

# An efficient method for the design and fabrication of 2D laminate robotic structures

Yufeng Chen<sup>1,2</sup>, Aaron Ong<sup>3</sup>, and Robert J. Wood<sup>1,2</sup>

**Abstract**—Microrobots are constructed based on processes involving lamination of 2D materials and laser ablation. To satisfy the size constraint of the laser cutting field, different microrobotic components are tiled within a square template during the design process. Here we develop a number of automated 2D tiling algorithms that aim to improve microrobot fabrication efficiency. We quantify and compare the performance of a deterministic method, a method based on simulated annealing and expectation maximization, and a method based on potential functions and stochastic gradient descent. These methods automatically tile irregular 2D shapes within a template, and they can be applied to improve the fabrication efficiency of microrobotic components. To demonstrate the effectiveness of this method, we design and fabricate 102 different robot wings in the same setting, which represents over five times improvement compared to previous methods. Flapping experiments with these wings are further conducted to identify the corresponding operating conditions.

## I. INTRODUCTION

Due to smaller size and weight, microrobots [1], [2], [3], [4] are better suited for search and rescue and environmental exploration missions in cluttered environments compared to large, traditional robots. The Harvard RoboBee [5], shown in figure 1A, is an 80 mg, flapping wing microrobot. The robot consists of a carbon fiber airframe, two piezoelectric actuators, two composite wings made of carbon fiber and polyester, and transmission mechanisms made of polyimide and composite materials. Traditional manufacturing methods such as CNC machining, molding, or casting cannot be applied to mesoscale fabrication due to their limitations in precision and applicable materials.

To address the challenges in microrobotic fabrication, a novel laminate-based manufacturing technique is developed [6], [7]. Using laser ablation and lamination techniques, this fabrication method builds 3D structures out of 2D laminate materials. In the fabrication process, different patterns are laser cut from different material layers. Next, these material layers are stacked together through pinhole alignments (figure 1B). By applying heat (up to 200 °C) and pressure (up to 70 lb/in<sup>2</sup>)(figure 1C), these material layers are laminated into a single layer. Finally, microrobotic components are released from this laminated layer, and the components either self-fold or are manually folded [8] into 3D functional structures.

1. John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, USA. 2. Wyss Institute for Biologically Inspired Engineering, Harvard University, Cambridge, MA 02138, USA. 3. Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093, USA. Correspondence email: yufengchen@seas.harvard.edu

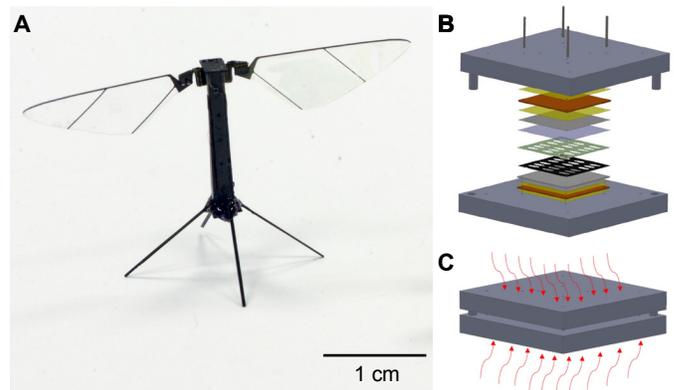


Figure 1. Microrobot fabrication. **A.** The Harvard RoboBee is an 80 mg flapping wing aerial microrobot. All the components of this robot are fabricated based on the 2D lamination process. **B.** The robot wing lamination process. Different material layers are laser machined, and then positioned and stacked using alignment pins. **C.** These layers are laminated together under pressure and heat, and the laminated assembly is released by laser to make functional robotic components.

This fabrication method imposes two major difficulties on the design process: 1) the cut patterns of different material layers need to be individually designed, which is both time consuming and error-prone; 2) the lamination procedure requires multiple robotic components to be fitted into a template of fixed size, and this tiling process requires human effort. Manually tiling a large number of objects is both time consuming and inefficient, and this often leads to a waste of materials. In this paper, we describe a new design tool that resolves these challenges. Given the design parameter values, the algorithm automatically generates the cut files for different material layers. In addition, given a set of objects that need to be fabricated, the algorithm automatically tile these objects within a template. Here we use the RoboBee wing fabrication as an illustrating example to demonstrate the new design method’s effectiveness.

