Towards the Coevolution of Novel Vertical-Axis Wind Turbines

Richard J. Preen and Larry Bull

Abstract—Renewable and sustainable energy is one of the most important challenges currently facing mankind. Wind has made an increasing contribution to the world’s energy supply mix, but still remains a long way from reaching its full potential. In this paper, we investigate the use of artificial evolution to design vertical-axis wind turbine prototypes that are physically instantiated and evaluated under approximated wind tunnel conditions. An artificial neural network is used as a surrogate model to assist learning and found to reduce the number of fabrications required to reach a higher aerodynamic efficiency. Unlike in other approaches, such as computational fluid dynamics simulations, no mathematical formulations are used and no model assumptions are made. Initially a conventional evolutionary algorithm is used to explore the design space of a single wind turbine and later a cooperative coevolutionary algorithm is used to explore the design space of an array of wind turbines.

Index Terms—Coevolution, surrogate-assisted evolution, three-dimensional printers, wind turbines.

I. INTRODUCTION

In recent years, wind has made an increasing contribution to the world’s energy supply mix. However, there is still much to be done in all areas of the technology for it to reach its full potential. Currently, horizontal-axis wind turbines (HAWTs) are the most commonly used form. However, “modern wind farms comprised of HAWTs require significant land resources to separate each wind turbine from the adjacent turbine wakes. This aerodynamic constraint limits the amount of power that can be extracted from a given wind farm footprint. The resulting inefficiency of HAWT farms is currently compensated by using taller wind turbines to access greater wind resources at high altitudes, but this solution comes at the expense of higher engineering costs and greater visual, acoustic, radar and environmental impact” [1]. This has forced wind energy systems away from high energy demand population centres and towards remote locations with higher distribution costs. In contrast, vertical-axis wind turbines (VAWTs) do not need to be oriented to wind direction and can be positioned closely together, potentially resulting in much higher efficiency. VAWT can also be easier to manufacture, may scale more easily, are typically inherently light-weight with little or no noise pollution, and are more able to tolerate extreme weather conditions (see, e.g., [2] for discussions). However, their design space is complex and relatively unexplored. Generally, two classes of design are currently under investigation and exploitation: the Savonius, which has blades attached directly upon the central axis structure; and the Darrieus, where the blades—either straight or curved—are positioned predominantly away from the central structure. Hybrids also exist.

The majority of blade design optimisation is performed through the use of computational fluid dynamics (CFD) simulations, typically described with three-dimensional Navier-Stokes equations [3]. However, three-dimensional CFD simulations are computationally expensive, with a single calculation taking hours on a high-performance computer, making their use with an iterative search approach difficult [4]. Moreover, assumptions need to be made, e.g., regarding turbulence or pressure distributions, which can significantly affect accuracy. Previous evolutionary studies have been undertaken with types of CFD to optimise the blade profile for both HAWT (e.g., [5]) and VAWT (e.g., [6]) to varying degrees of success/realism.

Evolutionary algorithms (EAs) have been used to design three-dimensional physical objects, such as furniture (e.g., [7]), aircraft engine blades (e.g., [8]) and wings (e.g., [9]). Notably, Lohn et al. [10] evolved and manufactured X-band satellite antenna for NASA’s ST5 spacecraft, representing the world’s first artificially evolved hardware in space. Significantly, the antenna’s performance was similar to a design hand-produced by an antenna-contractor. Most of these approaches, however, have used simulations to provide the fitness scores of the evolved designs.

The evaluation of physical artifacts for fitness determination can be traced back to the origins of evolutionary computation; for example, the first evolution strategies (ESs) were used to design jet nozzles as a string of real-valued diameters, which were then machined and tested for fitness [11]. Other well-known examples include robot controller design (e.g., [12]), electronic circuit design using programmable hardware (e.g., [13]), product design via human provided fitness values (e.g., [14]), chemical systems (e.g., [15]), and unconventional computers (e.g., [16]). Evolution in hardware has the potential to benefit from access to a richer environment where it can exploit subtle interactions that can be utilised in unexpected ways. For example, the EA used by Thompson [13] to work with field-programmable gate array circuits used some subtle physical properties of the system to solve problems where the properties used are still not understood to this day. Humans can be prevented from designing systems that exploit these subtle and complex physical characteristics through their lack of knowledge, however this does not prevent exploitation through artificial evolution. There is thus a real possibility that evolution in hardware may allow the discovery of new physical effects, which can be harnessed for computa-
tion/optimisation [17].

