Fast and Simple Model
For Free Hanging, Pre-impregnated Carbon Fibre Material

Ole W. Nielsen, Christian Schlette and Henrik G. Petersen
The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Campusvej 55, 5230 Odense M, Denmark
{own, chsch, hgp}@mmmi.sdu.dk

Keywords: Modeling Shapes of Deformable Objects, Robotic Layup of Fiber Plies.

Abstract: Automated one-of-a-kind grasping and draping of pre-impregnated (prepreg) fiber plies onto complex molds is a hitherto unsolved problem. In the project FlexDraper, we have gathered an international consortium with industrial keyplayers for addressing this gap. Our approach is based on model- and simulation-based control of the shaping of the plies during the draping procedure, which requires fast computable models in the draping control loop. In this paper, we present such a fast computable mathematical model for the shape of free hanging prepreg fiber plies with perpendicular fiber directions, where the ply is held (clamped) at a set of locations that are not too far apart. Rather than using expensive physically based models, e.g. FEM simulations, our model is only based on interpolations between the held locations satisfying curve length constraints in fiber directions. The paper also contains experimental tests for the accuracy of the model.

1 INTRODUCTION

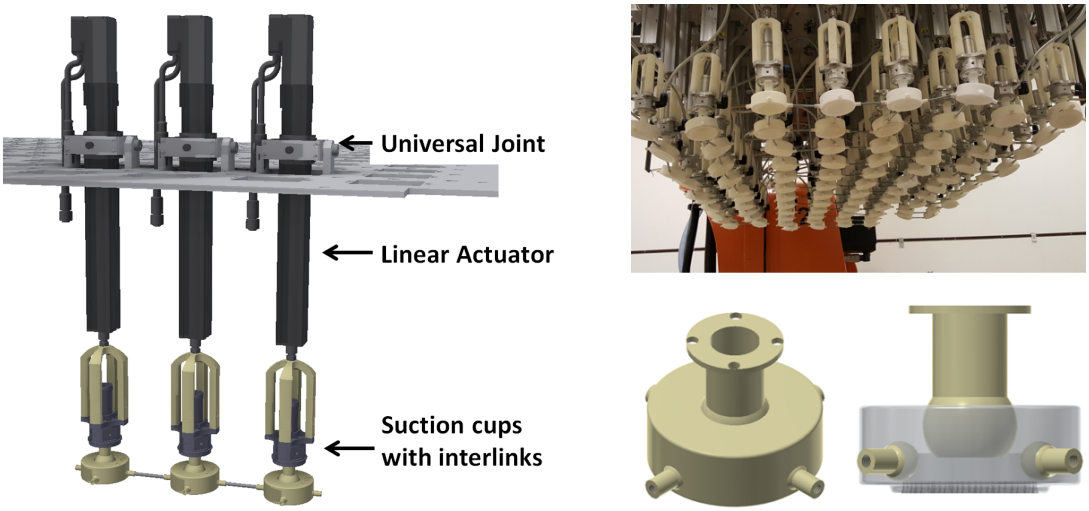
At SDU Robotics, we aim for transferring our examples from research to actual applications by building up close cooperations with industrial partners, such as Terma A/S, a Danish manufacturer of aerostructures from carbon fiber. Despite many attempts to automate it, draping of weaved pre-impregnated (prepreg) carbon fiber plies is a process that is - at least for low volume production - still performed manually. Automation of these processes would lead to huge economic benefits in many application domains, such as aerospace, automotive and offshore energy. To make such a system feasible, it has to be able to handle a large variety of plies as well as parts. Theoretically, with a proper draping tool, a trained operator will be able to teach/program a robot to layup the plies for a single part through experimentation. However, as we require a large variety of parts as well as plies, and that the quantity of each part is low, the time required for an operator working on the robot will lead to an infeasible solution with respect to ROI. Hence, we need to be able to deploy model-based automatic control of the draping process.

In the project FlexDraper (Ellekilde et al., 2018), we have gathered an international consortium for developing a new drape tool mounted on a conventional 6-axis robot. The drape tool (see Figure 1) consists of an array of suction cups where the up-down movement of each suction cup is individually controllable. At the fixed platform, a universal joint is placed and there is spherical joint at the cup (see Figure 2(a)). The nearest neighbour cups are connected by rigid interlinks with spherical joints at the end (see Figure 2(b)). This allow us to control the shape of fiber plies so that we can shape them to the form of the mould.

Figure 1: The FlexDraper robot.

However, one of the main issues with developing model-based control is that the shape of the materials between the suction cups needs to be accurately estimated to enable a draping "from one side" so that undesired air bubbles are avoided. Our approach to model-based control is thus deploying an accurate and

Schlette C., Nielsen O. and Petersen H.
Fast and Simple Model - For Free Hanging, Pre-impregnated Carbon Fibre Material.
DOI: 10.5220/0007239700000000
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018)*, pages 7-14
ISBN: 978-989-758-321-6



(a) Subset of tool showing the relative placement of the universal joints, actuators and suction cups.

(b) Close-up of suction cups with ball joints for connecting with the actuator and interlink structure.