## II. AUTOMATED DESIGN OF A SINGLE WING

A RoboBee wing is made of a carbon fiber structural frame and polyester membrane through the laser ablation and lamination processes. The fabrication requires three materials: 0°-45°-0° carbon fiber laminate, polyester membrane, and acrylic adhesive. The 0°-45°-0° carbon fiber laminate ensures high stiffness along the leading edge and wing spar directions. The acrylic adhesive laminates polyester membrane to the carbon fiber frame. Each material is laser cut using different cut files and laminated together under heat and pressure. Once cured,

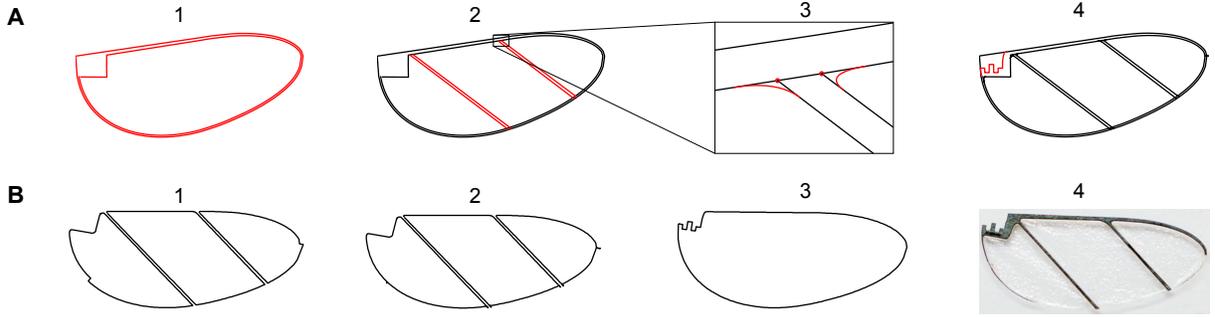


Figure 2. Illustration of wing cut file generation. **A.** Design processes of a single wing. After wing outer contour is generated based on morphological parameter inputs (**A-1**), wing spars are placed between the leading and the trailing edge (**A-2**). Next, stress relieving fillets (**A-3**) and mating feature (**A-4**) are added to the design. **B.** Cut files of the carbon fiber composites (**B-1**), the acrylic adhesive (**B-2**), and the final release (**B-3**). A sample wing designed by this algorithm is shown in (**B-4**).

the composite is laser cut again to release the wings. Here we discuss the design process and describe a design automation method.

Previous wing shape variation studies required manual generation in a professional CAD software. This process was both time consuming and inconsistent, since a human operator was in charge of hand determining fine details such as spar placement and fillet curves radii. We develop an algorithm that generates wing designs based on morphological parameter inputs. As discussed in a previous study [9], the wing planform is completely parametrized by a leading edge function  $y_{LE}(r)$ , wing span  $R$ , first moment  $\hat{r}_1$ , and mean chord length  $\bar{c}$ . Figure 2 illustrates the design process of a single wing. The algorithm computes the wing planform (figure 2A-1) based on user inputs. Next, two diagonal wing spars are placed between wing leading and trailing edges (figure 2A-2). These spars form  $45^\circ$  angle with respect of the wing leading edge to ensure alignment with the  $0^\circ$ - $45^\circ$ - $0^\circ$  carbon fiber laminate. In addition, stress relieving features are placed at wing spar locations (figure 2A-3). Finally, a mating feature is placed near the wing root for wing hinge attachment (figure 2A-4).

The completed design is further decomposed into three different laser cut files. Selected regions of carbon fiber (figure 2B-1) and acrylic adhesive (figure 2B-2) are removed before the lamination step. Compared to the carbon fiber cut pattern, the adhesive cut pattern is offset outward to account for material thermal expansion during the lamination process. The final release cut traces the wing outer contour (figure 2B-3). Figure 2B-4 shows a photograph of a robot wing designed by this algorithm. In the following sections, we describe algorithms that automatically pack many robot wings within the same template for efficient fabrication.

