2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

2021 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9077-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICRA48506.2021.9560939

A Simulation-Based Grasp Planner for Enabling Robotic Grasping during Composite Sheet Layup

Omey M. Manyar†, Jaineel Desai†, Nimish Deogaonkar†, Rex Jomy Joesph†, Rishi Malhan†, Zachary McNulty†, Bohan Wang*, Jernej Barbič*, and Satyandra K. Gupta†

***Abstract*— Composites are increasingly becoming a material of choice in the aerospace and automotive industries. Currently, many composite parts are produced by manually laying up sheets on complex molds. Composite sheet layup requires executing two main tasks: (1) grasping a sheet and (2) draping it on the mold. Automating the layup process requires automation of these two tasks. This paper is focused on the automation of the grasping task using robots. This requires an automated generation of grasp plans to enable robots to hold the sheet during the draping process. We present a simulation-based approach for determining robot grasp locations on the composite sheets. We also present an intervention controller that uses a real-time sheet tracking system during plan execution and can prevent failures. We demonstrate the performance of the developed system using a large complex part.**

I. Introduction

In high-performance structural applications, there has been a considerable increase in the demand for composite parts. Tape layup and fiber layup processes are highly automated and used for making parts with simpler geometries. Layup of prepreg sheets on molds remains a popular method for making complex composite parts. During the layup process, multiple sheets are stacked on the mold by stretching and conforming them to the 3D shape of the mold. After completing the layup process, the tool and part are placed in an autoclave to cure the part. Traditionally, sheet layup process has been performed manually. Manual processes can be inconsistent and ergonomically challenging.

Automating the sheet layup process requires automation of grasping and draping tasks. In the past, we have developed a robotic cell for automating these two tasks for small parts [1]–[6]. This paper is focused on robot-based automation of the grasping task for large parts. Large parts are draped in multiple stages. In each stage, a small region of the sheet is draped. For every stage, we need to determine how robots should grasp the sheet. If the sheet is not grasped properly, the draping process will fail. Large sheets pose a variety of problems. The sheet may droop over the mold and make an undesirable contact with it. The sheet may also be damaged due to excessive tension. Automating the grasping process requires intricate robot planning with careful considerations to process constraints. We present a simulation-based approach for determining robot grasp locations on the sheets. We use a physics-based thin-shell simulator [7] to determine how the sheet will deform during the draping process. We use a state-space search to find the grasping plan. We search for grasping locations that meet all the relevant process constraints and minimize the robot operation time.

Large parts require multiple operators to perform the layup operation. In a two-operator setup, one operator grasps the sheet while the second operator drapes the sheet. In our experimental setup, we have built a three-robots and one-human cell (see Fig. 1). Two robots grasp the sheet during the draping process. A third robot is used to perform simple draping tasks. The human performs complex draping tasks. We also describe an intervention controller that uses real-time sheet tracking system during plan execution and can make minor adjustment to the grasp plan if simulation model and actual sheet behavior exhibits discrepancy due to aging-induced changes in the material. We demonstrate performance of the system using a large complex part.

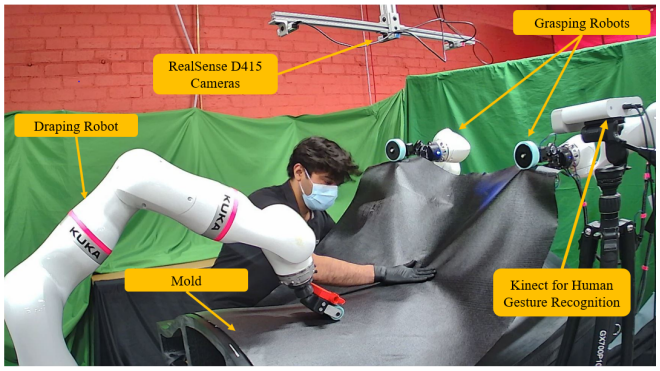


Fig. 1: A composite sheet layup cell consisting of three robots and one human.

II. Related Work

Work done in [8] was used to predict how a sheet will deform on a solid mold using numerical models. Numerical models were also built to predict prepreg behavior when it is grasped from specific points [9]. Kinematic algorithms to map discrete points on the sheet to a non-developeable surface have also been studied [10]. Such models are then used to estimate draping sequence which we use as an expert input in our work. Elkington et al. reported different techniques which are used by experts during layup to enable using motion primitives with robotic manipulators [11]. Researchers have also proposed cell concepts which can be used to automate the layup process in [12], [13]. However, these methods do not address the automated trajectory generation for multiple robots.