Figure 2: Actuation and suction cup design.

fast computable model for simulating the draping process. The model must be fast computable because the simulations are used to annotate the outcome of a studied draping strategy. These annotations are used in a simulation-based learning procedure. Hence, many simulations must be carried out for fine-tuning the parameters of each draping process. One of the key elements is a fast computable model for predicting the shape of the free hanging parts of the material (i.e. between the suction cups and before touching the mold). It is the main purpose of this paper to present our solution to this and some initial experimental verifications of its accuracy.

The paper is organized as follows: In Section 2, we outline the properties of the FlexDraper tool for draping and discuss the properties of the fiber ply material and in Section 3, we discuss the organization of the draping process. These discussions include the desired properties of our model. Related modelling work is discussed in Section 4 and in Section 5, we present the model, which is the core contribution of this paper. In Section 6, we discuss preliminary validation results and Section 7 concludes the paper.

2 THE FlexDraper TOOL AND MATERIAL

In the FlexDraper project, a drape tool to handle the material has been created. The tool consists of 48 suction cups arranged in a grid like structure, where all the suction cups are connected by interlinks to the nearest neighbours (see Figure 2(b)). The suctions cups are spaced in a loose grid roughly 11 cm apart in both, x and y directions. This gives rise to many small square segments or elements of free hanging material with sizes around $7\times7\text{cm}^2$ between four suction cups and strips of free-hanging material of sizes around $4 \times 7\text{cm}^2$ between two nearest neighbor cups (see Figure 4). It should be mentioned that although the tool is constrained by the interlink structure, it has many passive degrees of freedom that allows the suction cups to align with the mold.

The material is as mentioned a bi-directional carbon fiber weave, pre-impregnated with epoxy resin. It is virtually in-extensible in the perpendicular fiber directions but has a small slack in the weave. Furthermore, the pre-impregnation of the material means it is ”sticky/tacky” in nature and relatively stiff with respect to bending curvature and furthermore dynamic responses (vibrations) are very small.
We therefore make the following assumptions about the material in our model

- Since the material is relatively stiff with respect to bending and because the distance between neighbor suction cups is small, we can neglect gravity. For the same reasons, we can assume that the material is continuous differentiable everywhere
- Since the material is very damped, we can use an equilibrium model
- We assume that the material is unstretchable in the fiber directions

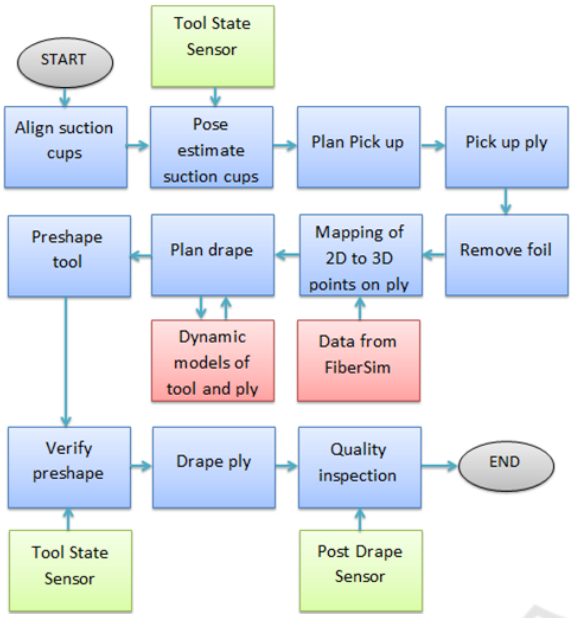


Figure 3: The process: Components in the FlexDraper System (Blue = flow of actions, green = sensor systems, red = data and computational models).

3 THE PROCESS AND MODEL REQUIREMENTS

The process flow is shown in Figure 3. The plies are picked with suction cups from a flat surface. We assume for simplicity of the model description that the suction cup array aligns with the fibre directions. From the tool state sensor, we have accurate predictions of the positions of all suction cups at pick up. After pick up, the suction cups are moved to shape the ply into what we call a preshape. With a preshaped model, each suction cup has a position and orientation which we can again accurately predict using the tool state sensor.

The plies are draped onto the mold by choosing a suitable starting point (choice is depending of the shape of the mold). Then the first suction cup with the material attached is placed on the mold. Subsequently, the other suction cups are placed in a wave like pattern running from the starting point to the boundary. To avoid undesired air bubbles, it is very important that also the material is attaching to the mold sequentially from the starting point and outwards. To ensure this, we have at any point in time the following available information

- The three dimensional shape of the mold surface
- The configuration of the robot
- The configuration of all the linear actuators controlling the up-down movements of the suction cups

We then have an dynamics based forward kinematics model of the drape tool that accurately predicts the position and orientation of all the suction cups after the relatively small movements from the sensor measured preshape. To finalize the simulation model used for learning, we then just need to be able to predict the shape of the free-hanging ply material between the suction cups. Before going into the detailed description of the shape predicting model, we will discuss the learning procedure because it impacts the desired property of the model. In the most general sense, a draping procedure is a coordinated movement of the robot and the linear actuators, i.e. a path in 48+6 = 54 dimensional space. To learn such a general trajectory in 54-dimensional space is computationally infeasible. Hence, we divide the learning into a sequence part and a trajectory part. The sequence part learning selects the starting point and the wave pattern for placing the suction cups. The trajectory part learning derives - for each step in the sequence - the trajectories of the relevant suction cups, which are those to be placed next and their neighbours. For the sequence part, there is in practise typically only around 5-10 relevant options, so the main task is the derivation of the trajectory part.