Moreover, the advent of high quality, low-cost, additive rapid fabrication technology—known as three-dimensional printing—means it is now possible to fabricate a wide range of prototype designs quickly and cheaply. Three-dimensional printers are now capable of printing an ever growing array of different materials, including food (e.g., chocolate [18] and meat [19] for culinary design), sugar (e.g., to help create synthetic livers [20]), chemicals (e.g., for custom drug design [21]), cells (e.g., for functional blood vessels [22] and artificial cartilage [23]), plastic (e.g., for prosthetics such as the synthetic mandible developed by the University of Hasselt and transplanted into an 83-year old woman), and liquid metal (e.g., for stretchable electronics [25]). One potential benefit of the technology is the ability to perform fabrication directly in the target environment; for example, Cohen et al. [26] recently used a three-dimensional printer to perform a minimally invasive repair of the cartilage and bone of a calf femur in situ.

Lipson and Pollack [27] were the first to exploit the emerging technology in conjunction with an EA using a simulation of the mechanics and control, ultimately printing mobile robots with embodied neural network controllers.

Techniques to reduce the number of candidate solution evaluations when they are computationally expensive or difficult to obtain/formulate have been developed as evolutionary computation has been applied to more complex domains, e.g., in systems where a human user is involved [28]. This is typically achieved through the construction of models of the problem space via direct sampling—the use of approximations is an established approach in the wider field of optimisation. That is, the evolutionary process uses one or more models to provide the (approximate) utility of candidate solutions, thereby reducing the number of real evaluations during iterations. Initially, all candidate solutions must be evaluated directly on the task to provide rudimentary training data for the modelling, e.g., by neural networks. Periodically, high utility solutions suggested by the model optimisation are then evaluated by the real system. The training data for the model is then augmented with these and the model(s) updated. Over time, as the quality of the model(s) improves, the need to perform real evaluations/fabrications reduces.

In this paper, we present results from a pilot study of surrogate-assisted EAs (SGAs) used to design VAWTs wherein prototypes are evaluated under approximated wind tunnel conditions after being physically instantiated by a three-dimensional printer. That is, unlike other approaches, no mathematical formulations are used and no model assumptions are made. Initially, artificial evolution is used to explore the design space of a single isolated VAWT and subsequently a coevolutionary genetic algorithm (CGA) is applied to explore the design space of an array of 2 closely positioned VAWTs. Both conventional EA and surrogate-assisted versions are examined. To date, no prior work has explored the use of three-dimensional printing within surrogate-assisted embodied evolution.

II. RELATED WORK

The evolution of geometric models to design arbitrary three-dimensional morphologies has been widely explored. Early examples include Watabe and Okino’s lattice deformation approach [29] and McGuire’s sequences of polygonal operators [30]. Sims [31] evolved the morphology and behaviour of virtual creatures that competed in simulated three-dimensional worlds with a directed graph encoding. Bentley [32] investigated the creation of three-dimensional solid objects via the evolution of both fixed and variable length direct encodings. The objects evolved included tables, heat sinks, penta-prisms, boat hulls, aerodynamic cars, as well as hospital department layouts. Eggenberger [33] evolved three-dimensional multicellular organisms with differential gene expression. Jacob and Nazir [34] evolved polyhedral objects with a set of functions to manipulate the designs by adding stellating effects, shrinking, truncating, and indenting polygonal shapes. More recently, Jacob and Hushlak [35] used an interactive evolutionary approach with L-systems [36] to create virtual sculptures and furniture designs.

EAs have also been applied to aircraft wing design (e.g., [9]) including aerodynamic transonic aerofoils (e.g., [37], [38]), and multidisciplinary blade design (e.g., [39]). Few evolved designs, however, have been manufactured into physical objects. Conventionally evolved designs tend to be purely descriptive, specifying what to build but not how it should be built. Thus, there is always an inherent risk of evolving interesting yet unbuildable objects. Moreover, high-fidelity simulations are required to ensure that little difference is observed once the virtual design is physically manifested. In highly complex design domains, such as dynamic objects, the difference between simulation and reality is too large to manufacture designs evolved under a simulator, and in others the simulations are extremely computationally expensive.

