

Aesthetic local search of wind farm layouts

Michael Mayo ^{Corresp., 1}, Maisa Daoud ¹

¹ Department of Computer Science, University of Waikato, Hamilton, New Zealand

Corresponding Author: Michael Mayo
Email address: michael.mayo@waikato.ac.nz

The visual impact of wind farm layouts has seen little consideration in the literature on the wind farm layout optimisation problem to date. Most existing algorithms focus on optimising layouts for power or cost of energy alone. In this paper, we consider the geometry of wind farm layouts and whether it is possible to bi-optimize a layout for both energy efficiency and the degree of visual impact that the layout exhibits. We develop a novel optimisation approach for solving the problem, with our approach towards measuring mathematically the degree of visual impact drawing inspiration from the field of architecture. To evaluate our ideas, we demonstrate them on three benchmark problems for the wind farm layout optimisation problem in conjunction with two recently published stochastic local search algorithms. Optimal patterned layouts are shown to be very close in terms of energy efficiency to optimal non-patterned layouts.

1 Aesthetic Local Search of Wind Farm 2 Layouts

3 Michael Mayo¹ and Maisa Daoud²

4 ^{1,2}Dept. of Computer Science, University of Waikato, Hamilton, New Zealand

5 Corresponding author:

6 Michael Mayo¹

7 Email address: michael.mayo@waikato.ac.nz; Tel.: +64-7-8384403

8 ABSTRACT

9 The visual impact of wind farm layouts has seen little consideration in the literature on the wind farm
10 layout optimisation problem to date. Most existing algorithms focus on optimising layouts for power or cost
11 of energy alone. In this paper, we consider the geometry of wind farm layouts and whether it is possible
12 to bi-optimize a layout for both energy efficiency and the degree of visual impact that the layout exhibits.
13 We develop a novel optimisation approach for solving the problem, with our approach towards measuring
14 mathematically the degree of visual impact drawing inspiration from the field of architecture. To evaluate
15 our ideas, we demonstrate them on three benchmark problems for the wind farm layout optimisation
16 problem in conjunction with two recently published stochastic local search algorithms. Optimal patterned
17 layouts are shown to be very close in terms of energy efficiency to optimal non-patterned layouts.

18 1 INTRODUCTION

19 Worldwide, renewable energy production via wind is becoming increasingly important. In particular, it
20 has been forecast that by 2030 approximately 18% of the planet's total energy production will be sourced
21 from wind farms (Global Wind Energy Council, 2014). The rapid growth in wind energy production is
22 well-illustrated by examining some of the current and planned wind farms installations around the world:
23 the London Array ¹, for example, generates 630MW of power (enough for 490,000 households); similarly,
24 the ongoing Gansu project in China (Watts, 2012) is planned to generate 20GW by 2020 (equating to
25 power for approximately 15 million households). Concerns about the environmental impact of wind
26 energy have also been increasing alongside growth in its production. In particular, it has been noted for
27 some time that wind turbines have considerable impact on local wildlife populations such as birds, bats,
28 and for offshore farms, various marine wildlife (Dai et al., 2015). So serious is the problem that there have
29 been recent calls for entirely new research programs to be developed solely to study the effects of wind
30 generation on wildlife and how to mitigate them (Piorkowski et al., 2012). The human impact of wind
31 energy production is also considerable: wind farms tend to generate significant noise, and they have a
32 major visual impact on the landscape due to the size of the turbines. On clear days turbines can be seen up
33 to 30 kilometers away depending on the turbine height and terrain conditions (Dai et al., 2015). Moreover,
34 the human impact can range in severity from a "mere" belief (strong or otherwise) that the wind turbines
35 detract from the visual value of the landscape, to shadow flicker, a phenomenon caused by the interaction
36 of a wind turbine's blades with direct sunlight. Such a phenomenon is known to cause severe headaches
37 when nearby residents are exposed to it for a long period (Tabassum-Abbasi et al., 2014).

38 This paper concerns itself with one aspect of the environmental impact of wind farm design, specifi-
39 cally, the arrangement of wind turbines into geometrical patterns, and the relationship of these geometric
40 patterns with the overall energy efficiency of the farm.

41 To partially mitigate the negative visual impacts of wind farms, it has been noted that farms with
42 a regular layout of turbines tend to be perceived as blending into the visual landscape in a better way
43 than farms with an irregular layout (Dai et al., 2015). More precisely, research by (Tsoutsos et al., 2006)
44 discusses the aesthetic principles of wind farm layout design: farms have a higher aesthetic appeal either

¹<http://www.londonarray.com/wp-content/uploads/London-Array-Brochure.pdf>

45 when turbines are arranged clearly into rows, or when they are arranged into uniform density small
46 clusters of 2-8 turbines which are separated by obvious landmarks. The latter arrangement is particularly
47 preferred when the wind farm must be integrated with agriculture.

48 One significant issue arising when considering the layout of turbines on a wind farm is the loss of
49 wind energy due to the interaction between nearby turbines, a phenomenon known as the “wake effect”.
50 Therefore, all configurations of turbines in a wind farm are not equal, and often a computationally-
51 expensive simulation (or an approximation thereof) is required to assess the wake effect so that it can be
52 mitigated as much as possible. Sometimes other objectives may be also be considered (e.g. construction
53 costs) but the commonality amongst many papers in the literature is a focus on wake effect minimisation.
54 Examples include the seminal work in the field by Mosetti et al. Mosetti et al. (1994) as well more recent
55 works such as that by Wagner et al. (2013), Rodrigues et al. (2013), Guirguis et al. (2016), Mayo and
56 Zhen (2016); and Mayo and Daoud (2016).

57 In general there is a trade-off between the geometric constraints required to minimise the negative
58 visual impact of a wind farm, and the energy output of the farm itself. A simple example would be
59 the arrangement of turbines equidistantly around the perimeter of a circle: although a circle is visually
60 interesting shape, there will always be a sizeable portion of turbines (on opposite sides of the circle) that
61 lie in each other’s wakes regardless of the predominant wind direction.

62 Beyond forcing turbines to be arranged into simple geometric shapes such as circles and grids, it is
63 not immediately clear how to define “visually appealing” arrangements in a more general way. Most of
64 the past and current works on wind farm layout optimisation (WFLO), therefore, tend to ignore geometric
65 appearances and consequently highly optimised layouts may taken on a “random scattering” appearance.
66 This is shown in some of the figures later in this paper.

67 Very few authors have considered the visual aspects of layouts as part of the optimisation process.
68 Two works exist as far as the authors are aware: an excellent paper by Neubert et al. (2010) in which
69 turbines are constrained to a skewed grid, and the orientation and skew of the grid is optimised; and an
70 approach by Al-Yahyai et al. (2015), in which turbines are also assigned positions on a grid but in this
71 case only the grid’s orientation is optimised.

72 In both cases, the geometric constraints are extreme and therefore the optimisation problem can be
73 solved by varying only two or one variables respectively. Despite the limitations of these approaches,
74 Neubert et al. show that the geometrically constrained layouts are almost (within a few percentage points)
75 as efficient as completely unconstrained layouts that are optimised purely for energy efficiency.

76 In this paper, we take a different tack. Rather than strongly constraining layouts in order to force a
77 geometric pattern on them (as the previous authors have done), we instead add a pattern-based metric to
78 the optimiser that assesses the quality of the pattern that the turbines form. Thus, each individual turbine’s
79 position on the layout is still a degree of freedom, but at the same time, a poorly-arranged layout with the
80 same energy efficiency as a well-arranged layout will score an overall worse objective value. Thus the
81 optimiser should focus its search towards layouts with either minimal visual impact (or, alternatively, a
82 strong aesthetic appeal).

83 To evaluate our novel approach, we use two stochastic local search algorithms for the wind farm
84 layout optimisation problem that have recently appeared in the literature. Both approaches are combined
85 with our novel objective function, thus producing two new approaches. The first existing approach we
86 utilise is called the Turbine Displacement Algorithm (TDA) (Wagner et al., 2013). The second approach
87 is our own recently published approach known as BlockCopy (Mayo and Zhen, 2016; Mayo et al., 2016).

88 In order to assess the geometric/pattern quality of layouts, we have utilised a pattern measure originally
89 proposed by Salingaros (1997) and Klinger and Salingaros (2000). This metric has its origin in the
90 architectural evaluation of building facades, but has been generalised for the evaluation of any kind of
91 symbolic pattern. Therefore it is ideally suited for our purposes.

92 The pattern metric we use as well as all other relevant technical details are described in the next
93 section. Following that, we describe how we modify the objective function of the TDA and BlockCopy
94 algorithms to optimise for geometric qualities. Sections 4 and 5 describe a comprehensive evaluation of
95 both algorithms that was performed, and finally Section 6 concludes the paper.