With the learning procedure in mind, we have the following desired properties of the model

- It must be fast computable to allow many simulations during learning
- It should be local so that a given region relevant for a step in sequence can be simulated independent of the rest of the ply
- The model must be accurate enough with respect to reality to allow for simulation-based learning

4 RELATED WORK

Modelling of composite materials is a broad topic and therefore we will only be able to give a rather sparse overview of some of the more detailed strategies. In the work by (Lin et al., 2012), the physical properties are modelled not only for the cloth as a whole but also the interactions between fibers. In (Newell and Khodabandehloo, 1995), the authors have defined a simplified mathematical model using finite element analysis to model deformations of plies, which is then used to control the handling trajectories of an automated composite manufacturing facility. More recently, (Krogh et al., 2017) used a finite element approach as part of the Flexdraper project. The calculation time

for simulating a drape using these type of methods is typically in the order of hours on a conventional computer. Finite difference modelling is another interesting approach used by (Do et al., 2006) to calculate deformation profiles of rectangular composite plies under the effect of multiple suction surfaces. Their results show good correspondence between expected results and simulations. They illustrate the importance of mesh vs accuracy vs computation time. In order to reach high accuracies their computational time was close to 30 minutes, whereas medium accuracies could be obtained with computational time in the range of 10 seconds. These physics based methods however all have a tendency of being computationally expensive. Therefore, we look a bit broader for fast modeling of deformable objects. Mass Spring Models is a typical approach (see e.g.(Kot et al., 2014)) where the approach is outlined in a nice detail. The paper (Liu et al., 2013) focuses more on the speed of the method. More advanced versions were presented in e.g. (Zhou et al., 2008) where cloth animation using mass spring models with dynamic stiffness is presented. The focus is on obtaining buckling and wrinkling. Mass spring models are relatively fast dynamical approximations, however for the material we are working with, the springs coefficients would be very high. Therefore timesteps for modelling would be very low and overall the model would hence again become computationally expensive. There has therefore been a lack of computationally fast and accurate model for prepreg fiber plies.

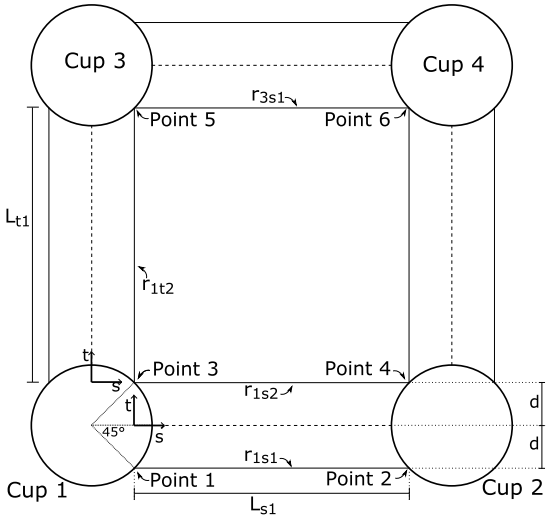


Figure 4: Scheme of cups on ply.

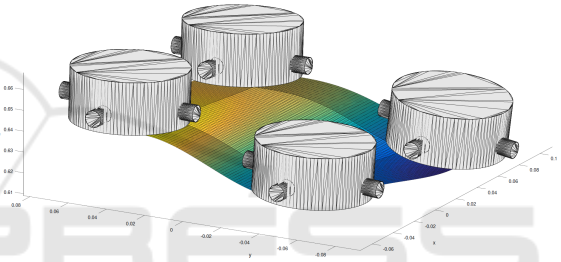


Figure 5: Model prediction for segment with 4 suction cups.

5 THE MODEL

In this section, we use the following notational conventions: a, b, c are used for scalars, $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$ are [3x1] vectors in x,y,z and $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ are [3x3] matrices. As mentioned, the input to our system are the positions and orientations of all the suction cups. We just need to derive formulas for the region around four suction cups as illustrated in Figure 4.

The first step is to model a fiber held by two suction cups. Using the assumption that the fiber direction is aligned with the suction cup array, as well as the information that the fiber ply was picked up from a flat configuration, it can be reasoned that a fiber runs from "Point 1" to "Point 2" in Figure 4. Similarly the fibers run along the other solid lines in Figure 4. To start defining the deflection model, we introduce a function $\boldsymbol{r}_{i\xi j}(\xi)$, where i is the suction cup number, $\xi \in \{s, t\}$ is the location along the line and j indicates which spline, with numbers 1 and 2 located as indicated on the figure. The function $\boldsymbol{r}_{i\xi j}(\xi)$ is hence a function of either s or t, and is defined in the interval $[0, L_{sj}]$ and $[0, L_{tj}]$ for $\boldsymbol{r}_{isj}(s)$ and $\boldsymbol{r}_{itj}(t)$ respectively. For a given 3D position and orientation of the four involved suction cups, the vector $\boldsymbol{r}_{i\xi j}(\xi)$ defines the model prediction of the $3D$ location of the point (i, ξ, j) on the line. It is derived in Section 5.1

Using the assumption of continuity, the region spanned by the points 1-4 in Figure 4 can be approximated by linearly interpolating between $\boldsymbol{r}_{1s1}(s)$ and $\boldsymbol{r}_{1s2}(s)$. The function describing this interpolation is called $\boldsymbol{\rho}_{i\xi}(s, t)$ and is a function (derived in Section 5.2) of both s and t, where i and ξ are inherited from the $\boldsymbol{r}_{i\xi j}(\xi)$ function. $\boldsymbol{\rho}_{1s}$ is a function in t for a given s, defined for $t \in [-d, d]$ and $s \in [0, L_s]$.