Funes and Pollack [40] performed one of the earliest attempts to physically instantiate evolved three-dimensional designs by placing physical LEGO bricks according to the schematics of the evolved individuals. A direct encoding of the physical locations of the bricks was used and the fitness was scored using a simulator which predicted the stability of the composed structures. Additionally, Hornby and Pollack [41] used L-systems to evolve furniture designs, which were then manufactured by a three-dimensional printer. They found the generative encoding of L-systems produced designs faster and with higher fitness than a non-generative system. Generative systems are known to produce more compact encodings of solutions and thereby greater scalability than direct approaches (e.g., see [42]).

The generative encoding, compositional pattern producing networks [43] have recently been used to evolve three-dimensional objects which were ultimately fabricated on a three-dimensional printer [44]. Both interactive and target-based approaches were explored.

Recently, Rieffel and Sayles [45] evolved circular two-dimensional shapes where each design was fabricated on a three-dimensional printer before assigning fitness values. Interactive evolution was undertaken wherein the fitness for
each printed shape was scored subjectively. Each individual’s genotype consisted of twenty linear instructions which directed the printer to perform discrete movements and extrude the material. As a consequence of performing the fitness evaluation in the environment, that is, after manufacture, the system as a whole can exhibit epigenetic traits, where phenotypic characteristics arise from the mechanics of assembly. One such example was found when selecting shapes that most closely resembled the letter ‘A’. In certain individuals, the cross of the pattern was produced from the print head dragging a thread of material as it moved between different print regions and was not explicitly instructed to do so by the genotype.

The application of EAs can be prohibitive when the evaluations are computationally expensive, an explicit mathematical fitness function is unavailable, or the original fitness function is noisy or multi-modal. Whilst the speed and cost of rapid-prototyping continues to improve, fabricating an evolved design before fitness can be assigned remains an expensive task when potentially thousands of evaluations are required (e.g., 10 mins print time for each very simple individual in [47]). A growing body of work is exploring the application of surrogate models (also known as meta-models or response surface models) to provide approximated fitness computations that assist the EA. The use of surrogate models has been shown to reduce the convergence time in evolutionary computation and multiobjective optimisation; see [48], [49], [50] for recent reviews. Alternative methods, such as fitness inheritance, and assignment can also be used.

Given a sample \( D \) of evaluated treatments \( N \), a surrogate model, \( y = f(\vec{x}) \), is constructed, where \( \vec{x} \) is the genotype, and \( y \), fitness, in order to compute the fitness of an unseen data point \( \vec{x} \notin D \). As such, the genotype must be sufficiently compact for the model to optimise. Typically, a set of evaluated genotypes and their real fitness scores are used to perform the supervised training of an MLP-based artificial neural network (e.g., [51]); however, other approaches have been explored, e.g., kriging [52], clustering [53], support vector regression [54], radial-basis functions [55], and sequential parameter optimisation [56]. The surrogate model is subsequently used to compute estimated fitness values for the EA to utilise. The model must be periodically retrained with new individuals under a controlled evolutionary approach to prevent convergence on local optima. Retraining can be performed by taking either an individual or generational approach [57]. In the individual approach, \( n \) number of individuals in the population are chosen and evaluated with the real fitness function each generation. In the generational approach, the entire population is evaluated on the real fitness function each \( n \)-th generation. Resampling methods and surrogate model validation remain an important and ongoing area of research, enabling the comparison and optimisation of models [58]. Both global modelling and local modelling using trust regions (e.g., samples within a certain euclidean distance) are popular approaches.

Surrogate assisted EAs that use CFD analysis for fitness determination have previously been used to design turbine blades, finding interesting solutions with reduced computational time [59]. Jin et al. [60] explored an ES with CFD analysis to minimise the pressure loss of a turbine blade while maintaining a certain outflow angle. The blade representation used consisted of a series of B-spline control points. The population was initialised with a given blade and 2 neural networks were used to approximate the pressure loss and outflow angle, finding faster convergence than without the surrogate models. Gränning et al. [4] used an ES with covariance matrix adaptation to minimise the pressure loss of a blade using three-dimensional CFD simulations. The ES was augmented by a neural network surrogate model and used a pre-selection resampling approach (where offspring are only generated from individuals evaluated on the real fitness function), however significant improvement over a plain ES was not found.