Work reported in [14] constructed a cell using industrial

†Center for Advanced Manufacturing, University of Southern California, Los Angeles, CA, USA
*Department of Computer Science, University of Southern California, Los Angeles, CA, USA

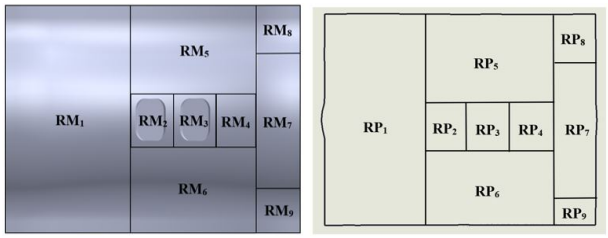
robot and manually programmed it to use custom end-effectors for applying pressure and conforming the sheet. Specialized grippers for handling carbon fiber are also required since we need to prevent the sheet from adhering to gripper surfaces, and also prevent sheet contamination [15], [16]. Our robotic cell extends the functionalities of such custom hardware for automation.

Picking and placing carbon fiber sheets is another area of active research. A detailed review of pick and place operations is provided in [17]. Similarly, state of the art grasping and automation technologies have been reviewed in [18]. Multi-arm manipulation of prepreg is what makes the process challenging since it requires coordination of different arms [19]–[28]. Robot motion plans need to be automatically generated for making the process economical. Survey papers on the manipulation of deformable objects include [29], [30]. Planning and control approaches have been developed for 1-D problems [31]–[38], cloth folding [39]–[51], and ply manipulation [52]–[55]. Most of these applications define a final shape of the material and planning and control algorithms are used to reach this desired shape. Layup, on the other hand, requires several intermediate steps to reach a desired shape on the mold. Each such step consists of applying pressure and deforming the viscoelastic material. Hence grasp planning algorithms need to account for the underlying physics and uncertainty. Thin-shell simulations are routinely done in multiple engineering communities, and are relatively well-understood [56], [57]. Although methods exist to tune thin-shell material properties to observations [58], thin-shell simulation alone is not sufficient for composite-sheet robotic grasp planning. This is because composite sheets must be laid in stages, and content-specific knowledge must be added to avoid damaging the sheets and ensure that real sheets are actually laid as predicted by simulation.

Learning-from-demonstration techniques have been used to solve some challenging manipulation problems [59]–[61]. However, the uncertainty in the process due to changing properties of the viscoelastic material over time makes the manipulation a challenging task. Additionally, the complex interaction between draping and manipulation is also difficult to learn. Our previous work done in [1]–[3], [6] proposed a grasp planner which generates tool paths for simpler geometries and executes them under impedance control. In this work, we extend our previous planning algorithms by a high fidelity physics simulator, account for the uncertainty during graph generation, and also process constraints that can accommodate newer variants of molds and larger sheets.

III. Problem Formulation

In this section, we formulate a multi-robot grasp planning problem for the composite layup process. Prepreg composite layup is executed in multiple stages that are characterized by the number of draping zones $\boldsymbol{n}$. A subject matter expert determines these zones for the prepreg along with the mold as depicted in Fig. 2. We represent these zones on the prepreg as RP_i and the corresponding ones on the mold as RM_i, where $i \in [1,2,\ldots,n]$ is an intermediary draping stage. These zones are defined such that there exists a 1:1 correspondence between RP_i and RM_i $\{RP_i \leftrightarrows RM_i\}$. The layup process is thus defined as a sequential procedure of conforming the draping zones of the prepreg RP_i to the corresponding ones on the mold RM_i.



a. Mold Draping Regions b. Prepreg Draping Regions

Fig. 2: (a) Definition of draping zones on the Mold, (b) Definition of corresponding draping zones on the Prepreg Sheet.

Let us consider an intermediate stage i of the draping process. We represent the prepreg composite as a deformable surface mesh where each element of the mesh is modelled with the prepreg's material parameters. At the stage i, the draping zones $1,\ldots,i-1$ have already been conformed to the corresponding regions on the mold; hence we only need to compute the grasping locations for the remaining portion of the prepreg.