### III. AUTOMATED TILING OF MULTIPLE WINGS

#### A. Problem formulation

Our manufacturing process requires multiple objects to be fitted into a template for batch fabrication. The 2D tiling problem can be formulated under the optimization framework. Given a list of objects  $s_i$ , we aim to choose the subset of objects whose total area is maximized. We further impose two constraints: no two objects overlap and each selected object

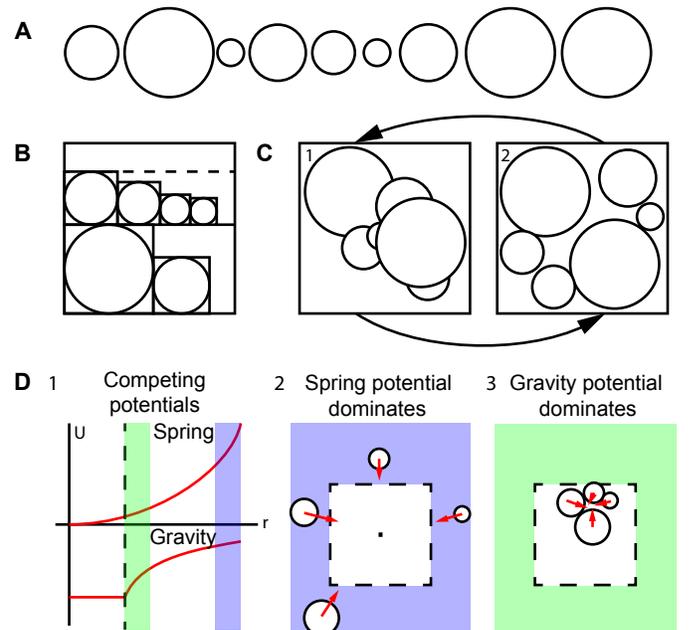


Figure 3. Problem formulation and packing algorithms. All convex shapes are represented by circles in this figure. **A.** Illustration of a list of circles that need to be tiled into a square template. **B.** A deterministic greedy method. The circles are ranked and then tiled into the template in several strips. **C.** A stochastic method based on simulated annealing and expectation maximization. In **C-1**, the method uses simulated annealing to choose the set of items to be packed. In **C-2**, the method packs the items into a template through sampling. **D.** Illustration of a potential function. **D-1** The spring potential and gravitational potential between two objects. The dotted line indicate the position when two objects collide. **D-2** Spring potential dominates when objects are far away from the template and attract the objects toward the template center. **D-3** Gravitational potential dominates when objects are close to each other and are near the template center.

must be placed within the template. Mathematically, this NP hard integer programming problem is described by:

$$\begin{aligned} & \operatorname{argmax} \sum_i z_i A_i \\ & \text{s.t. } s_i \cap s_j = \{\} \forall i, j \\ & \quad s_i \subset B \forall i \end{aligned}$$

We aim to solve for the selection indicator vector  $z$  and centroid position  $x_i$  for every selected object.  $A_i$  is the area of the  $i^{\text{th}}$  object and  $B$  is the bounding template. While

this formulation is similar to the 1D knapsack problem, it is difficult to search for a feasible solution given selection indices. Once the set of chosen items are mutated, the latent variables often need to be re-solved.

Figure 3A illustrates a set of tiled objects using only circles, while in our implementation the tiled shapes also include convex polygons. Limiting the tiled objects to convex polygons and circles makes object overlap detection faster, and it does not limit the algorithms’ usefulness because an arbitrary shape can be bounded by a convex polygon and then tiled by the algorithms. In addition, the latent variables only involve the centroid location  $(x_i, y_i)$  and we do not allow planar rotation. This assumption is justified because laminate materials such as carbon fiber composites are often directional. In the following section, we describe algorithms for solving this packing problem.

### B. Algorithm design

1) *First Fit Decreasing Height*: We implement a deterministic strip packing algorithm for baseline comparison. There is a class of strip packing algorithms discussed in [10], [11], [12], [13], [14]. This type of algorithms first find a rectangular bounding box, then rank the object in either height, width or height to width ratio. The objects are later put into the bounding area B according to the sorted sequence. These methods have complexity of  $O(n \log n)$ , and they give a greedy approximation of the solution. Figure 3B illustrates an example based on a deterministic method.

2) *Simulated annealing*: It is difficult to simultaneously solve for the labels and the latent variables through simulated annealing. Here we design a two-step process inspired by expectation maximization. The EM algorithm uncouples the problem into two iterative steps, where the first step solves for the free parameters and the second step solves for the labels. Due to the special concave property of the log-likelihood function, the EM algorithm is guaranteed to monotonically converge to a local minima. Our approach is inspired by the EM algorithm but does not have the monotonicity property, and consequently there is no likelihood or objective function that is strictly convex. It is similar to the EM method because this iterative approach uncouples the large search space of latent variables  $(x_i, y_i)$  and labels  $z_i$ .