The surrogate assisted evolution of aerofoil geometries (a type of blade) has been widely explored for use with aircraft design. Some examples include, Giotis and Giannakoglou [61] who used multiple output neural networks as surrogate models for multiobjective aerofoil optimisation. Emmerich and Naujoks [62] and Kumano et al. [63] used kriging to provide approximations for multiobjective aerofoil design. In addition, Zhou et al. [64] evolved aerodynamic aerofoil geometries with a representation consisting of Hicks-Henne bump function parameters. The EA was assisted by both a global and local surrogate model. Significantly, these approaches use simulations to evaluate candidate solutions and typically consider only two-dimensional blades (due to the cost of CFD analysis).

### III. Methodology

A vector of 10 integers is here used as a simple and compact encoding of a prototype VAWT. Each allele thus controls 1/10th of a blade. A workspace (maximum object size) of \( 30 \times 30 \times 30 \text{mm} \) is used so that the instantiated prototype is small enough for timely production (∼30 mins) and with low material cost, yet large enough to be sufficient for fitness evaluation. The workspace has a resolution of \( 100 \times 100 \times 100 \) voxels. A central platform is constructed for each individual to enable the object to be placed on to the evaluation equipment. The platform consists of a square torus, 1 voxel in width and with a centre of \( 14 \times 14 \times 100 \) empty voxels. An equilateral cross is then constructed using the genotype, with four blades bent at right angles, resulting in an allele range [1,42]. More specifically, each blade is constructed, starting from the central platform, by enabling the one-tenth of voxels controlled by the allele.

A simple approach to drawing the blades would be to use the allele value to mark the upper position (e.g., allele + centreline) and enable all voxels from that point towards the centreline; where the centreline is a horizontal line at \( y \)-axis=50 for north-east and south-west quadrants, and a vertical line at \( x \)-axis=50 for north-west and south-east quadrants. However, to provide more flexibility the following rules are applied. Where the current upper position is greater than or the same as the previous upper position, the voxels are enabled from the current upper position to the previous upper position.
and extended a further 2 voxels; see Fig. 1(c). In all other cases, 2 voxels are enabled from the current upper position towards the centreline; see Fig. 1(d). Once the base voxel layer is constructed, it is duplicated to fill the cube in the z-dimension. When production is desired, the three-dimensional binary voxel array is converted to stereolithography (STL) format where it may then undergo post-processing before being converted to printer-readable G-code.

Fig. 2 shows an example phenotype. Fig. 3 shows the same phenotype with 50 Laplacian smoothing steps subsequently applied to the object with MeshLab. Fig. 4 shows the smoothed object after fabrication by a three-dimensional printer.

The genetic algorithm (GA) used herein proceeds with a population of 20 individuals, a maximum mutation step size of ±10, and a per allele mutation rate of 25%. A tournament size of 2 is used for both selection and replacement.

Following previous work on constructing surrogate models, here a 3 layer, multilayer perceptron-based, artificial neural network is used; composed of 10 input neurons, 5 hidden neurons, 1 output neuron, and trained with backpropagation. The model input is the genome (scaled [-1,1]) and the computed output is the predicted fitness. Initially the entire population is evaluated on the real fitness function and the model is trained using backpropagation for 1,000 epochs; where an epoch consists of randomly selecting, without replacement, an individual from the evaluation set and updating the model weights. Each generation thereafter, the fittest unevaluated individual and a randomly chosen unevaluated individual are evaluated on the real fitness function and the model is iteratively retrained from the entire set of evaluated [genotype, real-fitness] pairs. The model parameters, \( \beta = 0.3, \theta = 0, \text{elasticity} = 1, \text{calming rate} = 1, \text{momentum} = 0, \)

\[1\text{MeshLab is an open source, portable, and extensible system for the processing and editing of unstructured 3D triangular meshes. http://meshlab.sourceforge.net}\]

IV. EXPERIMENTATION

A. Tip Speed-Based Evolution

As a first step towards the evolution of novel VAWTs, here the fitness computation for each individual becomes the maximum tip speed achieved during the application of constant wind generated by an approximated wind tunnel after fabrication by a three-dimensional printer (300mm propeller fan; 3,500rpm; treatment placed at 30mm distance; 4.4m/s wind speed). The tip speed is the significant measure of aerodynamic efficiency since the design space is constrained (including rotor radius and turbine height). However, in future work, the AC voltage generated will be preferred, which will take into account any slight weight variations that may affect performance. The tip speed is here measured in number of revolutions per minute (rpm) with a digital photo laser tachometer by placing a 10 × 2mm strip of reflecting tape on the outer tip of one of the treatment’s blades.