96 We acknowledge at this point that the work undertaken here is largely focussed on the aesthetics
97 of two dimensional layouts. Thus, it would be applicable to situations where the wind farm is located
98 off-shore or on a plain, but not in a situation where the farm is located on a three dimensional terrain
99 (e.g. along a ridge). Furthermore, adjustments would also have to be made to the proposed method if

100 the importance of aesthetics varies across the layout. For example, an area of the farm close to a tourist
 101 attraction is likely to have much more visual impact than the part of the farm furthest from the attraction.
 102 We address these concerns in the conclusion.

103 2 TECHNICAL BACKGROUND

104 2.1 Jensen Wake Model

105 The Jensen far wake model, originally proposed in the mid-1980s (Jensen, 1983; Katic et al., 1986), is
 106 the approach we use in this research to assess the wake interactions between turbines in a wind farm
 107 layout. Although dated, the Jensen is still used widely in the community. To illustrate, Samorani (2013)
 108 describes it precisely in a recent 2013 introductory survey to the Wind Farm Layout Optimisation (WFLO)
 109 problem, and in a 2016 comparison of three kinematic far wake models and two field-based far wake
 110 models, Shakoor et al. (2016) concluded that "...Jensen's far wake model is a good choice to solve the
 111 wind farm layout optimisation problem due to its simplicity and relatively high degree of accuracy." We
 112 therefore adopt the Jensen model for our initial investigations reported here, while acknowledging that
 113 more sophisticated models do exist that we will explore in future work.

114 In this section we therefore describe briefly the Jensen far wake model. As we are using, more or
 115 less, the same notation as Samorani (2013), the interested reader is referred to that publication for more
 116 specific details.

117 The first element required in wind farm modeling is a power curve, which describes the relationship
 118 between incoming wind speed and the power generated by a single wind turbine. This is generally
 119 dependent on the type of wind turbine being modelled and therefore will vary depending on manufacturer
 120 and model. In general, however, the relationship can be modelled as a cubic function from wind speed
 121 to power between two bounding wind speeds: (i) the cut in speed, which is the wind speed at which the
 122 turbine begins generating power; and (ii) the nominal speed, which is the wind speed at which maximum
 123 power production is reached. A final element of the power curve is the cut out speed. This is the maximum
 124 allowable wind speed that the turbine can tolerate before shutting down to avoid damage.

125 Due to the variability and manufacturer-dependence of different wind turbine models, we adopt in this
 126 paper the power curve used by Mosetti et al. (1994) and also described by Samorani (2013):

$$127 \text{power}(u) = \begin{cases} 0\text{kw} & \text{where } u < 2\text{m/s} \\ 0.3u^3\text{kw} & \text{where } 2\text{m/s} \leq u < 12.8\text{m/s} \\ 629.1\text{kw} & \text{where } 12.8\text{m/s} \leq u < 18\text{m/s} \\ 0\text{kw} & \text{where } u > 18\text{m/s} \end{cases} \quad (1)$$

128 In this power curve, wind speed u is measured in metres per second (m/s) and power in kilowatts (kw). We
 129 acknowledge here that this turbine model is somewhat dated (for example, modern turbines may produce
 130 8-10MW of power), but it is a model that is frequently used in the literature and therefore we adopt it in
 131 this paper for the purposes of reproducibility ease.

132 The next part of the Jensen far wake model is the modelling of the velocity deficit, i.e. the reduction
 133 in wind speed as wind passes through the blades of a turbine. This is best introduced schematically, and
 134 here we reproduce a diagram from Samorani (2013) in Figure 1.

135 The key elements of Figure 1 are an illustration of the initial wind speed u_0 ; the reduced wind speed
 136 u_j at a distance x metres from the wind turbine; and the notion that the wake spreads out with linearly
 137 increasing radius as it gets further away from the turbine.

In fact, the radius of the spreading wake is modelled by the following equation:

$$138 r_1 = \alpha x + r_r \quad (2)$$

139 where r_r is the turbine's radius and r_1 is the radius of the wake. This equation shows that the rate of
 140 spreading is determined by a constant α which in turn depends on two further factors: first, the height of
 the turbine z ; and second, the surface roughness, z_0 . The function to calculate α is:

$$\alpha = \frac{0.5}{\ln \frac{z}{z_0}} \quad (3)$$

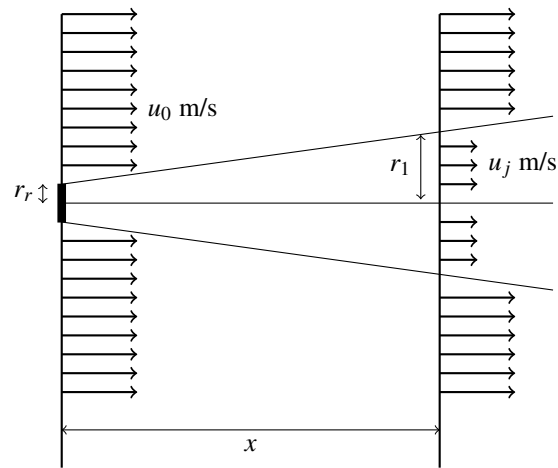


Figure 1. Depiction of the wake effect (reproduced from Samorani (2013)).

141 Once the size of the spreading wake can be determined as a function of distance x , the degree of wind
 142 speed velocity deficit needs to be next computed. The relationship between wind speed and velocity
 143 deficit is modelled as

$$u_j = u_0(1 - vd_{ij}) \quad (4)$$

144 where u_j is the wind speed at a position j which is inside the wake of a turbine at position i . The term vd_{ij}
 145 represents the velocity deficit between positions i and j due to the turbine, and is calculated (according to
 146 Samorani (2013)) thus:

$$vd_{ij} = \frac{2a}{1 + \alpha \left(\frac{x_{ij}}{r_d}\right)^2} \quad (5)$$

147 where x_{ij} is the distance between the two points and a , the “axial induction factor” is defined as

$$a = 0.5(1 - \sqrt{1 - C_T}) \quad (6)$$

148 while r_d , the “downstream wake radius” is calculated as

$$r_d = r_r \left(\sqrt{\frac{1-a}{1-2a}} \right) \quad (7)$$

149 This value r_d is used as the input to the velocity deficit calculation (Equation 5 above) as per the model.

150 A key corollary of these equations is that while power increases with the cubic of wind speed, velocity
 151 deficit decreases at a rate proportional to the square of the distance from a turbine. Therefore it follows
 152 that simply finding a “windier” site should increase power production regardless of whether wake effects
 153 are minimised or not.

154 Next, the model also accounts for the fact that a turbine may be in the wake of not one, but many other
 155 wind turbines, at the same time. In this case, the velocity deficit calculation is more complex because the
 156 different velocity deficits must all be aggregated and subtracted from the incoming wind speed together.
 157 This is achieved in the Jensen model by calculating the square root of the sum of the squared velocity
 158 deficits:

$$v_{def}^s(j) = \sqrt{\sum_{i \in W^s(j)} vd_{ij}^2} \quad (8)$$

159 where $v_{def}^s(j)$ is the total velocity deficit and $W^s(j)$ is the set of turbines affecting the turbine at position j .
 160 Turbines are represented as points for the purposes of this set membership calculation and therefore they
 161 are either completely inside or outside the wake; we acknowledge that other wake models allow partial
 162 wake overlaps of the rotors, but we have followed the Jensen model rigorously in this work which treats
 163 turbines as points. As pointed out earlier by Shakoor et al. (2016), this should be sufficiently accurate.

164 The index s denotes the *wind scenario* which specifies both the wind direction and the initial wind
 165 speed: both of these factors determine the wakes in which a turbine lies and therefore what the total
 166 velocity deficit will be.

167 Two things must be noted about the calculation of the set $W^s(j)$. Firstly, computing the wakes within
 168 which a turbine j lies for any arbitrary wind direction requires some non-trivial 2D geometric calculations.
 169 This is because the wind may blow in any direction, and therefore wakes may expand in any direction.
 170 However, this calculation is readily computable with some standard trigonometry.

171 Secondly, and much more significantly, a routine to calculate the velocity deficit for every turbine
 172 in a layout for a single wind direction is a function with quadratic complexity: this is because every
 173 turbine j in the layout must be compared to every other turbine in order to determine $W^s(j)$. Thus, layout
 174 evaluators can face scalability issues as the size of the layout increases.

