towards the trailing edge, due to wake effects.
- ▶ Comparisons have been made with optimized farm layouts from previous studies. When the methodologies of the previous studies were recreated, the resulting farm layout was comparable with the earlier work. When the more realistic LPC model was utilized, a lower LPC was achieved with a layout consisting of fewer turbines than had previously been obtained. This suggests that, for this case, the energy from the additional turbines did not outweigh the added cost. The improvement in the LPC over the previous studies reinforces the conclusion that the methodology described here is comparable, if not an improvement, over previous methods.
- ▶ The combination of the two optimization algorithms, genetic and greedy heuristic, in series operation—the genetic algorithm provides the initial solution for the greedy heuristic algorithm—yields better, more consistent results than either algorithm by itself. Between the two algorithms, the GA results in better solutions, but requires significantly greater processing time.
- ▶ The choice of the locations for the turbines in the Hull project was dominated by the higher cost of the support structures and electrical interconnection in deeper water farther from shore, which outweighed the larger energy potential farther from shore. In general, the optimization yielded reasonable results, including:
 - the turbines grouped in a line perpendicular to the prevailing direction, suggesting that wake effects played a significant part in the choice of layout,
 - an inter-turbine separation of 6- to 10-D—a value that is within the typical range for offshore farms—indicating that the balance between reduced wake effects and increased collection cable costs has been met,

- a preference for shallow locations, indicating that the cost of the support structure was important in the optimization process.

Applying the optimization methods that have been discussed here to the real-world project in Hull has illuminated their strengths and weaknesses. Overall, the optimization had a greater likelihood of finding good solutions within small search areas. The total area within Hull's boundaries proved too big an area for the optimization to produce good results consistently. This problem was resolved by running the optimization several times to identify a smaller target area, then confining the search to that smaller area and rerunning the optimization.

This project will benefit from additional refinement of the algorithms, perhaps with the addition of alternate strategies within the genetic algorithm. For example, the only selection scheme tested here was the roulette wheel scheme. Other schemes exist and it may be instructive to compare the results of some of these alternatives. In addition, the pattern search and simulated annealing algorithms show great potential for use in wind farm micrositeing and implementing them would be a good extension of this work.

The work presented in this paper is part an on-going project focused on the design of an offshore wind farm for the town of Hull, Massachusetts. As the Hull project moves forward, the optimization and analysis tools will be used to refine the layout of the farm.

10. ACKNOWLEDGEMENTS

The authors would like to thank the Massachusetts Technology Collaborative for funding the present phase of the Hull project, and the Hull Municipal Light Plant for providing a real-world test case.

11. REFERENCES

- [1] Mosetti, G., Poloni, C. and Diviacco, B. (1994). "Optimization of Wind Turbine Positioning in Large Windfarms by Means of a Genetic Algorithm", *Journal of Wind Engineering and Industrial Aerodynamics*, 51 (1), 105-116.
- [2] Ozturk, U.A. and Norman, B.A. (2004). "Heuristic Methods for Wind Energy Conversion System Positioning", *Electric Power Systems Research*, 70 (3), 179-185.
- [3] Grady, S.A., Hussaini, M.Y. and Abdullah, M.M. (2005). "Placement of Wind Turbines Using Genetic Algorithms", *Renewable Energy*, 30 (2), 259-270.
- [4] Moreno, S.V. "Wind Farm Optimisation", *CREST, Loughborough University*, Loughborough, UK, 2001.
- [5] Elkinton, C.N., Manwell, J.F. and McGowan, J.G. (2006). "Offshore Wind Farm Layout Optimization (OWFLO) Project: Preliminary Results", *Proc. 44th AIAA Aerospace Science Meeting and Exhibit*, AIAA, Reno, NV.
- [6] Sait, S.M. and Youssef, H., *Iterative Computer Algorithms with Applications in Engineering*, IEEE Computer Society, Los Alamiton, CA, 1999.
- [7] Cagan, J., Kenji, S. and Yin, S. (2002). "A Survey of Computational Approaches to Three-Dimensional Layout Problems", *Computer-Aided Design*, 34, 597-611.

- [8] Elkinton, C.N., Manwell, J.F. and McGowan, J.G. (2006). "Modeling the Trade-Offs in Offshore Wind Energy Micrositing", *Proc. WINDPOWER 2006 Conference and Exhibition*, American Wind Energy Association, Pittsburgh, PA.
- [9] Lackner, M.A. and Elkinton, C.N. (2007). "An Analytical Framework for Offshore Wind Farm Layout Optimization", *Wind Engineering*, 31 (1), 17-31.
- [10] Jensen, N.O. "A Note on Wind Generator Interaction", Riso-M-2411, *Riso National Laboratory*, Roskilde, DK, 1983.
- [11] Katic, I., Hojstrup, J. and Jensen, N.O. (1986). "A Simple Model for Cluster Efficiency", *Proc. European Wind Energy Association Conference and Exhibition*, A. Raguzzi, Rome, Italy, pp. 407-410.
- [12] Pohlheim, H. "GEATbx Introduction - Evolutionary Algorithms: Overview, Methods and Operators, Version 3.7", 2005. Available: www.geatbx.com.
- [13] Rogers, A.L., Rogers, J.W. and Manwell, J.F. (2006). "Uncertainties in Results of Measure-Correlate-Predict Analyses", *Proc. European Wind Energy Conference and Exhibition*, Athens, GR.
- [14] "New England Wind Map", AWS TrueWind, Accessed 2007, URL: truewind.teamcamelot.com/ne/.
- [15] Ackerman, S.D., Butman, B., Barnhardt, W.A., Danforth, W.W. and Crocker, J.M. "High-Resolution Geologic Mapping of the Inner Continental Shelf: Boston Harbor and Approaches, Massachusetts", Open-File Report 2006-1008, *U.S. Geological Survey*, Reston, VA, 2006. Available: pubs.usgs.gov/of/2006/1008/.
- [16] Letson, F. "50 Year Extreme Waves", Internal Report, *RERL*, Amherst, MA, 2007.