Initially, 20 random designs are generated, fabricated, and evaluated. Since many of the seed treatments are extremely aerodynamically inefficient (only 2 out of 20 yielded > 0rpm),
the GA is run for 2 further generations before the model-
augmented approach is used for comparison. The initial pilot
results from an experiment with the canonical GA and SGA
are presented in Fig. 5 which shows the maximum tip
speed achieved by the fittest treatments in each generation.
The GA and model-assisted approach identify increasingly
efficient aerodynamic designs, and the SGA shows improved
performance (1176 rpm vs. 1096 rpm after 100 fabrications).
The fittest treatments produced by the GA and SGA each
generation are shown in Figs. 6 and 7, respectively.

In order to provide an encoding simple enough for the
surrogate model to map over, the turbine representation used so
far has restricted the morphology in the z dimension. However,
more flexibility potentially enables the EA to discover fitter
solutions. To enable z-axis variability, the genome is extended
to include 5 additional parameters in the range [-42,42], each
controlling 1/6th of the z-axis. After drawing the top layer
as before, each new parameter transforms the genome for the
next successive z-layer by uniformly adding the allele value
(capped at the usual bounds), after which it is then drawn in
the usual way. The surrogate model is extended from 10 input
neurons to 15 to incorporate the additional genes.

Fig. 8 shows the maximum tip speed achieved by the fittest
treatments with z-variability in each generation. The fittest
treatments produced by the GA and SGA each generation are
shown in Figs. 9 and 10, respectively. Again, both the GA-
only and model-assisted approach design increasingly efficient
prototypes. Comparing the final 10 treatments from each
experiment, the average tip speed of the SGA ($M = 1217$,
$SD = 78$, $N = 10$) is significantly greater than the GA-
only approach ($M = 1110$, $SD = 41$, $N = 10$) using a
two-sample t-test assuming unequal variances, $t(14) = 2.14$,
$p < .0018$. Furthermore, the fittest treatment designed by
the SGA (1308 rpm) was greater than the GA-only approach
(1245 rpm) after 100 fabrications. The addition of an extra
degree of freedom on the z-axis resulted in improved perfor-
mance for both GA and model-assisted approaches (cf. Fig. 5).

B. Array Tip Speed-Based Evolution

Additional challenges are encountered when extracting large
amounts of wind energy since multiple turbines must be
arranged into a wind farm. As the turbines extract the energy
from the wind, the energy content decreases and the amount
of turbulence increases downstream from each. See [66] for
photographs and explanation of the well-known wake effect
at the Horns Rev offshore wind farm in the North Sea. Due
to this, HAWTs must be spaced 3–5 turbine diameters apart
in the cross-wind direction and 6–10 diameters apart in the
downwind direction in order to maintain 90% of the perfor-
The performance of isolated HAWTs \[1\]. The study of these wake effects is therefore a very complex and important area of research (e.g., see \[67\]), as is turbine placement (e.g., see \[68\] for an evolutionary approach). This work has almost exclusively considered HAWT. However, Dabiri et al. \[69\] have recently highlighted how the spacing constraints of HAWT often do not apply for VAWT, and even that performance can be increased by the exploitation of inter-turbine flow effects. Indeed, it has been shown \[1\] that power densities an order of magnitude greater can be potentially achieved by arranging VAWTs in layouts utilising counter-rotation that enable them to extract energy from adjacent wakes and from above the wind farm.

The use of approximations in a coevolutionary context has previously been shown capable of solving computationally expensive optimisation problems with varying degrees of epistasis more efficiently than conventional CGAs through the use of radial basis functions \[70\] and memetic algorithms \[71\].