As a last step the area spanned by the points 3-6 in Figure 4 can be approximated by performing a polynomial fit between $\boldsymbol{\rho}_{1s}(s, t)$ and $\boldsymbol{\rho}_{3s}(s, t)$, as well as performing polynomial fit between $\boldsymbol{\rho}_{1t}(s, t)$ and $\boldsymbol{\rho}_{2t}(s, t)$. These functions are referred to as $\boldsymbol{\gamma}_{\xi}(s, t)$, where ξ is inherited from $\boldsymbol{\rho}_{i\xi}(\xi)$. To achieve first order continuity everywhere, these two polynomial fits are combined using a weight function, denoted $\boldsymbol{w}$, resulting in $\boldsymbol{\gamma}(s, t)$. The derivation of these functions can be found in Section 5.3.

5.1 Derivation of $\boldsymbol{r}$

Consider first as an example $\boldsymbol{r}_{1s1}(s)$. Given that the suction cups position and rotation are well known, and that the material is continuous differentiable, the following boundary conditions apply

$$\boldsymbol{r}_{1s1}(0) = \boldsymbol{T}_1(d, -d, 0)' + (X_1, Y_1, Z_1)' \quad (1)$$

$$\boldsymbol{r}_{1s1}(L_s) = \boldsymbol{T}_2(-d, -d, 0)' + (X_2, Y_2, Z_2)' \quad (2)$$

$$\boldsymbol{r}'_{1s1}(0) = \boldsymbol{T}_1(1, 0, 0)' \quad (3)$$

$$\boldsymbol{r}'_{1s1}(L_s) = \boldsymbol{T}_2(1, 0, 0)' \quad (4)$$

where the vector $(X_i, Y_i, Z_i)'$ is the center position of suction cup i on the suction surface wrt. a global coordinate frame aligned with the pick position and $\boldsymbol{T}_i$ is the rotation of the suction cup relative to the global frame i.

We will have the same types of boundary conditions for all the $\boldsymbol{r}_{isj}(s)$ and $\boldsymbol{r}_{itj}(t)$. Hence, our general problem is to find a function $\boldsymbol{r}(x)$ satisfying

$$\boldsymbol{r}(0) = \boldsymbol{r}_0 \quad (5)$$

$$\boldsymbol{r}(L_x) = \boldsymbol{r}_{L_x} \quad (6)$$

$$\boldsymbol{r}'(0) = \boldsymbol{r}'_0 \quad (7)$$

$$\boldsymbol{r}'(L_x) = \boldsymbol{r}'_{L_x} \quad (8)$$

Having four boundary conditions, the simplest choice for a continuous function will be a cubic polynomial.

$$\boldsymbol{r}(x) = \boldsymbol{a}'_3 x^3 + \boldsymbol{a}'_2 x^2 + \boldsymbol{a}'_1 x + \boldsymbol{a}'_0 \quad (9)$$

Assuming independence between $\{x, y, z\}$ the a's are given as follows

$$\boldsymbol{a}'_0 = \boldsymbol{r}_0 \quad (10)$$

$$\boldsymbol{a}'_1 = \boldsymbol{r}'_0 \quad (11)$$

$$\boldsymbol{a}'_2 = \frac{3(\boldsymbol{r}_{L_x} - \boldsymbol{r}_0) - L_x(2\boldsymbol{r}'_0 + \boldsymbol{r}'_{L_x})}{L_x^2} \quad (12)$$

$$\boldsymbol{a}'_3 = \frac{-2(\boldsymbol{r}_{L_x} - \boldsymbol{r}_0) + L_x(\boldsymbol{r}'_0 + \boldsymbol{r}'_{L_x})}{L_x^3} \quad (13)$$

The polynomial in Eq. 9 does not take care of the inextensibility of the fibers. To deal with that boundary condition, another term is added to the polynomial

$$\boldsymbol{r}_{L_x}(x) = \boldsymbol{r}(x) + a'_4 \boldsymbol{d} x^2 (L_x - x)^2 \quad (14)$$

The extra term $a'_4 \boldsymbol{d} x^2 (L_x - x)^2$ will have no influence on the previously derived $\boldsymbol{a}'_0$ to $\boldsymbol{a}'_3$, as both the value and its derivative is 0 at $x = 0$ and $x = L_x$. It is important to note that $\boldsymbol{a}'_0$ to $\boldsymbol{a}'_3$ are vectors in x,y,z whereas a'_4 is a scalar. For reasons below, we choose the term $\boldsymbol{d}$ as

$$\boldsymbol{d} = \boldsymbol{r}(\tfrac{L_x}{2}) - \tfrac{1}{2}(\boldsymbol{r}(0) + \boldsymbol{r}(L_x)) \quad (15)$$