175 Finally, to complete our presentation of the Jensen model, there are several constants required.
 176 Specifically, these are r_r , the turbine radius, which we set to 20 meters; z , the hub height, which is
 177 initialised to 60 meters; z_0 , the surface roughness constant, which is 0.3 meters; and C_T , which is 0.88. A
 178 minimum allowable distance between turbines must also be specified, because the Jensen model is not
 179 accurate at close distances. We set this constant to three times the diameter of the rotors, namely 120
 180 meters. These constants are all as-used by Samorani (2013). We expect that other situations will require
 181 different unique values for the above parameters since they depend on the model of wind turbine being
 182 used as well as characteristics of the site. However, these values are good defaults for the purposes of
 183 reproducibility, and we don't expect the behavior of the approaches that we present later to be significantly
 184 dependent on particular choices of values.

185 2.2 Objective Function

186 Once the Jensen model is completely specified, the next step is to precisely specify the objective function
 187 for the WFLO problem. There are generally many different ways of doing this depending on what
 188 optimisation is required. One approach is to simply calculate the total expected power generated by a
 189 wind farm, which must be maximised (e.g. Song et al. (2016)). Another approach is to calculate the
 190 expected cost of energy: take the total expected power, convert it into units of currency that would be
 191 obtained if the power were sold at market, and divide that revenue by the cost of building and maintaining
 192 the wind farm (e.g. Mayo and Daoud (2016)). This is an objective that must be minimised.

193 For this preliminary assessment of our new approach, we use a simple objective function that divides
 194 the total expected power generated by the farm with wake interference by the total *hypothetical* expected
 195 power that would be generated by the farm without wakes. Clearly, this ratio should result in a value
 196 between 0 and 1, with higher values being more desirable.

197 If \mathbf{I} is a wind farm layout, the objective function therefore is:

$$198 F(\mathbf{I}) = \sum_{s \in S} r_s \frac{\sum_{j \in \mathbf{I}} \text{power}(u_s(1 - v_{def}^s(j)))}{\sum_{j \in \mathbf{I}} \text{power}(u_s)} \quad (9)$$

199 where j is a turbine's position in the layout, and $v_{def}^s(j)$ is the total velocity deficit at j . S is a set of wind
 200 scenarios, r_s is the probability of scenario $s \in S$ and u_s is the wind speed under scenario s . It should be
 evident that $\sum_{s \in S} r_s = 1.0$ in order to compute proper expected power values.

201 2.3 Turbine Displacement Algorithm

202 The Turbine Displacement Algorithm (TDA) is a highly effective stochastic local search algorithm for the
 203 wind farm layout optimisation problem first introduced by Wagner et al. (2013). The algorithm shifts
 204 a single turbine at a time, and then re-evaluates the layout to determine if the turbine move should be
 205 accepted or not.

206 Although initially designed to be used in conjunction with a specific wake model in order to reduce
 207 the computational complexity of layout evaluation, the algorithm is in fact competitive with many other

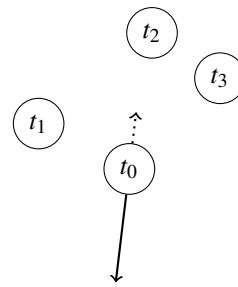


Figure 2. Illustration of the TDA operator. In this example, $K = 3$ and the displacement vector for t_0 (and its potential inverted displacement vector after rescaling to prevent collisions) is shown.

208 approaches that use different wake models. A recent evaluation by Wilson et al. (2014) showed that TDA
 209 outperformed several other metaheuristic algorithms including genetic algorithms and particle swarm
 210 optimisation. We therefore use TDA as one of the algorithms in our evaluation.

211 The basic behaviour of one iteration of TDA is shown in Figure 2. Essentially, a neighbourhood size
 212 K must be specified initially by the user. A random turbine is then picked, and it is moved either away
 213 from the K neighbours or, with reduced probability, towards the K neighbours. The direction that the
 214 turbine moves is called its displacement vector.

215 In the original paper on TDA (Wagner et al., 2013) a study on the best value of K for different layout
 216 sizes was performed. It was found that for very small layout, a small K (e.g. $K = 1$) lead to an efficiency
 217 gain of just over 1%; however, as the layout size increased, the difference in efficiency caused by varying
 218 K approached a negligible value. $K = 8$ was the highest value tested in that paper.

219 Each displacement vector has a specific size, and one feature of the algorithm is that the size of the
 220 displacement vectors is not constant. Instead, it varies on a per-turbine basis: if a turbine's moves are
 221 frequently accepted (i.e. lead to improvements in objective value) then the size of the displacement vectors
 222 is gradually increased; conversely if a turbine's moves are not accepted, the size decreases.

223 A complete specification of TDA can be found in Wagner et al. (2013), and it suffices to state the
 224 parameters that we used: the best neighbourhood size we found in initial experiments was $K = 8$; the
 225 initial displacement vector size was set to 120 meters; the scaling factor for reducing displacement vector
 226 sizes was 0.9, and conversely the factor for increasing sizes was $\frac{1}{0.9}$; and finally the amount of "distance
 227 noise" added to the displacement vectors was set to 40 meters. All other parameters and properties of
 228 TDA are the same as reported in Wagner et al. (2013).

229 2.4 BlockCopy Local Search Algorithm

230 In comparison to TDA which moves one turbine at a time, the BlockCopy local search algorithm described
 231 in Mayo and Zhen (2016) and further extended in Mayo et al. (2016) operates by copying entire groups
 232 of turbines at a time. This operation is illustrated in Figure 3. The basic idea to replace one random
 233 square region (a "block") of a wind farm layout with copy of another square region. Any turbines in the
 234 destination region before the copy occurs are deleted. After the copy, if the total number of turbines in the
 235 layout has either increased or decreased (because of differences in the number of turbines per block), then
 236 turbines are either randomly added or randomly purged in order to keep the total number of turbines in
 237 the layout fixed.

238 An advantage of this approach is that the relative configuration of turbines is maintained whenever
 239 a block is copied, and therefore if a particularly good configuration of turbines (for the conditions) is
 240 present in the layout, then this configuration will quickly replicate itself across the layout via successive
 241 BlockCopy operations.

242 In the initial evaluation of the algorithm that was recently published (Mayo and Zhen, 2016), the
 243 algorithm was shown to outperform TDA on a set of benchmark problems using a cost-based objective
 244 function and a different far wake model than Jensen. However, the number of iterations of each algorithm
 245 in that paper was only 2,000. In this paper, we give both algorithms ten times as many iterations, which
 246 should make TDA more competitive.

247 BlockCopy is the second layout optimiser used in this paper, and the single main parameter for the
 248 algorithm (i.e. the block size) is set in this research to 250 meters \times 250 meters. The block sizes and

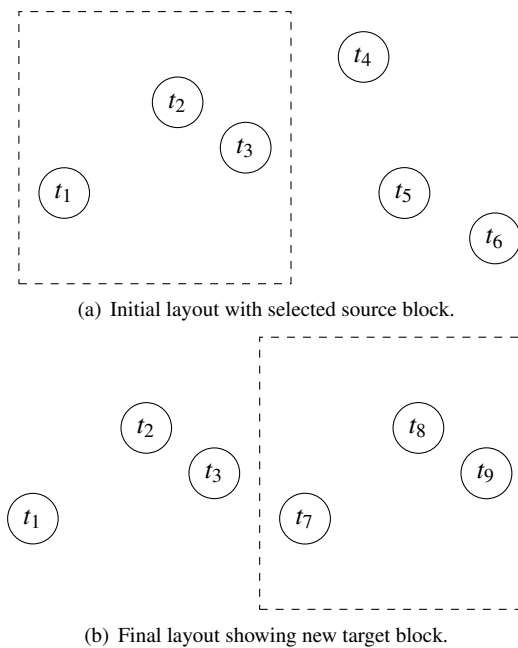


Figure 3. Illustration of the BlockCopy operator. In this example, the left block of a small layout is duplicated to the right hand side of the layout.

249 positions are fixed, non-overlapping, and completely exhaustive across the layout.

250 2.5 Harmony Pattern Metric

251 In order to assess the visual elegance of wind farm layouts, we selected an aesthetic pattern metric called
 252 harmony first proposed by Salingaros (1997) as a method of assessing the aesthetics of building designs.
 253 Subsequently the approach was generalised so that it could be applied to any type of pattern, as long as
 254 the pattern could be represented by an array of discrete symbols (Klinger and Salingaros, 2000).

255 One motivation for selecting this metric over some of the more recent methods from the field of
 256 computational aesthetics (den Heijer and Eiben, 2010; Galanter, 2012) is that harmony does not expect
 257 patterns to be derived from images. Other metrics frequently assume they are being used for image
 258 assessment and therefore rely on the calculation of quantities such as compression ratios, or statistics
 259 related to image properties such as colour, which make them difficult to apply to non-image patterns.

260 The harmony metric can be described as follows in the remainder of this section. We use a slightly
 261 more succinct notation than that presented in the original paper, mainly for improved clarity.