Simulated annealing is a very popular method for discrete optimization problems where it is difficult to derive analytical gradients. Here choosing a list of items is a discrete problem. Since we set the objective function value to be the net covered area, we can implement simulated annealing and use this value as the criteria for rejecting or accepting the proposal. Figure 3C-1 and 3C-2 illustrate this process of solving for the labels  $z_i$  and the latent variables  $(x_i, y_i)$  iteratively.

3) *Stochastic gradient descent*: It maybe inefficient to pack the objects using simulated annealing because gaps exist between neighboring objects. Consequently, it is important to design potential functions that remove unnecessary white space.

Our algorithm defines a potential function and searches for a local minimum. This potential function is defined as the sum of a global quadratic potential and local gravitational ones:

$$U = \frac{1}{2}K \sum_i x_i^T x_i - G \sum_i \sum_{j \neq i} A_j \frac{1}{\|x_j - x_i\|},$$

where K and G are the spring and gravitational coefficients. The solver starts by randomly placing each item outside of the template and then iteratively minimize the objective function.

In every iteration each object is displaced sequentially in the direction of the local gradient:

$$D_i \propto F_i = -\nabla_i U = Kx_i + G \sum_{j \neq i} A_j \frac{x_j - x_i}{\|x_j - x_i\|^3}.$$

As shown in figure 3D-1, the spring and gravitational potentials dominate in different regimes. The spring potential gives a “long range” force because it has a large gradient when objects are distant to the template center. In contrast, the gravitational potential gives a “short range” force because its gradient increases as two objects move close. A distant object is pulled toward the template by a long range spring force (figure 3D-2). Objects within the template pack densely due to local gravitational attraction (figure 3D-3). Finally, every object performs a biased random walk after a fixed number of iterations. This design allows objects to escape from local minima. This potential function is computationally expensive because all objects interact with every other ones. Instead of accounting for every pairwise interaction, we approximate the function gradient by randomly sampling five neighbor objects. At the start of the algorithm, we randomly place each item far away from the template and allow the spring potential to attract them toward the center.

### C. Comparison of algorithm performance

To evaluate the performance of each packing algorithm, we design a test case consisting of 30 randomly generated objects. Each algorithm solves the same packing problem with the goal of maximizing the total covered area given a fixed template size. Figure 4 compares the performance of each method, and further identifies the corresponding strengths and weaknesses.

1) *Greedy deterministic method result*: We find that the deterministic greedy algorithm is very efficient when tiling a collection of regularly sized items. Figure 4A shows the tiling result of this greedy algorithm. The algorithm packs 28 out of 30 items within 0.1s. However, as we introduce one additional item that is significantly taller than the others, the algorithm performance deteriorates quickly. In figure 4B, the greedy algorithm only tiles 11 items and suffers 22% reduction of covered area. This example shows that the deterministic method is sensitive to outlier shapes and dimensions.

2) *Simulated annealing result*: We find that simulated annealing is inefficient and returns a mediocre solution. Figure 4C shows the tiled pattern with a total area of 154 and run time of 21 minutes. While simulated annealing performs well at shuffling the selection indices, it takes a long time towards finding a feasible solution. The method does not have spacial information about relative item placement. Consequently, it can only place down a new item by sampling the remaining empty space. As the net covered area increases, the method of