To compute a'_4, we use that the arc length is given by

$$L_x \equiv \int_0^{L_x} \sqrt{\boldsymbol{r}'_{L_x}(x)^2}\, dx \quad (16)$$

In order to keep these derivations at an analytical level Eq. 16 is manipulated slightly

$$L_x = \int_0^{L_x} \sqrt{1 + (\boldsymbol{r}'_{L_x}(x)^2 - 1)}\, dx \quad (17)$$

under the assumption that $\boldsymbol{r}'(x)^2$ is relatively close to 1, a series expansion around 1 is performed.

$$\sqrt{1 + x} \approx 1 + \tfrac{1}{2}x \quad (18)$$

Resubsituting gives

$$\begin{aligned} L_x &\approx \int_0^{L_x} (\tfrac{1}{2} + \tfrac{1}{2}\boldsymbol{r}'_{L_x}(x)^2)\, dx \\ &= \frac{L_x}{2} + \frac{1}{2}\int_0^{L_x} \left\| \boldsymbol{r}(x) + a'_4 \boldsymbol{d} x^2 (L_x - x)^2 \right\|^2 dx \end{aligned} \quad (19)$$

This trick makes the integrand a polynomial which can easily be solved analytically. Inserting Eq. 9 into Eq. 19, multiplying by $\frac{2}{L_x}$ and reordering with respect to a'^2_4 and a'_4 results in.

$$\begin{aligned} 0 = &\left[\frac{2}{105}\|\boldsymbol{d}\|^2 L_x^6\right] a'^2_4 \\ &+ \left[-\frac{2}{15}(\boldsymbol{a}'_2 \cdot \boldsymbol{d}) L_x^4 - \frac{1}{5}(\boldsymbol{a}'_3 \cdot \boldsymbol{d}) L_x^5\right] a'_4 \\ &+ \|\boldsymbol{a}'_1\|^2 + \frac{4}{3}\|\boldsymbol{a}'_2\|^2 L_x^2 + 3(\boldsymbol{a}'_2 \cdot \boldsymbol{a}'_3) L_x^3 \\ &+ \frac{9}{5}\|\boldsymbol{a}'_3\|^2 L_x^4 + 2(\boldsymbol{a}'_1 \cdot \boldsymbol{a}'_2) L_x + 2(\boldsymbol{a}'_1 \cdot \boldsymbol{a}'_3) L_x^2 - 1 \end{aligned} \quad (20)$$

Eq. 20 can be redefined as

$$\alpha a'^2_4 + \beta a'_4 + \gamma = 0 \quad (21)$$

Hence the solutions are

$$a'_4 = \frac{-\beta \pm \sqrt{\beta^2 - 4\alpha\gamma}}{2\alpha} \quad (22)$$

This will produce two solutions for a'_4. To choose the correct solution, we first notice that α is positive(see Eq.20). If γ is negative, we should extend the length. Here we extend the length in the positive $\boldsymbol{d}$ direction. Hence, we must choose the "+" solution. If γ is positive, we must shorten the length, and here we wish to go in the negative $\boldsymbol{d}$ direction by choosing the least negative of the two solutions.

This provides a full description of all the $\boldsymbol{a}'$ parameters in Eq.14. For simplicity in implementation and the further derivations, Eq.14 will be redefined by expanding the brackets and reordering with respect to x to get a simple looking polynomial.

$$
\begin{aligned}
\boldsymbol{r}_{L_x}(x) &= \boldsymbol{a}'_3 x^3 + \boldsymbol{a}'_2 x^2 + \boldsymbol{a}'_1 x + \boldsymbol{a}'_0 \\
&\quad + a'_4 \boldsymbol{d} x^2 (L_x - x)^2 \\
&= (a'_4 \boldsymbol{d}) x^4 + (\boldsymbol{a}'_3 - 2a'_4 \boldsymbol{d} L_x) x^3 \\
&\quad + (\boldsymbol{a}'_2 + a'_4 \boldsymbol{d} L_x^2) x^2 + \boldsymbol{a}'_1 x + \boldsymbol{a}'_0 \\
&\equiv \boldsymbol{a}_4 x^4 + \boldsymbol{a}_3 x^3 + \boldsymbol{a}_2 x^2 + \boldsymbol{a}_1 x + \boldsymbol{a}_0 \\
&\equiv \sum_{k=0}^{4} \boldsymbol{a}_k x^k \qquad (23)
\end{aligned}
$$

Below we use the notation $\boldsymbol{r}_{isj}(s)$ and $\boldsymbol{a}_{isjk}$ for $\boldsymbol{r}_{L_x}(x)$ and $\boldsymbol{a}_k$ respectively for any considered isj. Similar notation is used for $\boldsymbol{r}_{itj}(t)$ and $\boldsymbol{a}_{itjk}$. All these functions and coefficients are as mentioned computed in exactly the same way using the four end point constraints and the constraint on arc length.

5.2 Derivation of $\boldsymbol{\rho}$

Similar to the previous section, the derivation will only be detailed for $\boldsymbol{\rho}_{1s}(s,t)$ as it is straightforward to generalize to all the $\boldsymbol{\rho}$.