262 Firstly, the pattern must be represented as a rectangular (preferably square) array of symbols, where
 263 each symbol corresponds to one basic constituent of the pattern. For example, in architecture, one element
 264 might correspond to a curved corner and another to a window. In our approach, we make use of only two
 265 symbols (1 or 0) which correspond to the presence or absence of a turbine in a particular small region in
 266 the layout. Examples of some small symbol arrays, two of which are from Klinger and Salingaros (2000)
 267 are shown in Figure 4. Note that in the figure, the entries $\{0,1,2,3\}$ denote symbols and H denotes the
 268 harmony metric which is defined next.

269 Once the symbol array is available, the harmony metric is computed by first of all evaluating a
 270 number of functions on the symbol array. Each function concerns one particular class of symmetry, either
 271 reflective, rotational, or in relation to another pattern. The functions (nine of them) are listed in Table 1,
 272 and each returns either 1 or 0 depending on whether the pattern has the particular class of symmetry that
 273 the function is concerned with.

274 Before showing how the overall harmony of a pattern is computed, we must first define the value
 275 $h(\mathbf{a}, B)$. This quantity, where \mathbf{a} is a pattern and B is a set of different patterns of the same size, is defined
 276 as follows:

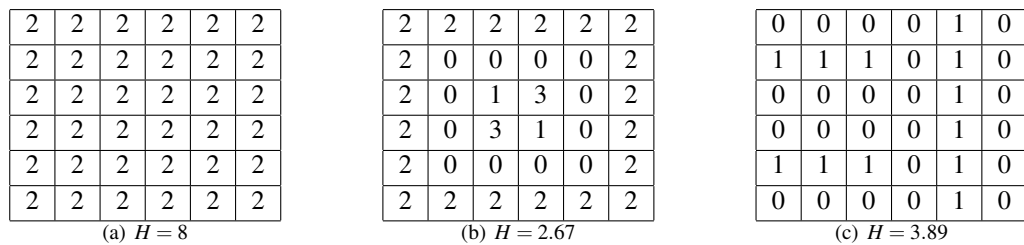


Figure 4. Examples of three 6×6 patterns and their harmonies, computed using levels $N = \{6, 3, 2\}$.

Harmony	Description
h_1	Symmetry about the x axis.
h_2	Symmetry about the y axis.
h_3	Symmetry about the $y = x$ diagonal.
h_4	Symmetry about the $y = -x$ diagonal.
h_5	$\pm 90^\circ$ rotational symmetry.
h_6	180° rotational symmetry.
h_7	Translational symmetry with another pattern.
h_8	Translation plus reflectional symmetry with another pattern.
h_9	Translation plus rotational ($\pm 90^\circ$ or 180°) symmetry with another pattern.

Table 1. The six possible internal symmetries and the three additional hierarchical symmetries required to compute the harmony metric. Each h value is either 1 or 0.

$$h(\mathbf{a}, B) = \sum_{i=1}^6 h_i(\mathbf{a}) + \sum_{i=7}^9 h_i(\mathbf{a}, B) \quad (10)$$

277 The functions h_1-h_6 measure top-level properties of the pattern. The functions h_7-h_9 measure properties
 278 of the pattern in relation to all of the patterns in B . For these three latter functions, if \mathbf{a} matches any of the
 279 elements in B , then 1 is returned; otherwise (or if B is empty) 0 is returned. Therefore $h(\mathbf{a}, B)$ must return
 280 a value either between 0 and 9 if B is non-empty, or 0 and 6 if B is empty.

281 We now define $B_{n \times n}(\mathbf{a})$ to be the set of all non-overlapping $n \times n$ sub-patterns of \mathbf{a} that can be obtained
 282 by dividing \mathbf{a} into $n \times n$ -sized subarrays. It can be seen that if the size of \mathbf{a} is 6×6 then $B_{3 \times 3}$ will have
 283 four distinct elements, $B_{2 \times 2}$ will have nine elements, and $B_{6 \times 6}$ will have one element. If the size of \mathbf{a} is
 284 36×36 , on the other hand, then $|B_{6 \times 6}| = 36$.

285 The final harmony for a pattern \mathbf{a} can thus be defined as:

$$H(\mathbf{a}) = \frac{1}{|N|(\mathbf{a})} \sum_{n \in N} \left[\frac{1}{|B_{n \times n}(\mathbf{a})|} \sum_{\mathbf{b} \in B_{n \times n}(\mathbf{a})} h(\mathbf{b}, B_{n \times n}(\mathbf{a}) \setminus \mathbf{b}) \right] \quad (11)$$

286 where $H(\mathbf{a})$, being an average across values computed by the h function, is also in the range 0 to 9
 287 inclusive. In the definition of H , the set N consists of positive integers which are not greater than (and
 288 preferably divide evenly into) the smallest dimensionality of \mathbf{a} . The elements of N define the sizes of the
 289 sub-patterns to consider. In the original paper (Klinger and Salingaros, 2000), $N = \{6, 3, 2\}$ and we use
 290 the same values, although we do increase the sizes of the patterns being considered from 6×6 to 36×36 .

291 A brief consideration of Equation 11 should make clear fact that what is being computed is the average
 292 of the harmonies of each sub-pattern at the various different scales defined by N . This is represented by
 293 the inner summation, with each individual $n \times n$ sub-pattern being represented by \mathbf{b} in the equation.

294 Then the average harmony across scales, represented by the outer summation and division by $|N|$, is
 295 calculated. The resulting quantity is the final metric. Thus, a higher harmony indicates that the pattern
 296 contains more symmetries at the various scales, while a lower harmony indicates fewer multi-scale
 297 symmetries.

Problem	Direction(s)	Expected Speed(s)	#Wind Scenarios
A	{0°}	{12m/s}	1
B	{0°, 10°, ..., 350°}	{12m/s}	36
C	{0°, 10°, ..., 350°}	{8m/s, 12m/s, 17m/s}	108

Table 2. Problems from Samorani Samorani (2013).

3 OPTIMISING WIND FARM LAYOUTS FOR BOTH ENERGY EFFICIENCY AND HARMONY

We define in this section two new approaches that extend both TDA and the BlockCopy local search method with the harmony metric explained in the previous section. We dub these new approaches TDA* and BC*.

The basis is fairly straightforward. Rather than optimising directly for energy efficiency (i.e. maximising F only, which is defined by Equation 9), we instead replace F with a composite objective function F' obtained by adding F and H . This new objective function is defined as follows:

$$F'(\mathbf{l}) = F(\mathbf{l}) + \lambda H(\text{pattern}(\mathbf{l})) \quad (12)$$

Two key parts of the new approach are (i) a function $\text{pattern}()$ that converts a layout (consisting of real-valued double coordinates) into a symbolic pattern array, and (ii) a parameter λ that specifies how much influence harmony will have in objective calculations.

For the $\text{pattern}()$ function, we simply divide the layout (which is square in our experiments) into 36×36 “cells”. Each cell maps to a symbol in a 36×36 symbol array representing the layout, and corresponds to the number of turbines in that cell. As it turns out, on our test layout, the cells were relatively small and so they only ever contain either one turbine or no turbines due to the minimum turbine distance constraint.

The choice of value for λ was a more difficult decision, however, and we therefore decided to test four different values: 0, corresponding to harmony having no influence on the optimisation process; 0.001 and 0.01, corresponding to harmony having small to medium effects on the objective function; and 0.1, corresponding to harmony being nearly equally weighted with energy efficiency. The choice of these values was made because the range of the H parameter (0..9) is nine times the range of the F parameter (0..1) and thus small values of λ make sense. Any larger values of λ would result in harmony overwhelming the combined objective function.

The case of $\lambda = 0$ is thus our baseline because it essentially reverts the TDA* and BC* approaches back into their original versions.

4 EXPERIMENTS

To evaluate our new modified objective functions, we implemented Jensen’s far wake model and used it to simulate a wind farm layout of size 1.5 kilometers \times 1.5 kilometers, with 64 turbines to be sited. The turbine power curve and the other settings for the wake model are described in Section 2.1. The optimisers used are the algorithms described in the previous sections. Each algorithm was initialised with a starting random layout created by iteratively adding turbines at random locations within the layout bounds, subject to the constraint of not placing any two turbines too closely together, until all 64 turbines were placed.

Samorani (2013) describes three different problems of increasing complexity for testing wind farm layout optimization algorithms, and we adopted these three benchmark problems for our experiments. The problems, referred to as A, B and C, are described by Table 2.