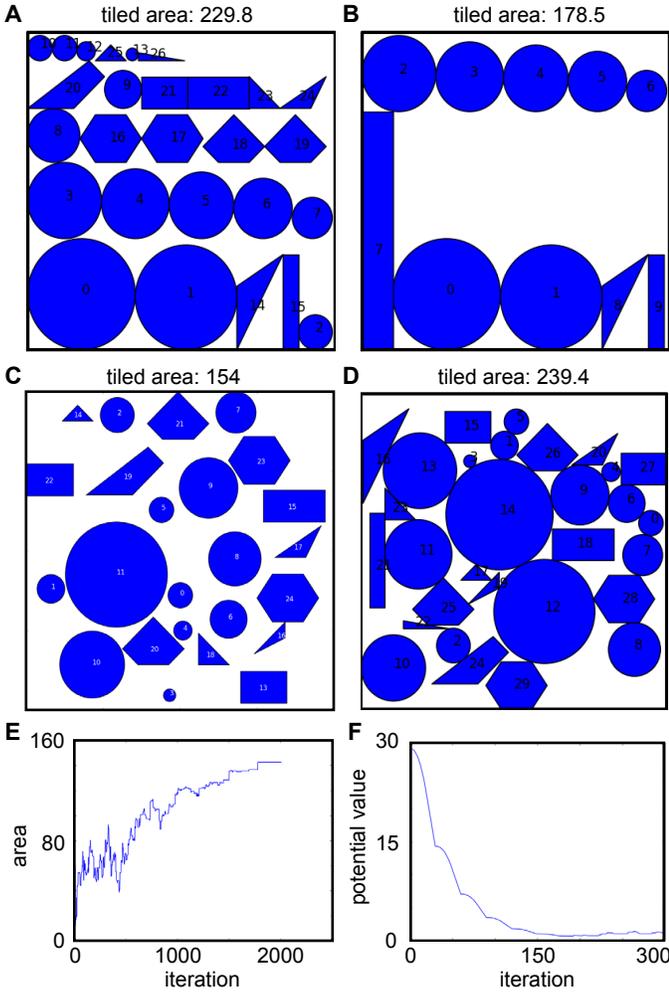


Figure 4. Comparison between different packing methods. **A.** Tiled pattern of the greedy algorithm for a set of randomly chosen items. The method tiles 28/30 items. **B.** Tiled pattern with one added irregular item. Here only 11/31 items are tiled. **C.** Tiled pattern of the simulated annealing method. The method tiles 24/30 items. **D.** Tiled pattern of the stochastic gradient descent method. The method tiles all 30 items. **E.** Tiled area as a function of iteration for the simulated annealing method shown in **C.** **F.** Potential function value as a function of iteration for the stochastic gradient descent method shown in **D.**

random placement through sampling suffers from having high rejection rate. Figure 4E shows the net covered area increases slowly as a function of algorithm iterations. In addition, there is usually a large amount of empty space between neighboring items, and the uncovered area is characterized by the length scale of the smaller items. These shortcomings make simulated annealing an unpromising packing method.

3) *Stochastic gradient descent result:* We find the stochastic gradient descent method to be both efficient and robust. Figure 4D shows the tiled pattern and here the method fits all the given items within two minutes. Figure 4F shows the reduction of the potential function value as the algorithm runs. The small kink in figure 4F is caused by the random walk after every 30 iterations, and this random walk is important for escaping local minima. We find that the items are tightly packed due to the gravitational attraction.

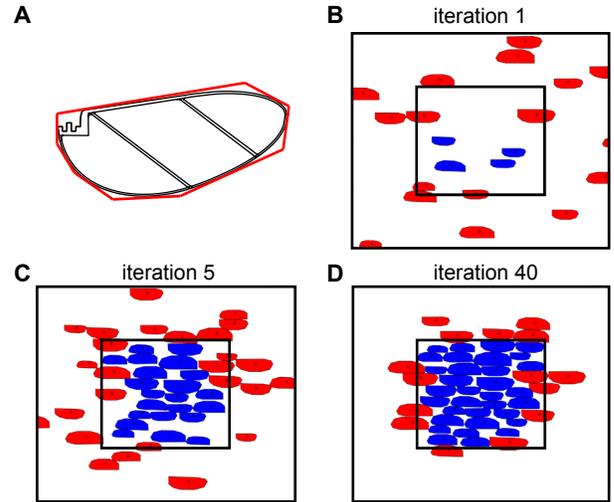


Figure 5. Demonstration of automated wing design and tiling. **A.** A wing design is automatically generated and it is bounded by a convex polygon. **B-D.** Initialization, iteration 5, and iteration 40 of the wing tiling process. In **B-D**, the inner black square indicates the template size, the blue colored polygons represent wings that are tiled within the template, and the red colored polygons represent wings that have not been fitted into the template.

#### D. Demonstration of wing tiling

As discussed in the introduction section, the design of RoboBee wings is crucial for flight efficiency and vehicle payload. The robot actuation and transmission are highly nonlinear, and existing quasi-steady models cannot predict the lift and drag forces associated with a particular wing design. Consequently, characterizing the wing performance requires extensive experimental efforts [9], [15]. A number of wing parameters can be varied during experimental evaluations. For instance, increasing the wing area reduces the system resonance frequency; increasing the wing area moment or reducing the wing aspect ratio increases the load on the wing tip; changing the wing leading edge profile affects the wing's passive pitch rotation. Meanwhile, varying these parameters changes the wing shape, which requires modification in the design cut files. The method discussed in section II can automatically generate the cut files for a single wing given modified morphological parameters. Here we apply the packing method based on stochastic gradient descent to improve the fabrication efficiency of a large number of different wings.