Using the assumption of continuity, $\boldsymbol{\rho}_{1s}(s,t)$ can be approximated by linearly interpolating between $\boldsymbol{r}_{1s1}(s)$ and $\boldsymbol{r}_{1s2}(s)$. For any given s, $\boldsymbol{\rho}_{1s}(s,t)$ will have a linear dependency in t ,that is:

$$\boldsymbol{\rho}_{1s}(s,t) = \boldsymbol{b}_{1s1}(s)t + \boldsymbol{b}_{1s0}(s) \qquad (24)$$

which then leads to the following conditions.

$$\boldsymbol{\rho}_{1s}(s,d) = \boldsymbol{r}_{1s2}(s) = \boldsymbol{b}_{1s1}(s)d + \boldsymbol{b}_{1s0}(s) \qquad (25)$$

$$\boldsymbol{\rho}_{1s}(s,-d) = \boldsymbol{r}_{1s1}(s) = -\boldsymbol{b}_{1s1}(s)d + \boldsymbol{b}_{1s0}(s) \qquad (26)$$

Subtracting 25 and 26 gives

$$\boldsymbol{b}_{1s1}(s) = \frac{\boldsymbol{r}_{1s2}(s) - \boldsymbol{r}_{1s1}(s)}{2d} \qquad (27)$$

Adding 25 and 26 gives

$$\boldsymbol{b}_{1s0}(s) = \frac{\boldsymbol{r}_{1s2}(s) + \boldsymbol{r}_{1s1}(s)}{2} \qquad (28)$$

Using the formulae for the $\boldsymbol{r}_{isj}(s)$'s, we get

$$
\begin{aligned}
\boldsymbol{b}_{1s0}(s) &= \sum_{k=0}^{4} \frac{\boldsymbol{a}_{1s2k} - \boldsymbol{a}_{1s1k}}{2} s^k \\
&\equiv \sum_{k=0}^{4} \boldsymbol{b}_{1s0k} s^k \qquad (29)
\end{aligned}
$$

and

$$
\begin{aligned}
\boldsymbol{b}_{1s1}(s) &= \sum_{k=0}^{4} \frac{\boldsymbol{a}_{1s2k} - \boldsymbol{a}_{1s1k}}{2d} s^k \\
&\equiv \sum_{k=0}^{4} \boldsymbol{b}_{1s1k} s^k \qquad (30)
\end{aligned}
$$

where the $\boldsymbol{a}_{1s1k}$'s were discussed in the previous section. Complelety equivalent, we can find the $\boldsymbol{\rho}_{is}(s,t)$, $\boldsymbol{b}_{is0}(s)$, $\boldsymbol{b}_{is1}(s)$ for $i = 3$ and the $\boldsymbol{\rho}_{it}(s,t)$, $\boldsymbol{b}_{it0}(t)$, $\boldsymbol{b}_{it1}(t)$ for $i = 1, 2$.

5.3 Derivation of $\boldsymbol{\gamma}$

We now consider the model of deformation in the region between the four innermost lines. We first consider the curve $\boldsymbol{\gamma}_s(s,t)$ interpolating between $\boldsymbol{\rho}_{1s}(s,d)$ and $\boldsymbol{\rho}_{3s}(s,-d)$. The derivation of the model for $\boldsymbol{\gamma}_s$ is very similar to that of $\boldsymbol{r}$ as the material is still assumed first order continuous and therefore will have both position and first derivative as boundary conditions. Similar to $\boldsymbol{\rho_{1s}}$, $\boldsymbol{\gamma}_s$ is considered a function of t, for a given s.

$$
\begin{aligned}
\boldsymbol{\gamma}_s(s,0) &= \boldsymbol{\gamma}_{s0} = \boldsymbol{\rho}_{1s}(s,d) \\
&= \boldsymbol{b}_{1s1}(s)d + \boldsymbol{b}_{1s0}(s) \qquad (31) \\
\boldsymbol{\gamma}_s(s,L_t) &= \boldsymbol{\gamma}_{sL_t} = \boldsymbol{\rho}_{3s}(s,-d) \\
&= -\boldsymbol{b}_{3s1}(s)d + \boldsymbol{b}_{3s0}(s) \qquad (32)
\end{aligned}
$$

$$\frac{\partial \boldsymbol{\gamma}_s(s,0)}{\partial t} = \boldsymbol{\gamma}'_{s0} = \frac{\partial \boldsymbol{\rho}_{1s}(s,d)}{\partial t} = \boldsymbol{b}_{1s1}(s) \qquad (33)$$

$$\frac{\partial \boldsymbol{\gamma}_s(s,L_t)}{\partial t} = \boldsymbol{\gamma}'_{sL_t} = \frac{\partial \boldsymbol{\rho}_{3s}(s,-d)}{\partial t} = \boldsymbol{b}_{3s1}(s) \qquad (34)$$

the simplest choice for four boundary conditions is a cubic polynomial in t.

$$\boldsymbol{\gamma}_s(s,t) = \boldsymbol{c}_{s3}(s)t^3 + \boldsymbol{c}_{s2}(s)t^2 + \boldsymbol{c}_{s1}(s)t + \boldsymbol{c}_{s0}(s) \qquad (35)$$

which leads to a similar solution as for $\boldsymbol{r}$

$$\boldsymbol{c}_{s0}(s) = \boldsymbol{\gamma}_{s0} \qquad (36)$$

$$\boldsymbol{c}_{s1}(s) = \boldsymbol{\gamma}'_{s0} \qquad (37)$$

$$\boldsymbol{c}_{s2}(s) = \frac{3(\boldsymbol{\gamma}_{sL_t} - \boldsymbol{\gamma}_{s0}) - L_t(2\boldsymbol{\gamma}'_{s0} + \boldsymbol{\gamma}'_{sL_t})}{L_t^2} \qquad (38)$$

$$\boldsymbol{c}_{s3}(s) = \frac{-2(\boldsymbol{\gamma}_{sL_t} - \boldsymbol{\gamma}_{s0}) + L_t(\boldsymbol{\gamma}'_{s0} + \boldsymbol{\gamma}'_{sL_t})}{L_t^3} \qquad (39)$$

