AM 207 final project proposal

Yufeng Chen, Jeewon Hwang, Zelong Qiu

April 7, 2016

We have came up with 3 possible final project topics and we think topic 1 (2D tiling) is most interesting to us. Specifically, it allows us to explore a variety of optimization methods towards addressing a research related application.

1 2D tiling

Background

This problem is motivated by applications in laser machining developed by the Harvard Microrobotics Lab. In the field of meso-scale fabrication it is advantageous to construct 3D structures out of 2D laminate materials. Composite materials such as carbon fiber are often anisotropic which makes traditional manufacturing unfeasible. Consequently, we cut out 2D features and patterns from planar templates using diode pulsed laser. An illustration of this manufacturing process is shown in figure 1. The material templates have fixed size to facilitate the lamination process, which implies that multiple manufactured parts must be tilted onto a common template. Usually this tiling process is done by a human, which can be time consuming and inefficient. Here we aim to improve tiling efficiency and speed by using stochastic optimization techniques discussed in AM207.

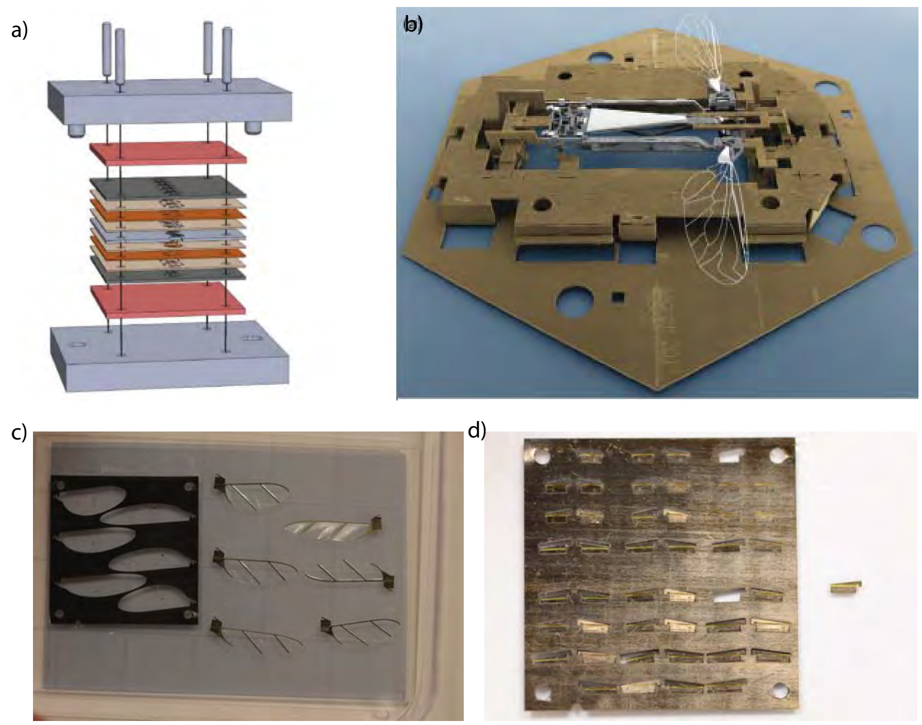

Figure 1: Illustration of laser machining process. a) The lamination process. b) A micro-robot fabricated from the process. c) Example of manually tiled wings. d) Example of manually tiled wing hinges.

Problem formulation

This optimization problem can be formulated as:

$$\begin{gathered} argmax \sum_i z_i A_i \\ \text{s.t } S_i \cap S_j = \{\} \forall i, j \\ S_i \subset B \ \forall i \end{gathered}$$

Here we want to find the best indicator vector z_i and centroid position (x_i, y_i) for each cut feature such that we maximize the sum feature area. The constraint is that no two region can overlap and that each feature is entirely contained within the template. This is an integer programming problem and it is NP hard. Here we aim to find an approximate solution using stochastic methods. Here $z_i = \{0, 1\}$ which is the selection vector. $A_i \in R$ is the area of feature i and S_i is the set covered by feature i. Finally, B denotes the set covered by the bounding box.

This problem is derived from a real world application and is a generalization of the 1D knapsack problem we've done in homework 3. Specifically, we need to choose a set of items to maximize an objective function subject to constraints.