Essentially, Problem A is the simplest benchmark, and comprises a single wind scenario in which wind blows with a single expected speed and in a single constant direction. The set of scenarios S therefore (used in Equation 9 for calculating F) consists of only a single element. Problem B, on the other hand, consists of 36 different wind scenarios, each differing only in the wind direction. Unlike Problem A, there is no dominant wind direction: instead, the expected wind speed is the same for all directions. Although this is an unrealistic setting, it is useful for testing purposes.

Problem C, on the other hand, is the most interesting and challenging benchmark. In Problem C, as in Problem B, there are 36 possible wind directions. In Problem C’s case, however, for each different

Direction	$u_s = 8\text{m/s}$	$u_s = 12\text{m/s}$	$u_s = 17\text{m/s}$
0°-260°	0.00404	0.00865	0.0115
270°	0.00404	0.0107	0.0127
280°	0.00404	0.0121	0.0156
290°	0.00404	0.0141	0.0185
300°	0.00404	0.0138	0.0300
310°	0.00404	0.0190	0.0352
320°	0.00404	0.0138	0.0300
330°	0.00404	0.0141	0.0185
340°	0.00404	0.0121	0.0156
350°	0.00404	0.0107	0.0127

Table 3. Probabilities used for the 108 wind scenarios under Problem C (rounded to three significant figures) derived from a chart in Samorani (2013). Please note that the first row the table, labelled “0°-260°”, represents 27 rows, each of which have the same values. We have written these values once only to prevent unnecessary duplication in the table.

341 direction, there are also three different expected wind speeds which evidently correspond to three different
 342 meteorological conditions. Furthermore, there is a clear dominant wind direction: the probability of the
 343 greatest wind speed (and therefore the greatest power production) is maximised at 310°. In total, this
 344 problem comprises 108 different wind scenarios.

345 Since Samorani (2013) only describes this problem benchmark graphically by means of a histogram of
 346 wind speeds vs. directions, in order to implement this benchmark, we reverse-engineered the probabilities
 347 from his published chart. The probabilities we used for Problem C are given in Table 3.

348 To summarise, our experiments consist of algorithms with a new modified objective function (TDA*
 349 and BC*) with four different λ values on three different benchmark problems. This amounts to 24 different
 350 configurations.

351 Next, for each configuration, we ran 30 repetitions. Each repetition consisted of one run of a local
 352 search algorithm for 20,000 iterations. We note that the number of iterations is significantly higher than
 353 the number of iterations (2,000) in a previous comparison of TDA and BlockCopy (Mayo and Zhen,
 354 2016).

355 5 RESULTS

356 The summary results are depicted in Figures 5–7. The figures show the mean and maximum F and
 357 H values achieved by algorithm over all thirty runs. We note in this section that our use of the term
 358 “performance” refers to the best objective values achieved by the various algorithms and not, as is the
 359 common interpretation, to computational efficiency.

360 Broadly speaking, then, the figures show that BlockCopy local search “outperforms” TDA in terms of
 361 energy efficiency when λ is small. The difference between the algorithms for Problem A is approximately
 362 up to 10%; for Problems B and C the difference is less pronounced.

363 In terms of the harmony metric, however, the BlockCopy approach generally scores a much higher
 364 value than TDA, especially as λ increases. This is most likely due to the approximate preservation of
 365 translational symmetry, a property of the BlockCopy operator discussed earlier.

366 Overall, under both algorithms, the best layouts degrade in terms of energy efficiency as λ increases.
 367 This energy loss is approximately 6% on average for Problem C. Analysing the final best layouts produced
 368 by every 30-run configuration, we found that there is a significant negative correlation between final F
 369 and H values. This confirms that the two objectives, energy efficiency and harmony, tend to trade off.
 370 Testing more advanced multi-objective algorithms may therefore be worthwhile in the future.

371 Next, we were also interested in the statistically significant differences in performances for a more
 372 quantitative comparison.

373 To this end, we performed a post-hoc Tukey Honest Significant Difference test comparing F value
 374 mean between all eight techniques on each of the three different benchmark problems. The test was
 375 performed at 95% significance. The results indicate that for Problems A and B, there is no significant
 376 difference in mean performance between algorithms in the following sets:

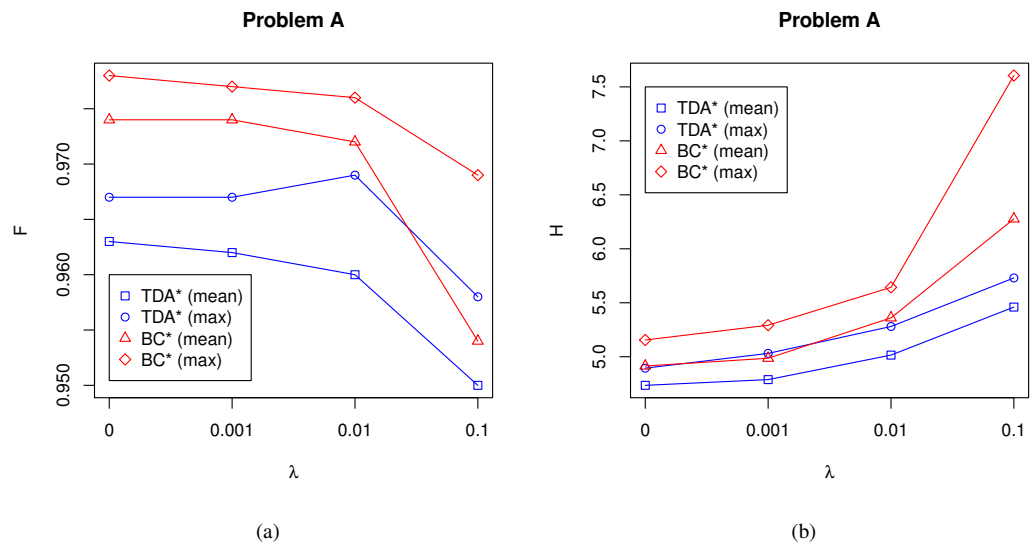


Figure 5. Optimisation performance over 30 runs with different λ values on Problem A

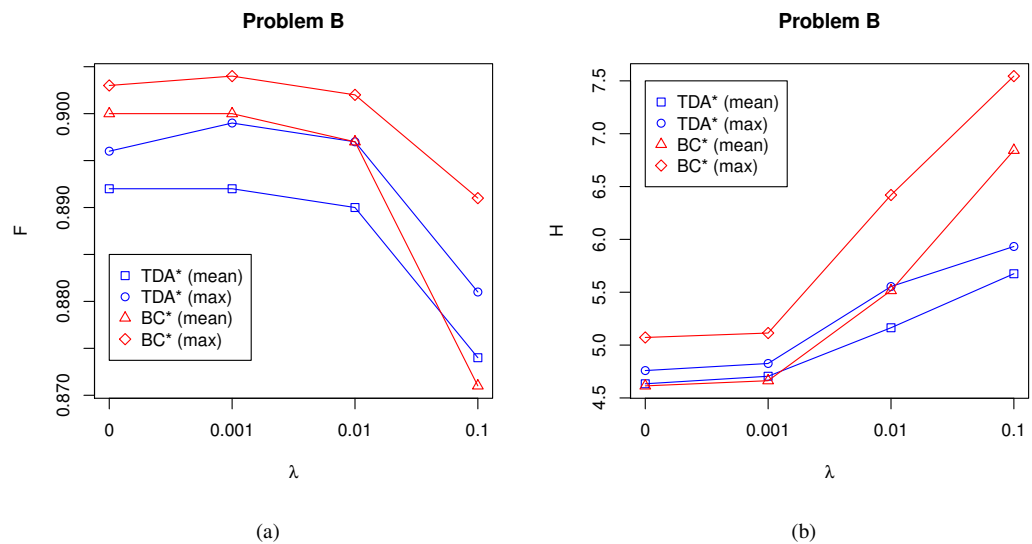


Figure 6. Optimisation performance over 30 runs with different λ values on Problem B