Here, we investigate a surrogate-assisted cooperative coevolutionary approach (SCGA) to design wind farms, utilising the aggregated tip speed of the array as fitness. Each VAWT is treated separately by evolution and approximation techniques, i.e., heterogeneous designs could emerge. In addition, a Boolean gene is added to designate the rotation of a turbine in order to allow counter-rotating arrays to potentially emerge. Two turbines are positioned 33 mm adjacently and 30 mm from the propeller fan; that is, there is a 3 mm spacing between the blades at their closest point. Each species population initially consists of the first 10 (> 0 rpm) treatments from the previous experiment, both normally rotated and counter-rotated. This is done to help increase performance and save fabrication time since the treatments still retain a good degree of randomness while possessing some useful aerodynamic properties. Each species thus has a population size of 20.

The treatments in each species population are initially evaluated in collaboration with a single randomly selected treatment from the other species population. Thereafter, the GA proceeds as before, however alternating between species after each offspring is formed and evaluated with the elite member from the other species; see outline in Alg. \[1\]. In the SCGA the models use identical parameters to the previous experiments, however 16 input neurons are now required. In addition, the model weights are reinitialised each time before training due to the temporal nature of pairing with the elite member, and the GA runs for one generation (using the model approximated fitnesses where real fitness is unknown) before the fittest unevaluated individual and a randomly selected
Algorithm 1: Coevolutionary genetic algorithm

1. Generate and fabricate individuals for all species;
2. for each species population do
   3. Select random representative from each species;
   4. for each individual in population do
      5. Evaluate;
   6. end
3. end
4. while fabrication budget not exhausted do
   5. for each species population do
      6. Create an offspring using evolutionary operators;
      7. Select representatives for each species;
      8. Fabricate and evaluate the offspring;
      9. Add offspring to species population;
   10. end
6. end
7. while fabrication budget not exhausted do
8. end
9. for each species population do
10. Initial model weights;
11. Train model on species evaluated list;
12. for each individual in population do
13. if individual unevaluated then
14. Set approximated fitness;
15. end
16. end
17. for population size number of times do
18. Create offspring using evolutionary operators;
19. Set offspring approximated fitness;
20. Add offspring to species population;
21. end
22. Select representatives for each species;
23. Fabricate, evaluate, and add to species evaluated list, the fittest unevaluated individual in species;
24. Fabricate, evaluate, and add to species evaluated list, a random unevaluated individual in species;
25. end
26. end
27. end

The CGA and SCGA system performances are shown in Fig. 13, which illustrates the combined maximum tip speed achieved by the fittest individuals each half-generation. As can be seen, the performance of the CGA initially increases rapidly from 1523 rpm after 40 fabrications (that is, the initial random population) to 2088 rpm after 80 fabrications (the second generation). This is largely due to the transition from initial random collaboration to pairing with the elite member in each species. Thereafter, the CGA continues to identify increasingly higher aggregated tip speeds: 2158 rpm after 120 fabrications, and 2209 rpm after 160 fabrications (see array designs in Fig. 14). Similar to the previous experiments, the SCGA is used for comparison after the CGA has been run for 2 evolutionary generations since sufficient training data is required to provide useful approximations. Again, similar to the previous experiments, the SCGA identifies improved designs within the same number of fabrications as the CGA, finding a solution yielding 2296 rpm after 140 fabrications and 2429 rpm after 160 fabrications (Fig. 13). Comparing the final 20 treatment combinations from each experiment, the average tip speed of the SCGA ($M = 2205, SD = 47, N = 20$) is significantly greater than the GA-only approach ($M = 1894, SD = 43, N = 20$) using a two-sample t-test assuming unequal variances, $t(33) = 3.7, p \leq .00076$. Furthermore, the fittest treatment combination designed by the SCGA (2429 rpm) was greater than the CGA approach (2209 rpm) after 160 fabrications.

V. CONCLUSIONS

This paper has shown that EAs are capable of identifying novel and increasingly efficient VAWT designs wherein a sample of prototypes are fabricated by a three-dimensional printer and examined for utility in the real-world. The use of a neural network surrogate model was found to reduce the
number of fabrications required by an EA to attain higher aerodynamic efficiency (tip speed) of VAWT prototypes. This approach represents the first surrogate-assisted embodied evolutionary algorithm using three-dimensional printing, and completely avoids the use of three-dimensional computer simulations, with their associated processing costs and modelling assumptions. In this case, three-dimensional CFD analysis was avoided, but the approach is equally applicable to other real-world optimisation problems, for example, those requiring computational structural dynamics or computational electromagnetics simulations. We anticipate that in the future such approaches will yield unusual yet highly efficient designs that exploit characteristics of the environment that are extremely difficult to capture in a simulation. In particular, the wind turbine array experiment has shown that it is possible to use surrogate-assisted coevolution to iteratively increase the performance of two closely positioned turbines, taking into account the inter-turbine flow effects, which is especially difficult to achieve under a high-fidelity simulation. The SCGA represents a scalable approach to the design of wind turbine arrays since the number of inputs to the surrogate-models remains constant regardless of the number of turbines undergoing evolution.