Figure 5 illustrates this automated tiling process. Given an auto-generated wing profile, this method first simplifies the wing geometry by computing a convex bounding box (figure 5A). Next, the algorithm initializes and assigns each wing with a random centroid position (figure 5B). For visualization purpose, a wing is colored blue if it is completely contained within a 45 mm  $\times$  45 mm square template. A wing is colored red if any part of the wing lies outside of the template. As the algorithm proceeds, the spring potential dominates and pulls more wings into the template (figure 5C). Once the wings are pulled into the template, the gravitational potential dominates and it densely pack the neighboring wings. If several wings form an unfavorable configuration, then a random walk breaks this configuration, leading to an improved arrangement. As

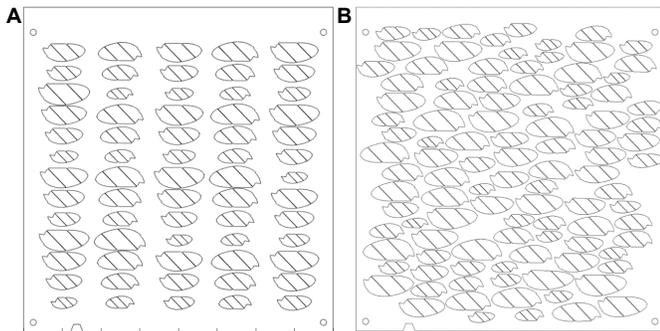


Figure 6. Comparison between a deterministic tiling method and the stochastic tiling method. **A.** The deterministic method tiles 65 wings in one template. **B.** The stochastic method fits 102 wings into the same template, representing 57% improvement.

shown in figure 5D, the white space between the packed wings substantially reduces after the 40<sup>th</sup> iteration. In this example, the algorithm tiles 33 wings in a 45 mm  $\times$  45 mm template within 2 minutes of run time. Once the tiling algorithm terminates, it rejects all the red colored wings and outputs cut files only containing the blue colored wings. The rejected wings are tiled in another empty template for ensuing batch fabrication.

#### IV. RESULTS AND DISCUSSION

The demonstration shown in the previous section illustrates the algorithm convergence process for a medium sized problem. In this section, we apply the single wing auto-generation method and the auto-tiling algorithm toward fabricating over 100 wings in a single batch. In our fabrication setup, the laser machining system has a maximum cutting field of 85 mm  $\times$  85 mm, and consequently we decide to demonstrate efficient wing fabrication using this large template.

Figure 6 compares the performance of a hand-tuned, deterministic tiling method and the method based on the potential function and stochastic gradient descent. The deterministic method bounds each wing using a rectangle and tiles these rectangles in the template (figure 6A). This method fits a total of 65 wings and it shows two shortcomings: 1) large wings and small wings occupy the same area inside the template, and 2) there remains unfilled space between neighbor wings. Both problems are mitigated by the stochastic tiling method. As shown in figure 6B, the stochastic tiling method reduces unnecessary space between neighboring wings, and wings of different area are densely packed together. The stochastic tiling method fits 102 wings in the same template, representing 57% improvement.

Without any manual modification, we use the auto-generated cut files to fabricate the wings. First, the carbon fiber layer, the acrylic adhesive layer, and the polyester layer are separately laser machined and then laminated together. Figure 7A shows the laminated material prior to the laser release cut. Before releasing the wings from this template, we label the design parameters on each wing root. The numbers shown in figure 7B represent the wing aspect ratio, wing span, first area moment, and leading edge sweep ratio, respectively. Labeling each wing

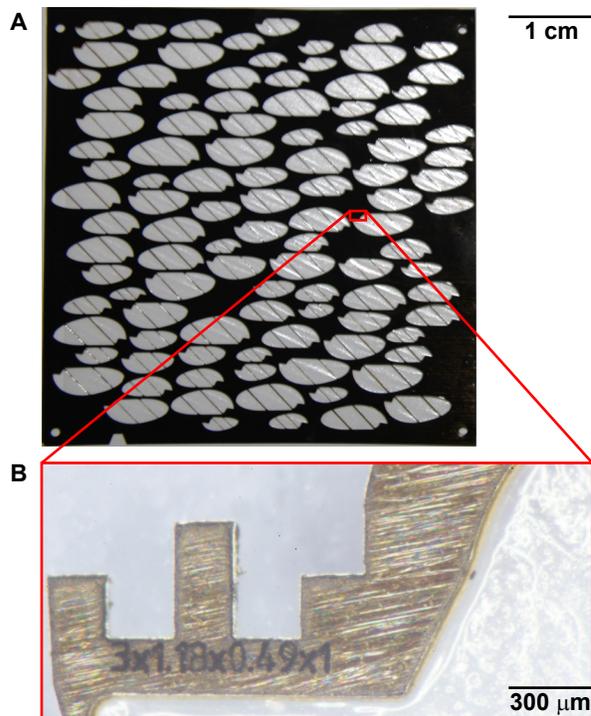


Figure 7. Illustration of robot wing fabrication. **A.** Fabrication of 102 different wings within one template. **B.** Wing design parameters are labeled on the wing root.