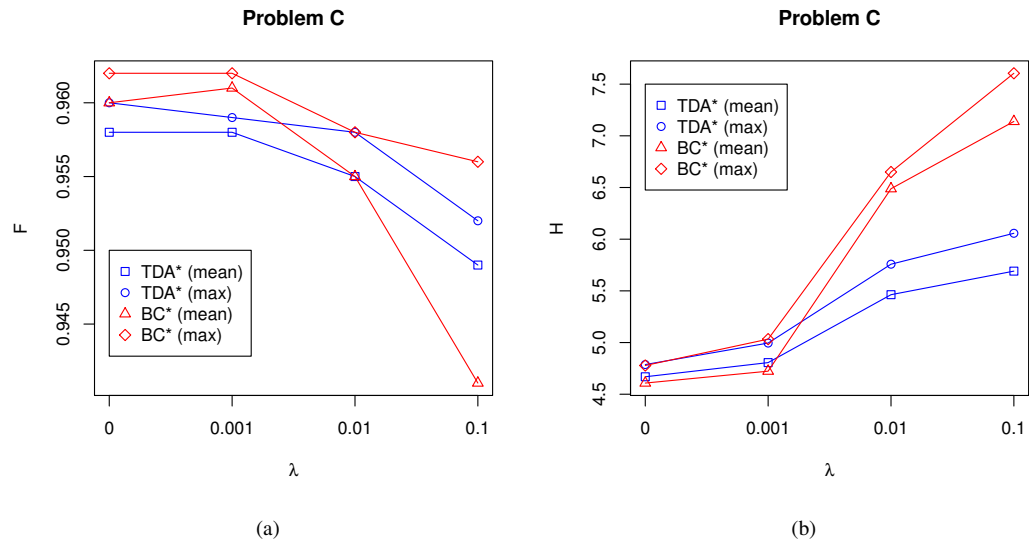


Figure 7. Optimisation performance over 30 runs with different λ values on Problem C

- 377 • $\{(TDA^*, \lambda = 0), (TDA^*, \lambda = 0.001), (TDA^*, \lambda = 0.01)\}$
- 378 • $\{(BC^*, \lambda = 0), (BC^*, \lambda = 0.001), (BC^*, \lambda = 0.01)\}$
- 379 • $\{(TDA^*, \lambda = 0.1), (BC^*, \lambda = 0.1)\}$ (Problem B only)

380 The result of these tests show that small to moderate values for λ do not significantly impact on optimisation performance. The figures quantify the actual difference in mean performances between algorithms.

381 For Problem C, the situation is a slightly more complex. The sets of algorithms with no statistically significant difference in mean performance are:

- 384 • $\{(BC^*, \lambda = 0.001), (BC^*, \lambda = 0), (TDA^*, \lambda = 0.001)\}$
- 385 • $\{(TDA^*, \lambda = 0.001), (TDA^*, \lambda = 0), (BC^*, \lambda = 0.01)\}$
- 386 • $\{(TDA^*, \lambda = 0), (BC^*, \lambda = 0.01), (TDA^*, \lambda = 0.01)\}$
- 387 • $\{(TDA^*, \lambda = 0.1)\}$
- 388 • $\{(BC^*, \lambda = 0.1)\}$

389 These results clearly are more difficult to interpret, which suggests that no conclusion can be drawn without more repetitions of the algorithms.

390 We note at this point that tests for statistical significance only determine the likelihood of average algorithm performance variations. The results, however, show that best-of-run performance is often quite different from the mean. This is illustrated most clearly in Figure 7(a) for the algorithm BC^* with $\lambda = 0.1$: the best layout identified by the algorithm is comparable to average layouts found by the other algorithms with $\lambda < 0.1$, even though the mean F values performance of this algorithm is quite low.

391 Finally, we examined visually some of the optimal layouts found after different runs of the various techniques. Examples of different optimised layouts are shown in Figures 8–10.

392 Focussing firstly on Figure 8, which depicts some layout solutions to Problem A, we can see clearly the difference between the TDA and BC. For the TDA-based layouts (Figure 8(a) and (b)), the arrangement of turbines has a clear random character. This is even the case where the λ value is at its highest in Figure 8(b) – in this case, the H objective is not much different than it is in (a) where it is not optimised at all.

393 In contrast, the BC^* algorithms tend to produce more patterned layouts. For example, Figure 8(c) does not make use of the harmony metric at all but it still produces a degree of translational symmetry

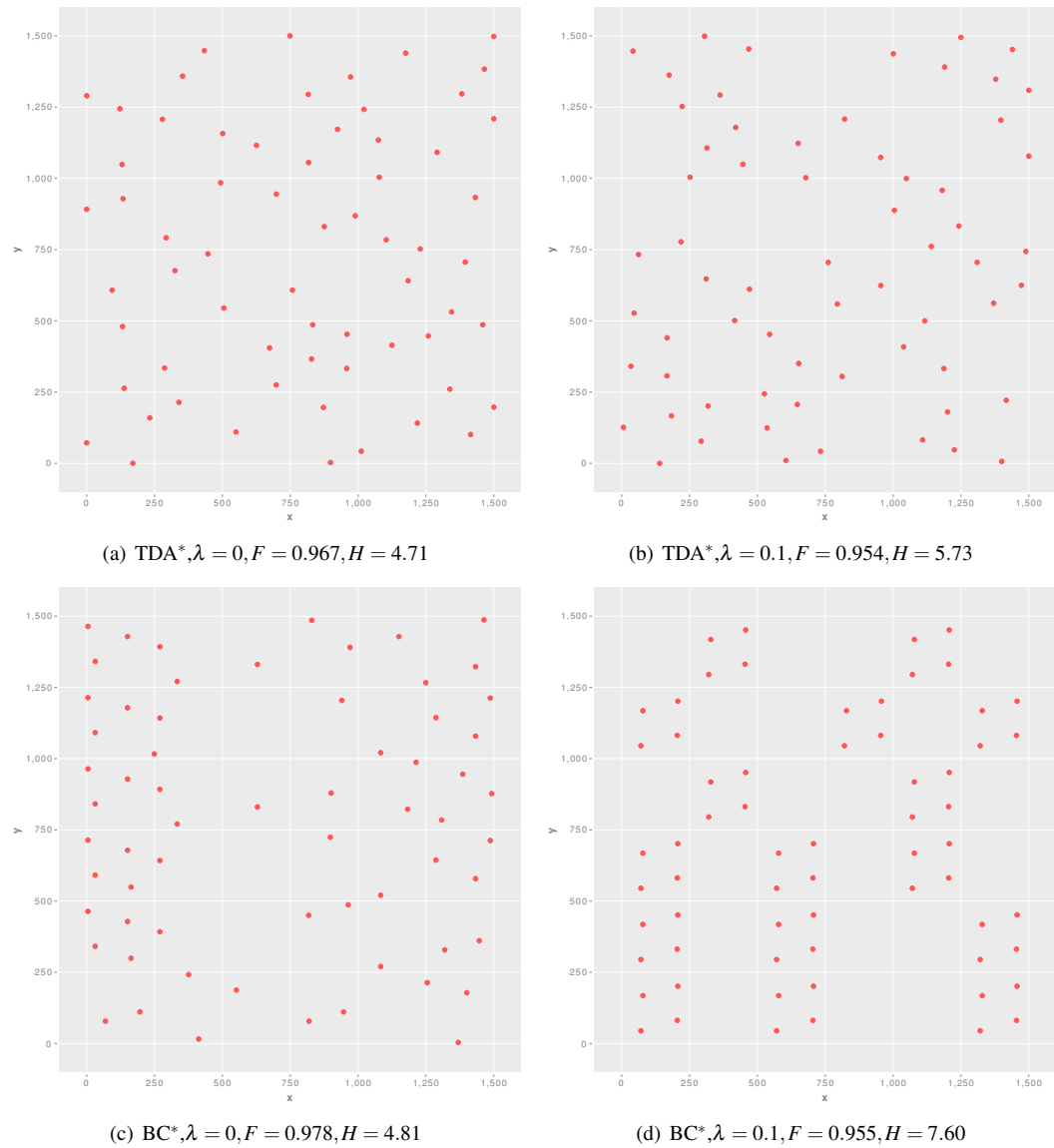


Figure 8. Example layouts produce by TDA^* (a,b) and BC^* (c,d) on Problem A.