For single turbine comparison, the prototype in Fig. 4 which is a common 4 blade Savonius, measured 827 rpm; a z-rotated variant similar to the design in U.S. patent 7,371,135 (see Fig. 11) measured 650 rpm; and a common 3 blade Savonius design (see Fig. 12) measured 1055 rpm. That is, the prototypes produced here through artificial evolution are aerodynamically more efficient than several common human designs under the current experimental conditions and performance metrics.

Although the array experiment did not elicit counter-rotation as might have been expected, evolution is clearly exploiting characteristics unique to its environment. Duplicating the individual treatments from the fittest array pairing to form homogeneous arrays yields a maximum combined tip speed of 2359 rpm compared with 2428 rpm produced by the heterogeneous array. In addition, while direct comparison cannot be made due to the initial population seeding, the fittest individual from the single turbine z-varying experiment (see Fig. 10(b)) was duplicated to form a homogeneous array and yielded a significantly slower combined tip speed of 2178 rpm compared with the 2428 rpm observed from the array design in Fig. 15.

Future work will include the use of the AC voltage generated by the VAWT prototypes as the fitness computation under various wind tunnel conditions; the coevolution of larger arrays, including the turbine positioning; the exploration of more advanced assisted learning systems to reduce the number of fabrications required; examination of the effect of seeding the population with a given design; investigation of alternative representations that provide more flexible designs including variable number of blades, for example, superquadrics (e.g., [72]); and the production of full-scale designs.

If the recent speed and material advances in rapid-prototyping continues, along with the current advancement of evolutionary design, it will soon be feasible to perform a wide-
array of automated complex engineering optimisation in situ, whether on the micro-scale (e.g., drug design), or the macro-scale (e.g., wind turbine design). That is, instead of using mass manufactured designs, EAs will be used to identify bespoke solutions that are manufactured to compensate and exploit the specific characteristics of the environment in which they are deployed, e.g., local wind conditions, nearby obstacles, and local acoustic and visual requirements for wind turbines.