However, this is more difficult because it involves "latent variables" that specify the placement of items. We let (x_i, y_i) be the latent variables that specify the location of the items.

Here we make a simplifying assumption that we may come back to if time permits. Specifically, we assume all features are circles of different radius r_i. This assumption makes object overlap detection much faster. In addition, the latent variables only involve the centroid location (x_i, y_i). If time permits, we will generalize to non-circle objects and introduce the orientation latent variable θ_i. We expect this generalization to add additional computational cost to the optimization problem. In this proposal, we illustrate our proposed ideas using only circles.

Proposed algorithm

In AM207 we discuss 4 stochastic optimization algorithms: simulated annealing, genetic algorithms, stochastic gradient descent, and expectation maximization. We think this problem is an appropriate final project because it allows us to explore all 4 methods. First we introduce a deterministic method, then we discuss our ideas on the use of all 4 methods towards solving this problem below.

- Strip packing

For a base line comparison, we can implement a deterministic algorithm called strip packing. This is a greedy algorithm that can be used for base line comparison. There are a class of strip packing algorithms discussed in http://cgi.csc.liv.ac.uk/~epa/surveyhtml.html. 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(nlogn), which gives a greedy approximation of the solution. We plan to implement several deterministic algorithms for base line comparison. Figure 2b illustrates the deterministic methods.

Two-step approach

Here we propose to tackle this problem using a method similar to expectation maximization. Here we want to uncouple the optimization step into 2 iterative steps. First, we use simulated annealing or genetic algorithms to select the set of items that will be included in the bounding box. Next, we formulate a potential field to solve for the most packed configuration. Figure 2c illustrates our proposed method. The overall proposed method will make use of all 4 optimization algorithms discussed in class. We explain the details below.

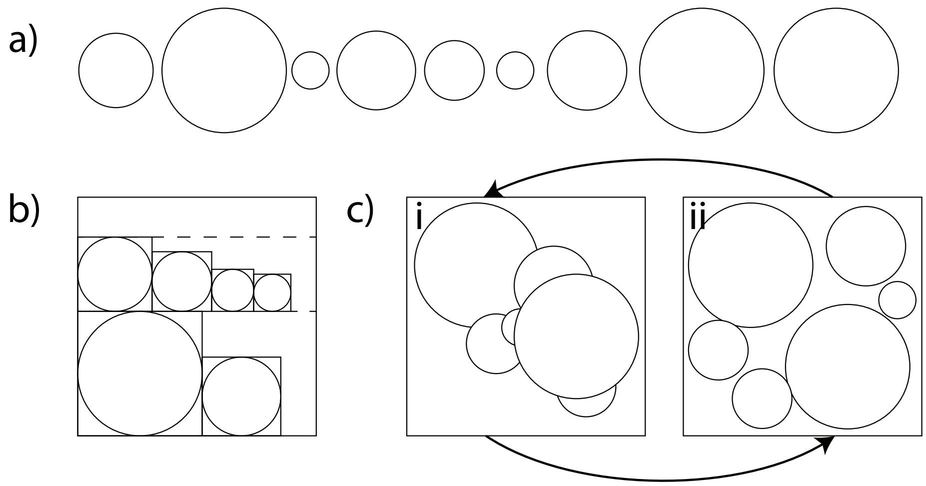

Figure 2: Deterministic vs stochastic methods. a) The list of circles we want to tile into a template. b) Deterministic greedy method. The circles are ranked and then tiled into the template in several strips. c) Proposed stochastic methods. The methods involve 2 steps. c-i) involves simulated annealing or genetic algorithms to choose the set of items. c-ii) involves stochastic gradient descent to tile the items in a template given the set of chosen items.

- Expectation maximization

Expectation maximization methods are ideal for problems with latent variables such as clustering. The EM algorithm uncouples the problem into 2 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. Here our approach is inspired by the EM algorithm but does not have the monotonicity property. Currently we cannot write down a likelihood or objective function that is convex or concave. However, we think this iterative approach makes intuitive sense because it naturally uncouple the large search space of latent variables (x_i, y_i) and labels z_i.

- Simulated annealing

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 define an objective function, we can implement simulated annealing and use the function value as criteria for rejecting or accepting the proposal.

- Genetic algorithm

Genetic algorithm is another stochastic optimization method we've discussed in lecture. Here we can start from a number of arrangement and configuration.