Notice that as the material is not clamped at the boundaries, we do here not constrain the arc length to a fixed value. Observe now that we can write

$$\boldsymbol{c}_{sj}(s) = \sum_{i=0}^{4} \boldsymbol{c}_{sij} s^i \qquad (40)$$

where the $\boldsymbol{c}_{sij}$'s are constants. The constants can be computed by inserting Eq. 29 and Eq. 30 and their equivalents into the boundary conditions and subsequently the boundary conditions into the equations for

the $\boldsymbol{c}_{sj}(s)$'s. We get

$$\boldsymbol{c}_{si0} = d\boldsymbol{b}_{1s1i} + \boldsymbol{b}_{1s0i} \tag{41}$$

$$\boldsymbol{c}_{si1} = \boldsymbol{b}_{1s1i} \tag{42}$$

$$\boldsymbol{c}_{si2} = \frac{3(d\boldsymbol{b}_{3s1k} + \boldsymbol{b}_{3s0k} - d\boldsymbol{b}_{1s1k} - \boldsymbol{b}_{1s0k})}{L_t^2} - \frac{2\boldsymbol{b}_{1s1k} + \boldsymbol{b}_{3s1k}}{L_t} \tag{43}$$

$$\boldsymbol{c}_{si3} = \frac{2(d\boldsymbol{b}_{3s1k} - \boldsymbol{b}_{3s0k} + d\boldsymbol{b}_{1s1k} + \boldsymbol{b}_{1s0k})}{L_t^3} + \frac{\boldsymbol{b}_{1s1k} + \boldsymbol{b}_{3s1k}}{L_t^2} \tag{44}$$

We then get

$$\boldsymbol{\gamma}_s(s,t) = \sum_{i=0}^{4}\sum_{j=0}^{3} \boldsymbol{c}_{sij} s^i t^j \tag{45}$$

Completely in an equivalent way, we derive $\boldsymbol{\gamma}_t(s,t)$ as the model of the curve between $\boldsymbol{\rho}_{1t}(d,t)$ and $\boldsymbol{\rho}_{2t}(-d,t)$.

Finally, to get smooth interpolations, also at the corners, a location dependent weighted average of $\boldsymbol{\gamma}_s(s,t)$ and $\boldsymbol{\gamma}_t(s,t)$ is used:

$$\boldsymbol{\gamma}(s,t) = \frac{w_t(s,t)\boldsymbol{\gamma}_t(s,t) + w_s(s,t)\boldsymbol{\gamma}_s(s,t)}{w_s(s,t) + w_t(s,t)} \tag{46}$$

where the weight functions are given as.

$$w_s(s,t) = \frac{\varepsilon + f(t)w(\frac{t}{L_t}) + f(s)(1 - w(\frac{s}{L_s}))}{\varepsilon + f(t) + f(s)} \tag{47}$$

$$w_t(s,t) = \frac{\varepsilon + f(t)(1 - w(\frac{t}{L_t})) + f(s)w(\frac{s}{L_s})}{\varepsilon + f(t) + f(s)} \tag{48}$$

with the following sub functions

$$w(\xi) = 0,5(e^{-a\xi^2} + e^{-a(1-\xi)^2}) \tag{49}$$

$$f(\xi) = \left(\frac{\xi}{L_\xi}\right)^k (1 - \frac{\xi}{L_\xi})^k \tag{50}$$

where a is a tunable parameter to adjust the smoothness of the weighing. and ε is a small number to ensure numerical stability. A heat plot of the weight function is shown in Figure 6.

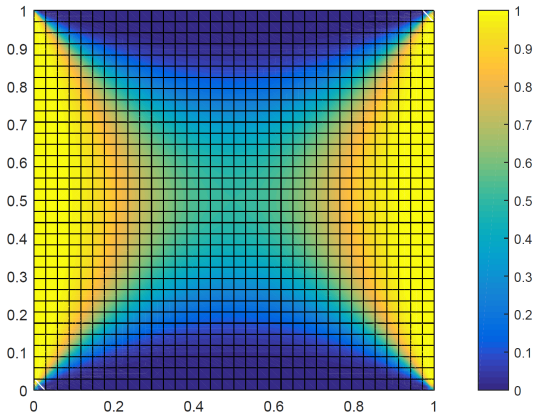


Figure 6: Overview of the 2D weight function, $a = 10^4$.

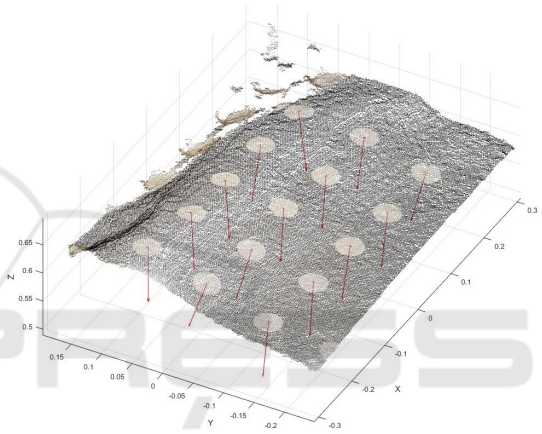


Figure 7: Pointcloud captured of the material in a held/shaped configuration. The suction cups are displayed with white markers, the normals are shown from the suction cup centers.