facilitates the following wing characterization experiments. Finally, the wings are released from the template through a laser release cut. This entire fabrication process takes less than a working day to complete, and it requires approximately four hours of human supervision.

Traditionally, wing design and cut file generation are more time intensive compared to the fabrication process. Manually designing the cut files of a single wing requires at least an hour, and manually tiling over 100 wings is both tedious and inefficient. Consequently, the entire design process can take up to a working week whereas the fabrication procedure takes less than a working day. By developing these design and tiling methods, we completely automate the wing design process. By only specifying the wing morphological parameters, we leave all the design details to the algorithm, which takes under 20 minutes to compute and generate all the required laser cut files. Hence, this development represents a substantial improvement of efficiency (over five times in terms of the total design and fabrication time).

Finally, we demonstrate flapping wing motion using a sample wing chosen from the batch. The area, aspect ratio, and area moment  $\hat{r}_1$  of this wing are 27 mm<sup>2</sup>, 3, and 0.49, respectively. The wing is mounted onto a wing hinge whose stiffness is 2.4  $\mu Nm/rad$ , and the assembly is installed on a wing driver (figure 8A). Using the method described in a previous study [15], 77 flapping experiments are conducted to identify the wing operating conditions. In these experiments, the input voltage and flapping frequency vary in the range of 100 - 200 V, and 170 - 230 Hz, in steps of 10 V and 10 Hz, respectively. For the wing under test, the optimal operating

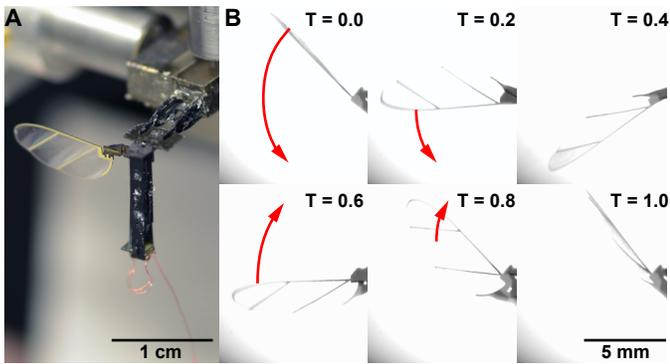


Figure 8. Demonstration of a flapping experiment using a wing from the batch fabrication. **A.** The wing is installed on a wing driver. **B.** Images taken from a high speed video of the flapping experiment. The wing flapping amplitude is maximized at 220 Hz. The labeled time in B is normalized to a flapping period.

condition is found to be 200 V and 220 Hz, with a maximum stroke amplitude of  $82^\circ$ . Figure 8B shows an image sequence of the flapping wing motion at the optimal operating condition. In future studies, these experiments can be repeated for other wings to compare wing performance. This fabrication method allows efficient evaluation of different wing designs, and this shows promise to improve the aerodynamic efficiency of future flapping wing vehicles.

## V. CONCLUSION AND FUTURE WORK

In this work, we have developed several 2D tiling algorithms that aim to improve the microrobot fabrication efficiency. We quantify and compare the algorithms' performance and show that a method using stochastic gradient descent to optimize a custom-designed potential function is the most robust and efficient. We further demonstrate the effectiveness of this algorithm by designing, tiling, and fabricating 102 different wings in the same template, which represents 57% improvement over a deterministic method. Compared to the existing wing fabrication process, this auto-design and auto-tiling approach substantially reduces the total design and fabrication time by over five times. This improvement of fabrication efficiency paves the way for comprehensive experimental studies of wing performance.