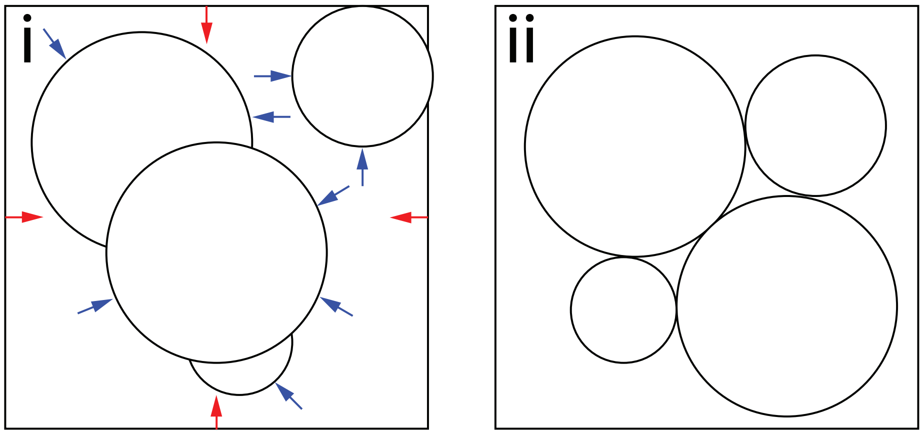

Figure 3: Illustration of using potential functions to tile the template. i) initial condition. The red arrows represent repulsive forces from the bounding box edge. The blue arrows represent attractive forces among the items. ii) Final configuration. This represents local minimum of the potential function.

Next, we combine the best scoring sets and produce descending generations. We repeat this process to converge to an approximate solution. We will compare the effectiveness of the simulated annealing and genetic algorithm.

- Stochastic gradient descent

We think finding a feasible tile configuration is a very challenging optimization task. Given a set of items, we aim to tile them into a template respecting the constraints. While stochastic methods such as simulated annealing can address this problem, it is inefficient because it does not consider the gradient functions. Here we can define a smooth potential function and take advantage of analytical gradients. Intuitively we want the items to touch others' edges to minimize wasted space. Here we can let the bounding box boundaries be repulsive sources and each item be attractive sources. This potential function is easy to evaluate but suffers from many local extrema. In addition, if we have a large number of items then evaluating gradient may become expensive. Consequently, stochastic gradient only samples a subset of selected items and maybe a promising algorithm. Figure 3 illustrates our proposed tiling approach.

2 Social network

Background

Here we can model a social network using a graph $G(V, E)$, where V is a set of nodes that denote the individuals consist of the network and E is a set of connections that denote the relationship between the individuals. Suppose we have a limited number of rewards that we want to distribute to a selected group such that we want to influence the most number of people. Here we can set up this problem in a probabilistic manner, which means we can split a reward between several people. Consequently, we aim to maximize the expected number of people under influence given a constraint on the number of samples we have. This model is usually used to study marketing problems.

Mathematically, the optimization problem can be formulated as:

$$\begin{gathered} argmax \sum_{i=1}^{n} 1 - \Pi_{j:(j,i)\in E}(1 - x_j) \\ \text{s.t. } \sum_{j=1}^{n} x_j \leq K, x_j \in [0, 1] \end{gathered}$$

This problem is a non-convex optimization but can be approximated using a convex upper bond. Usually an approximate solution is obtained through interior point methods. Here we can use stochastic methods discussed in lectures to tackle this problem.

Proposed algorithm

- Simulated annealing

Since we define an objective function, we can implement simulated annealing and use the function value as criteria for rejecting or accepting the proposal. In each iteration we can modify the distribution of x_j under the constraint K. We can later compare the algorithm effectiveness to deterministic ones.

- Genetic algorithm

We can start with many different distributions and then merge the promising ones. This algorithm is usually good for objective functions with many local extrema. We can compare the effectiveness of this algorithm to others.

3 Image classification

Background

Here we can identify features from an image and classify images using the expectation maximization methods discussed in lectures. The expectation step involves identifying image features and the maximization step involves classifying the images accordingly to the identified features. We can compare this method to other machine learning algorithms such as orthogonal matching pursuit.

Proposed algorithm

- Expectation maximization