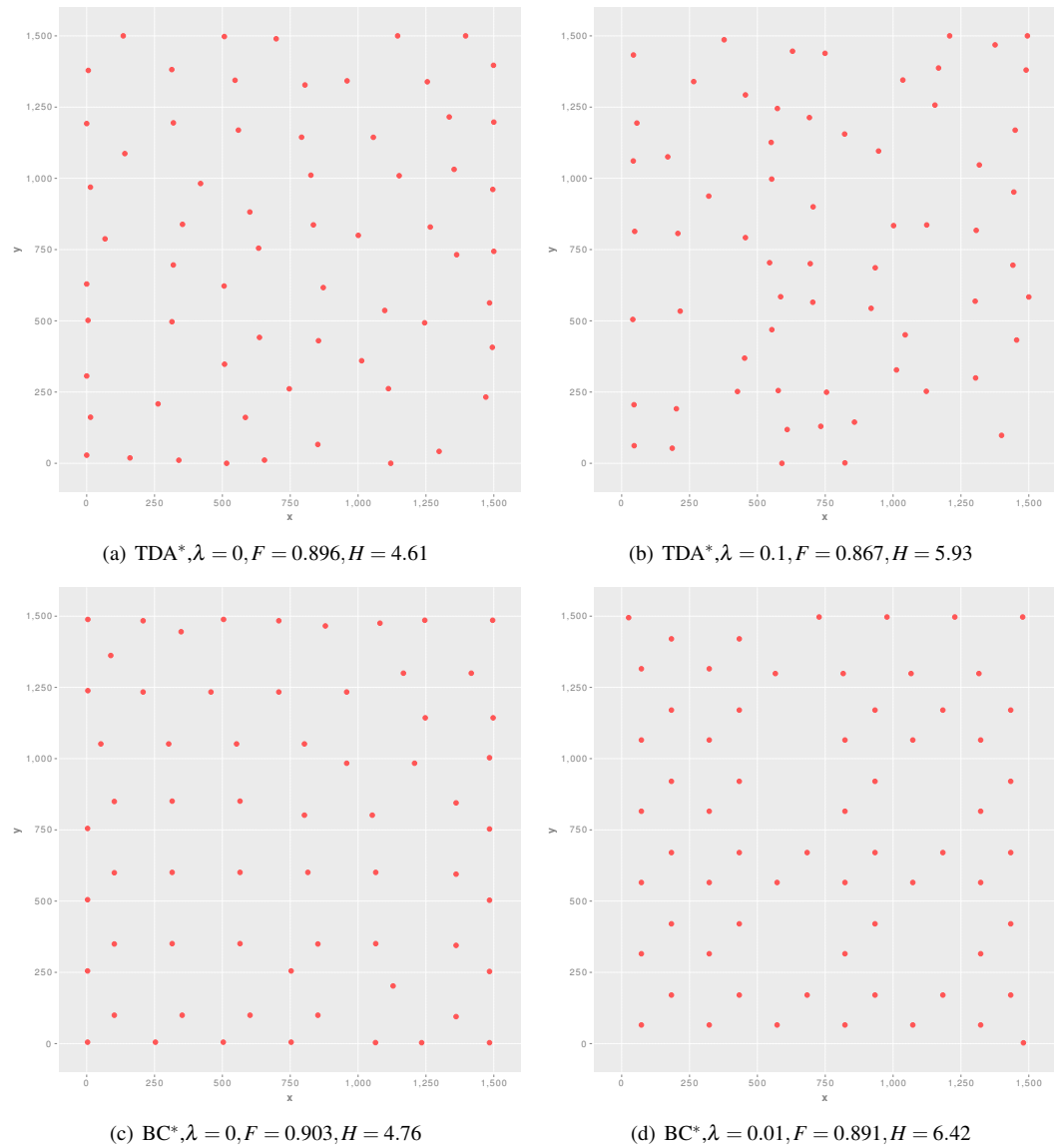


Figure 9. Example layouts produce by TDA^* (a,b) and BC^* (c,d) on Problem B.

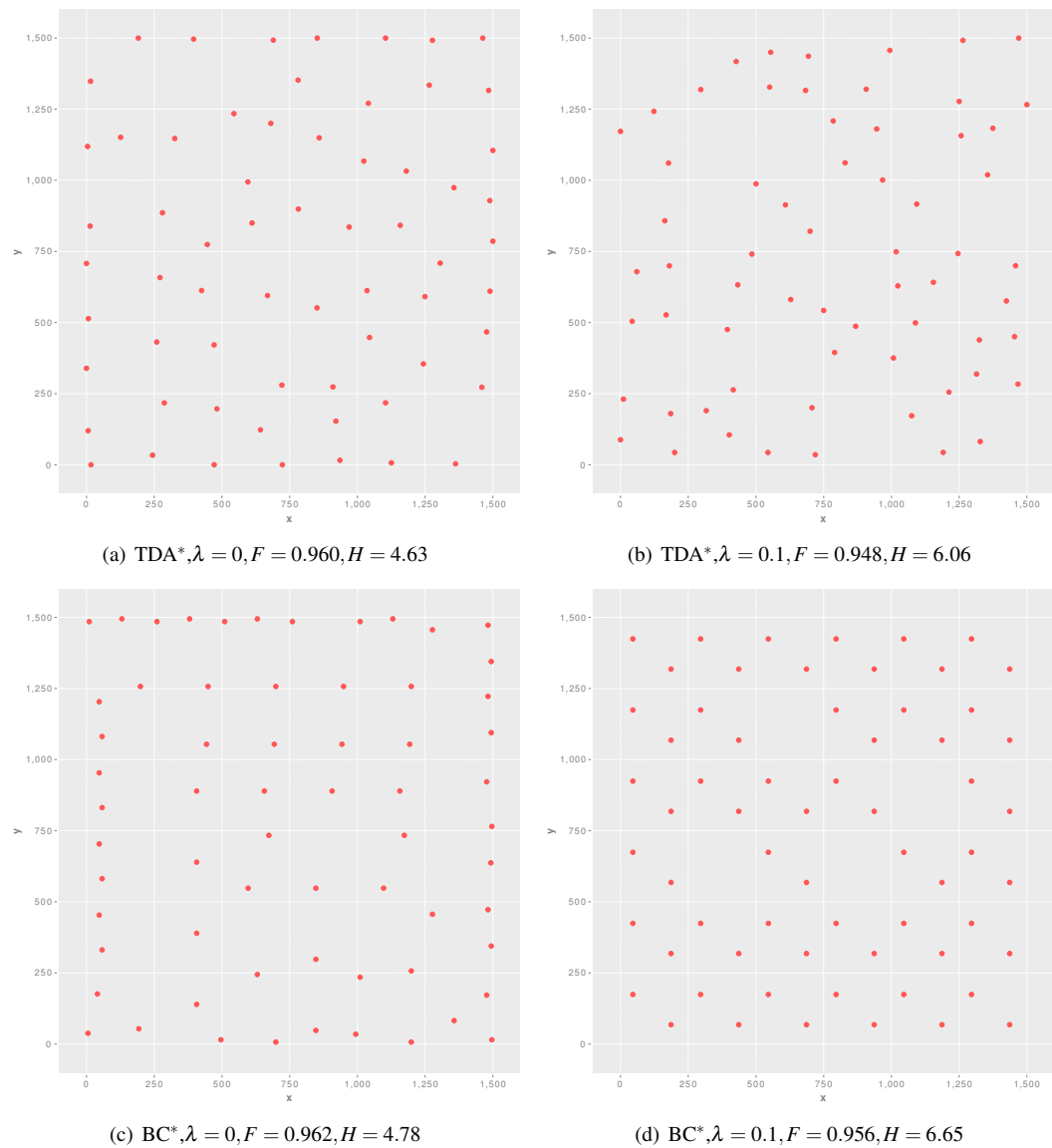


Figure 10. Example layouts produce by TDA^* (a,b) and BC^* (c,d) on Problem C.

404 across the layout. When λ is high however, as in Figure 8(d), the translational symmetry is increased
 405 dramatically to produce a considerably more regular arrangements of turbines.

406 A somewhat different story can be told for the examples shown in Figure 9 which depicts example
 407 solutions to Problem B. This benchmark problem has no dominant wind direction, which becomes evident
 408 when the layouts are examined. To illustrate, both algorithms optimising solely for efficiency (see Figures
 409 9(a) and (c)) tend to push turbines out to the furthest possible edge of the layouts, thus maximising
 410 the space between turbines in all directions. The BC algorithm interestingly produces a layout that is
 411 somewhat grid-like in structure in this case (see Figure 9(c)).

412 Finally, Figure 10 depicts selected solutions to Problem C. Similar trends can be observed in the
 413 solutions to this problem as were observed for the other problems. For example, BC^* produces more
 414 regular layouts than TDA regardless of λ value on this problem. Also of considerable interest with regards
 415 to Problem is the difference in efficiencies achieved: the BC^* algorithm with $\lambda = 0.1$ produces an optimal
 416 layout in the Figure with an efficiency of only about half a percent less than TDA^* 's best layout with
 417 $\lambda = 0$.

418 As a final comment, it will be pointed out that the harmony metric is maximised for patterns that are

419 completely uniform. For example, see Figure 4(a). The practical effect of this bias is that the search is
420 more likely to be focussed on sparse layouts – i.e. layouts with several empty blocks. This is because
421 empty blocks correspond to uniform patterns which maximise H .

422 To illustrate, see Figure 8(d) for an extreme example, and Figure 10(d) for a less extreme example. The
423 practical consequences of this are considerable, since less land can be used for the same or approximately
424 the same efficiency. This may positively impact on both the cost of land (e.g. see Chen and MacDonald
425 (2012)) and the effect on wildlife (e.g. see the survey on environmental implications of wind energy by
426 Tabassum-Abbasi et al. (2014)) and warrants further investigation as well. This finding also suggests that
427 purely regular approaches (e.g. Neubert et al. (2010)) may not be ideal solutions to this problem because
428 such approaches distribute turbines uniformly across the layout without any chance of free space areas
429 such as those in figures appearing.