While this work uses wing fabrication as a motivating example, this automation approach can be widely applied to the design of other microrobotic components. For instance, the RoboBee transmission and wing hinge consist of compliant polyimide flexures and rigid carbon fiber linkages. Modifying the dimensions of the flexures and linkages change the transmission ratio and stiffness, respectively. These parameters have a large impact on system resonance and consequently influence the robot flight efficiency and payload. We can use our algorithm to facilitate the design of these components, thus preparing for further experimental optimization of robot performance. Finally, the RoboBee is an example of the many microrobots [17], [18] and microrobotic devices [19], [20] that are fabricated through the 2D lamination process. The design and tiling algorithms presented in this paper can be applied in the fabrication process of similar microrobotic systems.

## VI. ACKNOWLEDGMENT

We thank Nicolas Gravish for comments and discussions. This work is supported by the Wyss Institute for Biologically Inspired Engineering. This material is based upon work supported by the NSF (award no. 1537715). In addition, the prototypes were enabled by equipment supported by the Army Research Office DURIP program (award no. W911NF-13-1-0311). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

## REFERENCES

- [1] Guo, Shuxiang, Toshio Fukuda, and Kinji Asaka. "A new type of fish-like underwater microrobot." *Ieee/Asme Transactions on Mechatronics* 8.1 (2003): 136-141.
- [2] Donald, Bruce R., et al. "An untethered, electrostatic, globally controllable MEMS micro-robot." *Journal of microelectromechanical systems* 15.1 (2006): 1-15.
- [3] Pawashe, Chytra, Steven Floyd, and Metin Sitti. "Modeling and experimental characterization of an untethered magnetic micro-robot." *The International Journal of Robotics Research* 28.8 (2009): 1077-1094.
- [4] Kim, Byungkyu, et al. "An earthworm-like micro robot using shape memory alloy actuator." *Sensors and Actuators A: Physical* 125.2 (2006): 429-437.
- [5] K. Ma, P. Chirarattanon, S. Fuller, and R.J. Wood, "Controlled Flight of a Biologically Inspired, Insect-Scale Robot", *Science*, vol. 340, pp. 603-607, 2013.
- [6] Wood, Robert J., et al. "Microrobot design using fiber reinforced composites." *Journal of Mechanical Design* 130.5 (2008): 052304.
- [7] Sreetharan, Pratheev S., et al. "Monolithic fabrication of millimeter-scale machines." *Journal of Micromechanics and Microengineering* 22.5 (2012): 055027.
- [8] Whitney, John P., et al. "Pop-up book MEMS." *Journal of Micromechanics and Microengineering* 21.11 (2011): 115021.
- [9] Chen, Yufeng, Kevin Ma, and Robert J. Wood. "Influence of wing morphological and inertial parameters on flapping flight performance." *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016.*
- [10] Hopper, Eva, and Brian CH Turton. "A review of the application of meta-heuristic algorithms to 2D strip packing problems." *Artificial Intelligence Review* 16.4 (2001): 257-300.
- [11] Hopper, E., and B. Turton. "A genetic algorithm for a 2D industrial packing problem." *Computers & Industrial Engineering* 37.1-2 (1999): 375-378.
- [12] Jia, X., and R. A. Williams. "A packing algorithm for particles of arbitrary shapes." *Powder technology* 120.3 (2001): 175-186.
- [13] Bortfeldt, Andreas. "A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces." *European Journal of Operational Research* 172.3 (2006): 814-837.
- [14] Deterministic methods for 2D tiling: <http://cgi.csc.liv.ac.uk/~epa/surveyhtml.html>.
- [15] Chen, Yufeng, et al. "Experimental and computational studies of the aerodynamic performance of a flapping and passively rotating insect wing." *Journal of Fluid Mechanics* 791 (2016): 1-33.
- [16] Chen, Yufeng, Alexis Lussier Desbiens, and Robert J. Wood. "A computational tool to improve flapping efficiency of robotic insects." *Robotics and Automation (ICRA), 2014 IEEE International Conference on. IEEE, 2014.*
- [17] Baisch, Andrew T., and Robert J. Wood. "Design and fabrication of the Harvard ambulatory micro-robot." *Robotics Research. Springer Berlin Heidelberg, 2011. 715-730.*
- [18] Donald, Bruce R., et al. "An untethered, electrostatic, globally controllable MEMS micro-robot." *Journal of microelectromechanical systems* 15.1 (2006): 1-15.
- [19] Wang, Yueping, et al. "A biorobotic adhesive disc for underwater hitchhiking inspired by the remora suckerfish." *Science Robotics* 2.10 (2017): eaan8072.
- [20] Gafford, Joshua B., et al. "Force-sensing surgical grasper enabled by pop-up book MEMS." *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013.*