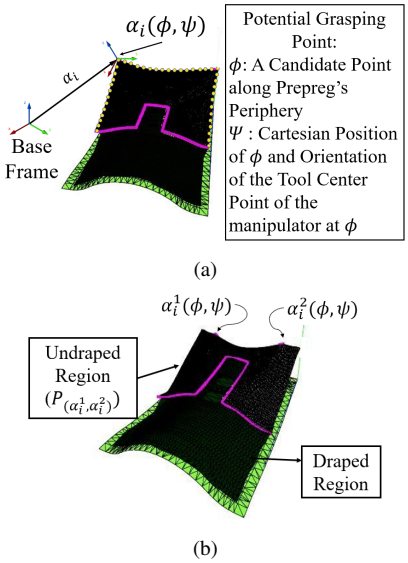


Fig. 3: (a) Potential Grasping Location with corresponding $\{\Phi,\Psi\}$ & (b) State Space Representation of the sheet

In order to understand how we define grasping locations, let us consider an example of the composite layup for a mold, Part A shown in Fig. 8. During the layup process, a prepreg is always grasped along its periphery, such potential grasping points along the edge of the prepreg at a draping stage i are depicted in Fig. 3a. We denote these candidate points by a variable Φ, where Φ represents a potential grasping point along prepreg's boundary at stage i. Additionally, every Φ can assume a 3D location $\{x,y,z,r,p,y\}$ * within the feasible

*note: The orientation $\{r,p,y\}$ is defined for the Tool Center Point of the manipulator that will grasp the prepreg

workspace, as shown in Fig. 3a. We denote this 3D location of Φ by another variable Ψ. Here, Ψ is a representation of the cartesian position and orientation of a particular grasping point Φ. Hence, a grasping location for the prepreg gets characterized as a tuple of variables $\{\Phi,\Psi\}$. We represent this tuple by $\alpha_i = \{\Phi,\Psi\}$ that denotes a grasping location at a stage i in the draping process.

In this study, we have focused on prepregs that can be supported by only two grasping locations. We denote these two grasping locations by α_i^1 and α_i^2 as depicted in Fig. 3b. At a particular value of α_i^1 and α_i^2, the undraped portion of the prepreg will assume certain configuration. We define this portion of the prepreg by $P_{\alpha_i^1,\alpha_i^2}$. Consequently, the free region of the mold at this stage on which draping is yet to be performed is represented as $M_{\alpha_i^1,\alpha_i^2}$. Fig. 3b gives an overview of these parameters for $i = 4$.

The state space representation $S_{\alpha_i^1,\alpha_i^2}$ of the overall system can therefore be formulated as follows.

$$S_{\alpha_i^1,\alpha_i^2} = \{\alpha_i^1, \alpha_i^2, P_{\alpha_i^1,\alpha_i^1}, M_{\alpha_i^1,\alpha_i^2}\}, \forall i \in \{1,2,...,n\} \quad (1)$$

A particular state $S_{\alpha_i^1,\alpha_i^2}$ is considered feasible if $S_{\alpha_i^1,\alpha_i^2}$ satisfies a set of process constraints. We have identified eight such process constraints:

1) *Elastic Energy*: The elastic energy [7] represents the degree of deformation experienced by $P_{\alpha_i^1,\alpha_i^2}$ under external forces and constraints. Elastic energy exceeding a threshold value is an indication that the prepreg is experiencing excessive deformation.
2) *Sheet to Mold Collision*: Collision between undraped sections of the prepreg and the mold can introduce innumerable defects. In the worst case, it might lead to scrapping of the currently manufactured part. This constraint determines whether $P_{\alpha_i^1,\alpha_i^2}$ and $M_{\alpha_i^1,\alpha_i^2}$ are in collision.
3) *Sheet Self Collisions*: At a particular state $S_{\alpha_i^1,\alpha_i^2}$, there is a possibility that the prepreg $P_{\alpha_i^1,\alpha_i^2}$ is self-colliding. Self collisions are undesirable in any configuration as they cause wrinkles and other major defects.
4) *Distance between the current Draping Region and the Mold*: To achieve successful draping for a zone $\{RP_i \leftrightarrows RM_i\}$, the layup technician needs to apply forces without affecting fiber alignment or overstretching the sheet. To ensure this, the distance between RP_i and RM_i should be below a minimum threshold.
5) *Distance between the Undraped Region and the Mold*: The undraped section of the prepreg should maintain a minimum threshold distance from the corresponding $\{RM_{i+1},...,RM_n\}$. This constraint ensures that there is no undesirable contact between the prepreg and the mold while the technician is draping a particular RP_i.
6) *Droop Factor*: Drooping is an undesirable phenomenon in layup process which can potentially lead to prepreg misalignment and self collisions. At a system state $S_{\alpha_i^1,\alpha_i^2}$, we define the droop factor by a linear function $f(d_1,d_2)$. Where, d_1 represents the maximum vertical distance between the extremities of the prepreg and $\{\alpha_i^1,\alpha_i^2\}$. While, d_2 represents the maximum vertical distance between the vertices of the grasped edge and $\{\alpha_i^1,\alpha_i^2\}$. If this deformation value is larger than a certain threshold, we can conclude that there is excessive drooping.
7) *Sheet Alignment*: A typical mold for composite draping possesses demarcations which define a bounding region for the prepreg draping. Sheet alignment is the measure of undershoot or overshoot of the prepreg $P_{\alpha_i^1,\alpha_i^2}$ beyond this demarcated region.
8) *Robot Manipulability Index*: We introduce this constraint as a planning constraint rather than a process constraint. The robot manipulability index [62] is a quality measure of closeness of the grasping manipulator to a singular configuration. This index ensures that the manipulator can transition between stages $i \rightarrow i+1$ successfully.