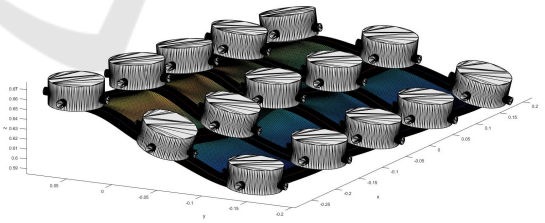


Figure 8: Model prediction of material placement when held by suction cups as detected in Figure 7.

6 VALIDATION

The test case examines the shape of a ply while being held by an array of suction cups. We consider a scenario where a full size ply was picked up from flat surface and the suction cup configuration was reshaped into a typical tool-mould shape. The validation study involves an interior region of the ply. As input to the model, we need the L_s and L_t values for the considered region. The current suction cups are not optimal, so to avoid mistakes due to sliding, we estimate L_s and L_t from the shaped configuration. To estimate this arc length and filtering out noise, a NURBS surface-fit has been performed on the point cloud segment.The suction cup positions are estimated by manually placing markers on the ply, for the vision system to detect. These markers are also used to estimate the normals of the suction cups. The vision system is a commercially available "Apple PrimeSense

Table 1: Summary of validation results.

RMS Error	1.6 mm
Max Error	4.6 mm
Calculation Time	<1 second

Carmine 1.09". The recorded data can be seen in Figure.7 where as the model prediction can be seen in Figure 8. The visualization shown in Figure 8 takes less than one second to calculate using non optimized MATLAB code on a conventional PC. The metric used for this evaluation only takes the offset in the "z" direction into account, and does not perform a one to one mapping of points in the ply. Seen over the whole test area (5 by 3 suction cups), the maximum deviation between ply and model is 4.6 mm and the RMS error is 1.6 mm. This is comparable to the spatial resolution of 1 mm for the used camera configuration. This is already quite satisfying for the purpose of learning draping strategies and with better suction cups and camera estimates, we expect the accuracy to be further improved.

It should however be noted that there in the test data are some unrealistic discrepancies with respect to suction cup positions. These test data errors give rise to the models arc length adjustment failing on some lines, and as a result some unexpected bulges are formed.

7 CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a fast computable model for predicting the shape of prepreg fiber plies clamped in the corners by suction cups. Our presentation included a detailed derivation of the model parameters as function of the suction cup positions and orientations. We also did some preliminary experimental validations and found that the accuracy of the model is promising. However, the quality of the experimental setting was partly insufficient and therefore further validation in is needed.

Together with our partners, we are currently improving the experimental settings by replacing the suction cups with new versions customized for composites. Furthermore, the sensor settings is being significantly improved. Our model currently only supports completely "free hanging" regions of the ply. In the near future, we will therefore extend the model to include situations where part of the region between four suction cups has made contact with the mould and where we estimate only the shape of the remaining part. This will lead to some additional issues because we need to estimate the boundary of the free hanging part, which will be a 3D curve and interpolate the shape from that boundary to the remaining 1, 2 or 3 suction cups.

ACKNOWLEDGEMENTS

This work was supported by Innovation Fund Denmark through the strategic platform MADE - Platform for Future Production and the project FlexDraper.

REFERENCES

Do, D., John, S., and Herszberg, I. (2006). 3d deformation models for the automated manufacture of composite components. *Composites Part A: Applied Science and Manufacturing*, 37(9):1377–1389.

Ellekilde, L., Wilm, J., Nielsen, O. W., Krogh, C., Glud, J. A., Gunnarsson, G. G., Stenvang, T. S., Kristiansen, E., J. Jakobsen, M. K., AanÃęs, H., de Kruijk, J., Sveidahl, I., Ikram, A., and Petersen, H. (2018). An outline of the design and control of the flexdraper robot system for automatic draping of prepreg composites fabrics. To be submitted.

Kot, M., Nagahashi, H., and Szymczak, P. (2014). Elastic moduli of simple mass spring models. *The Visual Computer*, pages 1–12.

Krogh, C., Glud, J. A., and Jakobsen, J. (2017). Modeling of prepregs during automated draping sequences. In *AIP Conference Proceedings*, volume 1896, pages 030–036. AIP Publishing.

Lin, H., Ramgulam, R., Arshad, H., Clifford, M., Potluri, P., and Long, A. (2012). Multi-scale integrated modelling for high performance flexible materials. *Computational Materials Science*, 65:276–286.

Liu, T., Bargteil, A. W., O'Brien, J. F., and Kavan, L. (2013). Fast simulation of mass-spring systems. *ACM Transactions on Graphics (TOG)*, 32(6):214.

Newell, G. and Khodabandehloo, K. (1995). Modelling flexible sheets for automatic handling and lay-up of composite components. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 209(6):423–432.

Zhou, C., Jin, X., Wang, C. C., and Feng, J. (2008). Plausible cloth animation using dynamic bending model. *Progress in Natural Science*, 18(7):879–885.