430 6 CONCLUSION

431 To conclude, we have investigated a metaheuristic optimisation approach to solving the wind farm layout
432 optimisation problem, in which one objectives concerns maximising layout aesthetics while the other
433 concerns maximising energy efficiency. The aesthetics of layouts is an important consideration which
434 most other literature in this field to date has not considered. Our experiments reported here were successful
435 and encourage future research to further refine this initial approach. In particular, while the Jensen model
436 is adequate for this initial work, future work should explore more realistic wind farm simulations that
437 better account for both wind (e.g. Feng and Shen (2015a)) and partial wakes (e.g. Feng and Shen (2015b)).

438 As mentioned in the Introduction section, there are two main limitations of the work presented here.
439 Firstly, our approach is largely constrained to two dimensional layouts, as would typically be encountered
440 offshore or on sites that are plains. Clearly therefore, Salingeros' approach must be generalised to three
441 dimensions before the same ideas can be applied to other types of site. Our current thinking is that there
442 are two possible approaches to this generalisation: (i) topography could be included in the definition of
443 the symbols, which would complicate the definition of what a symbol is somewhat (e.g. a turbine on the
444 top of a knoll would result in a different symbol compared to a turbine at the same relative position but in
445 the middle of plain); or alternatively (ii) the method could be generalised to include transformations in the
446 z dimension – this approach would require expanding the set of symmetries listed in Table 1 to include all
447 3D symmetries as well.

448 The second main limitation of this work is that it focusses on the overhead view of the layout only and
449 assumes by default that aesthetics is uniformly important across the entire layout. This is in fact a false
450 assumption for farms located near places that people frequent such as nearby towns, tourist attractions
451 and highways. In such situations, the aesthetics of the portion of the farm in clear view of the people will
452 be far more important than the parts of the farm that are obscured. Therefore vantage point is important in
453 this situation. We therefore feel that our “global” approach to aesthetics could be complemented by a
454 corresponding “local” approach that takes into account viewpoint. The local approach could take an image
455 aesthetics-based approach and render the farm and its surrounding terrain, skyline and other features as an
456 eye-level scene, and then assess its aesthetics using machine learning. Marchesotti et al. (2015) is one
457 example of such an approach that could be gainfully employed here.

458 REFERENCES

- 459 Al-Yahyai, S., Charabi, Y., and Gastli, A. (2015). Geometrical approach for wind farm symmetrical layout
460 design optimization. In *GCC Conference and Exhibition (GCCCE), 2015 IEEE 8th*, pages 1–6.
- 461 Chen, L. and MacDonald, E. (2012). Considering landowner participation in wind farm layout optimization.
462 *Journal of Mechanical Design*, 134(8):084506–084506.
- 463 Dai, K., Bergot, A., Liang, C., Xiang, W.-N., and Huang, Z. (2015). Environmental issues associated with
464 wind energy – a review. *Renewable Energy*, 75:911 – 921.
- 465 den Heijer, E. and Eiben, A. (2010). Comparing aesthetic measures for evolutionary art. In *Applications*
466 *of Evolutionary Computation*, volume 6025 of *Lecture Notes in Computer Science*, pages 311–320.
- 467 Feng, J. and Shen, W. (2015a). Modelling wind for wind farm layout optimisation using joint distribution
468 of wind speed and wind direction. *Energies*, 8:3075–3092.
- 469 Feng, J. and Shen, W. Z. (2015b). Solving the wind farm layout optimization problem using random
470 search algorithm. *Renewable Energy*, 78(0):182 – 192.

- 471 Galanter, P. (2012). Computational aesthetic evaluation: Past and future. In McCormack, J. and d’Inverno,
472 M., editors, *Computers and Creativity*, chapter 11, pages 255–293. Springer Berlin Heidelberg.
- 473 Global Wind Energy Council (2014). *Global Wind Energy Outlook 2014*.
- 474 Guirguis, D., Romero, D. A., and Amon, C. H. (2016). Toward efficient optimization of wind farm
475 layouts: Utilizing exact gradient information. *Applied Energy*, 179:110 – 123.
- 476 Jensen, N. (1983). A note on wind generator interaction. Technical report, Risø DTU National Laboratory
477 for Sustainable Energy.
- 478 Katic, I., Høstrup, J., and Jensen, N. (1986). A simple model for cluster efficiency. In *Proc. Europe and*
479 *Wind Energy Association Conference and Exhibition*.
- 480 Klinger, A. and Salingaros, N. (2000). A pattern measure. *Environment and Planning B: Planning and*
481 *Design*, 27:537–547.
- 482 Marchesotti, L., Murray, N., and Perronin, F. (2015). Discovering beautiful image attributes for aesthetic
483 image analysis. *International Journal of Computer Vision*, 113:246–266.
- 484 Mayo, M. and Daoud, M. (2016). Informed mutation of wind farm layouts to maximise energy harvest.
485 *Renewable Energy*, 89:437–448.
- 486 Mayo, M., Daoud, M., and Zheng, C. (2016). Randomising block sizes for blockcopy-based wind farm
487 layout optimisation. In *Proc 20th Asia Pacific Symposium on Intelligent and Evolutionary Systems*,
488 pages 277–289.
- 489 Mayo, M. and Zhen, C. (2016). Blockcopy-based operators for evolving efficient wind farm layouts. In
490 *Proc 2016 IEEE Congress on Evolutionary Computation*, pages 1085–1092.
- 491 Mosetti, G., Poloni, C., and Diviacco, B. (1994). Optimization of wind turbine positioning in large wind
492 farms by means of a genetic algorithm. *Journal of Wind Engineering and Industrial Aerodynamics*,
493 51(1):105–116.
- 494 Neubert, A., Shah, A., and Schlez, W. (2010). Maximum yield from symmetrical wind farm layouts. In
495 *Proc. 10th German Wind Energy Conference, DEWEK*.
- 496 Piorkowski, M. D., Farnsworth, A. J., Fry, M., Rohrbaugh, R. W., Fitzpatrick, J. W., and Rosenberg,
497 K. V. (2012). Research priorities for wind energy and migratory wildlife. *The Journal of Wildlife*
498 *Management*, 76(3):451–456.
- 499 Rodrigues, S. M. F., Bauer, P., and Pierik, J. (2013). Modular approach for the optimal wind turbine micro
500 siting problem through CMA-ES algorithm. In *Proceedings of the 15th Annual Conference Companion*
501 *on Genetic and Evolutionary Computation*, GECCO ’13 Companion, pages 1561–1568. ACM.
- 502 Salingaros, N. (1997). Life and complexity in architecture from a thermodynamic analogy. *Physics*
503 *Essays*, 10(1).
- 504 Samorani, M. (2013). The wind farm layout optimization problem. In Pardolas, P., editor, *Handbook of*
505 *Wind Power Systems*, pages 21–38. Springer-Verlag.
- 506 Shakoor, R., Hassan, M., Raheem, A., and Wu, Y. (2016). Wake effect modelling: A review of wind farm
507 layout optimization using Jensen’s model. *Renewable and Sustainable Energy Reviews*, 58:1048–1059.
- 508 Song, Z., Zhang, Z., and Chen, X. (2016). The decision model of 3-dimensional wind farm layout design.
509 *Renewable Energy*, 85:248 – 258.
- 510 Tabassum-Abbasi, Premalatha, M., Abbasi, T., and Abbasi, S. (2014). Wind energy: Increasing de-
511 ployment, rising environmental concerns. *Renewable and Sustainable Energy Reviews*, 31:270 –
512 288.
- 513 Tsoutsos, T., Gouskos, Z., Karterakis, S., and Peroulaki, E. (2006). Aesthetic impact from wind parks.
514 Technical report, Technical University of Crete, Chania, Greece.
- 515 Wagner, M., Day, J., and Neumann, F. (2013). A fast and effective local search algorithm for optimizing
516 the placement of wind turbines. *Renewable Energy*, 51:64–70.
- 517 Watts, J. (2012). Winds of change blow through china as spending on renew-
518 able energy soars, [http://www.theguardian.com/world/2012/mar/19/
519 china-windfarms-renewable-energy](http://www.theguardian.com/world/2012/mar/19/china-windfarms-renewable-energy). *The Guardian*.
- 520 Wilson, D., Cussat-Blanc, S., Veeramachaneni, K., O’Reilly, U., and Luga, H. (2014). A continuous
521 development model for wind farm layout optimization. In *Proceedings of the 2014 Conference on*
522 *Genetic and Evolutionary Computation*, GECCO ’14, pages 745–752, New York, NY, USA. ACM.