The states satisfying these constraints are then termed as feasible states. We represent one such feasible state at i by a variable ω_i. Additionally, we introduce a new parameter t_{i+1}^i which represents the time required for transitioning between contiguous states $\{\omega_i$ to $\omega_{i+1}\}$. The total time for the overall grasping process then gets defined by T, such that $T = \sum_{i=1}^{n-1} t_{i+1}^i$. We formulate the grasp planning problem as an optimization problem where the objective is to minimize T for the overall draping process. An optimal grasp plan Ω is thus represented by

$$\Omega = \{\omega_1, \omega_2, ..., \omega_n\}, \quad (2)$$

such that Ω minimizes the total time T across all possible combinations of feasible grasping locations ω_i, for the entire draping process (all n draping zones).

IV. Grasp Planning

In order to solve the multi robot grasp planning problem formulated in section III, we simulate the prepreg as a thin shell finite element model with appropriate material parameters using VegaFEM [7]. We use this model to simulate the sheet deformation at a state $S_{\alpha_i^1,\alpha_i^2}$. The thin shell FEA simulator can compute the Elastic Energy and Droop Factor constraints. The rest of the collision and proximity constraints are evaluated using the Flexible Collision Library [63]. We use this architecture to perform constraint satisfaction across $S_{\alpha_i^1,\alpha_i^2}$ to construct a search space graph of feasible states comprising of ω_i's. Subsequently, we compute a shortest-time path across the graph that defines the optimal grasp plan Ω.

A. State Space Discretization

State space representation for the grasp planning problem was outlined in Section III. The states are defined by $S_{\alpha_i^1,\alpha_i^2}$. Our objective is to explore the space of feasible grasping locations α_i^1 and α_i^2. We discretize our state space in order to search for feasible $S_{\alpha_i^1,\alpha_i^2}$'s. Based on experimentation, we use a discretization factor ranging between 3-5% of the prepreg dimensions. We discretize the variables Φ and Ψ accordingly. This makes the discrete search-space graph construction for the composite grasp planning a combinatorial problem which we tackle by bounding our search space.

B. Bounding the Search Space

Let us consider an input prepreg P approximated as a m-sided polygon as shown in Fig. 4. Since P_0 is a section of P that has already been draped, we won't be considering P_0 in our heuristic design. Similarly, P_1 is the incumbent section on which draping will be performed in direction $\vec{\mathbf{v}}$. The sections $\{P_2, P_3, P_4\}$ represents the undraped portion of the prepreg as described in Fig. 4. Parameters $\{a_i, b_i\}$ denote the characteristic length and width of the corresponding sheet region P_i. Based on our experiments with prepreg draping, we have observed that the direction of draping $\vec{\mathbf{v}}$ and the length and width of the entire prepreg $\{L, W\}$ play a crucial role in selecting the edge along which the prepreg should be grasped. The edge which is orthogonal to the direction of draping $\vec{\mathbf{v}}$ and is positioned at a maximum distance from P_1, is optimal for grasping the prepreg. Grasping along any other edges introduced process constraint violations. If a region P_1 has multiple directions of draping, we would choose multiple edges accordingly. This discards the potential $\omega_{i,j}$ which exist along the other edges.