REFERENCES

[1] J. O. Dahiri, “Potential order-of-magnitude enhancement of wind farm power density via counter-rotating vertical-axis wind turbine arrays,” Journal of Renewable and Sustainable Energy, vol. 3, no. 4, 2011.
[2] S. Eriksson, H. Berntoff, and M. Leijon, “Evaluation of different turbine concepts for wind power,” Renewable and Sustainable Energy Reviews, vol. 12, pp. 1419–1434, 2008.
[3] J. D. Anderson, Computational Fluid Dynamics: The Basics with Applications. McGraw Hill, 1995.
[4] L. Gränning, Y. Jin, and B. Sendhoff, “Individual-based management of meta-models for evolutionary optimization with application to three-dimensional blade optimization,” in Evolutionary Computation in Dynamic and Uncertain Environments, ser. Studies in Computational Intelligence, S. Yang, Y.-S. Ong, and Y. Jin, Eds. Springer, 2007, vol. 51, pp. 225–230.
[5] M. Humpsey, “Multiobjective evolutionary optimisation of small wind turbine blades,” Ph.D. dissertation, University of Newcastle, 2002.
[6] T. J. Carrigan, B. H. Dennis, Z. X. Han, and B. P. Wang, “Aerodynamic shape optimization of a vertical-axis wind turbine using differential evolution,” ISRN Renewable Energy, 2012.
[7] P. Bentley and J. Wakefield, Genetic Algorithms in Engineering Systems (GALESIA). Springer, 1995, ch. The Table: An Illustration of Diversity Using Genetic Algorithms.
[8] J. Lee and P. Hajela, “Parallel genetic algorithms implementation for multidisciplinary rotor blade design,” Journal of Aircraft, vol. 33, no. 5, pp. 962–969, 1996.
[9] Y. S. Ong and A. J. Keane, “Meta-Lamarckian learning in memetic algorithms,” IEEE Transactions on Evolutionary Computation, vol. 8, no. 3, pp. 99–110, April 2004.
[10] J. D. Lohn, G. S. Hornby, and D. S. Linden, “Human-competitive evolved antenna,” Artificial Intelligence for Engineering Design, Analysis and Manufacturing, vol. 22, no. 3, pp. 235–247, August 2008.
[11] I. Rechenberg, “Evolutionstrategie – optimierung technischer systeme in situ und in situ manufacturing for free standing liquid metal microstructures,” Advanced Materials, doi:10.1002/adma.201301400 2013.
[12] D. L. Cohen, J. I. Lipton, L. J. Bonassar, and H. Lipson, “Additive manufacturing for in situ repair of osteochondral defects,” Biofabrication, vol. 2, no. 3, September 2010.
[13] H. Lipson and J. Pollack, “Automatic design and manufacture of robotic lifeforms,” Nature, vol. 406, no. 6799, pp. 974–978, August 2000.
[14] L. Bull, “Model-based evolutionary computing: a neural network and genetic algorithm architecture,” in Proceedings of the IEEE International Conference on Evolutionary Computation. IEEE Computer Society, 1997, pp. 611–616.
[15] H. Watabe and N. Okino, “A study on genetic shape design,” in Proceedings of the 5th International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc., 1993, pp. 445–451.
[16] F. McGuire, “The origins of sculpture: evolutionary 3D design,” IEEE Computer Graphics Applications, vol. 13, no. 1, pp. 9–11, January 1993.
[17] K. Sims, “Evolving 3D morphology and behavior by competition,” Artificial Life, vol. 1, no. 4, pp. 353–372, 1994.
[18] P. Bentley, “Generic evolutionary design of solid objects using a genetic algorithm,” Ph.D. dissertation, University of Huddersfield, 1996.
[19] P. Eggenberger, “Evolving morphologies of simulated 3D organisms based on differential gene expression,” in Proceedings of the Fourth European Conference on Artificial Life. MIT Press, 1997, pp. 205–213.
[20] C. J. Jacob and A. Nazir, “Polyhedron evolver—evolution of 3D shapes with evolvia,” in Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics: Volume VII Information Systems Development II, July 14–18. International Institute of Informatics and Systemics, 2002.
[21] C. J. Jacob and G. Hushak, “Evolutionary and swarm design in science, art, and music,” in The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music. Springer, 2007, pp. 145–166.
[22] P. W. Prusinkiewicz and A. Lindenmayer, The algorithmic beauty of plants. Springer, 1990.
[23] A. Hacioglu and I. Ozkol, “Transonic airfoil design and optimization by using vibrational genetic algorithm,” Aircraft Engineering and Aerospace Technology, vol. 75, no. 4, pp. 350–357, 2003.
[24] D. Quagliarella and A. D. Cioppa, “Genetic algorithms applied to the aerodynamic design of transonic airfoils,” Journal of Aircraft, vol. 32, no. 4, pp. 889–891, 1995.
[25] P. Hajela and J. Lee, “Genetic algorithms in multidisciplinary rotor blade design,” in Proc. 36th AIAA/ASME/ASC/THS/AHS/ASC Structures, Structural Dynamics and Material Conference, 1995, pp. 2187–2197.
[26] P. Fumeaux and J. Pollack, “Evolving truss structures for adaptive physical designs for robots,” Artificial Life, vol. 4, no. 4, pp. 337–357, October 1998.
[27] L. Hao, O. Seaman, S. Mellor, J. Henderson, N. Sewell, and M. Sloan, Innovative Developments in Design and Manufacturing – Advanced Research in Virtual and Rapid Prototyping CRC Press, 2009, ch. Extrusion behavior of chocolate for additive layer manufacturing, pp. 245–250.
[28] J. I. Lipton, D. Arnold, F. Nigl, N. Lopez, D. L. Cohen, N. Noren, and H. Lipson, “Multi-material food printing with complex internal structure suitable for conventional post-processing,” in 21st Solid Freeform Fabrication Symposium. The University of Texas at Austin, 2010.
[29] S. Eriksson, H. Berntoff, and M. Leijon, “Evaluation of different turbine concepts for wind power,” Renewable and Sustainable Energy Reviews, vol. 12, pp. 1419–1434, 2008.