In order to set the bounds on Ψ we consider the characteristic lengths and widths of P_1 and of the set $\{P_2, P_3, P_4\}$. The selected edge is divided into two sections for $\{\alpha_i^1, \alpha_i^2\}$. In our study, for an arbitrary Ψ we set the orientation (r, p, y) to a value equal to (r_e, p_e, y_e) which represents the orientation of selected edge $e \in \{1, m\}$ in the base frame. We introduce a bound in the cartesian space for corresponding $\Psi' s$ for $\{\alpha_i^1, \alpha_i^2\}$ as a function of the parameters $\{a_1, b_1, a_2, b_2, a_3, b_3, L, W\}$.

Additionally, we assume symmetry between the positions of the manipulators along the draping axis. This assumption is valid only if draping is performed along the central axis of the prepreg. This further constrains the search space. Hence we achieve a bounding region within which we can search for all feasible grasping locations $\omega_i' s$ for every draping stage $\{1, 2, ..., n\}$. Once we generate a set of appropriate $\{\Phi, \Psi\}$ for $\{\alpha_i^1, \alpha_i^2\}$ at i^{th} stage based on our heuristic, we apply the eight constraints and populate the search space graph with all the feasible ω_i. This leads to the generation of a search-space graph of depth n, recall that n is the total number of draping zones.

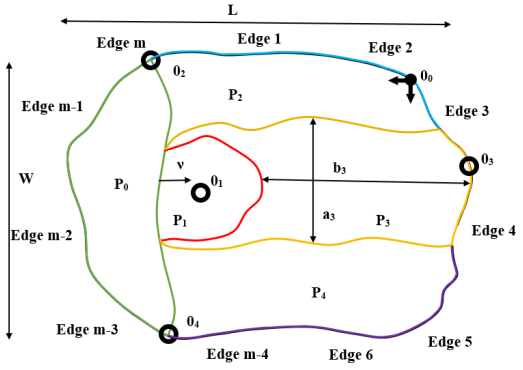


Fig. 4: The Prepreg P is divided into different sections. P_0 : Draped Section, P_1 : section about to Be Draped, P_2 : Left Undraped section, P_3 : Front Undraped section, P_4 : Right Undradped section, a_3, b_3 : Characteristic length and width of the section with index 3.

The objective of the graph-based search is to compute a grasp plan that can be executed in the minimal amount of total execution time T, as defined in Section III. Currently, we have a set of nodes representing the feasible grasping locations $\omega' s$. To create a complete graph $G = \{V, E\}$ we need to define and compute the cost associated with edges E.

During draping, the manipulators are operating under impedance control. This is mainly done to avoid any potential deformation of the prepreg under external draping forces. As a result of the impedance control, the robots tend to move in a direction radially towards the line of draping which is defined by the boundary between the draped and the undraped section of the prepreg. This movement is proportional to the impedance control parameters. In order to account for this minor displacement, we add an additional layer at each i representing the displacement in the location of the robots. We denote this layer by i'. The manipulators travel to the next region $i+1$ from this intermediate layer.

C. Graph Construction

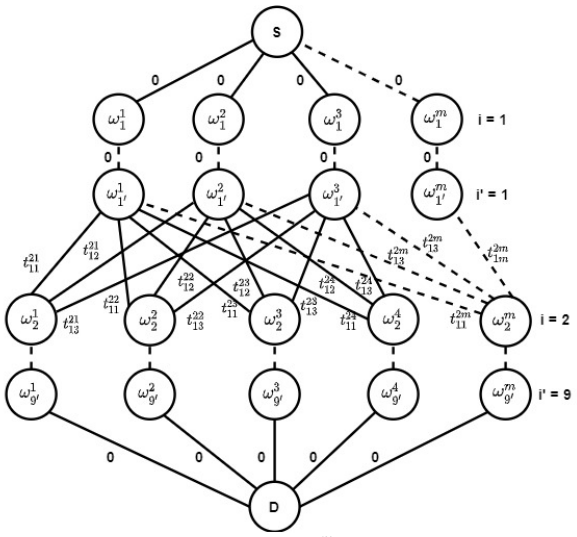


Fig. 5: Grasp Planner Search Graph. ω_1^m is one of the feasible states at $i = 1$ and t_{11}^{21} is time taken to travel from node $\omega_{1'}^1$to ω_2^1

We set the edge cost to the time required for the manipulator to travel from region i' to $i+1$. Note that edges are added only among nodes in adjacent regions. The transition time cost of the manipulators for $i \rightarrow i'$ can be assigned to zero as this transition does not influence the overall time for grasping actions. In this work we have assumed that the manipulators will execute the trajectories at nearly constant velocities. They follow a linear path along the edge in case of a change in the grasping vertex, and then diagonally to the next grasping location. We compute the time $t_{i'}^{i+1}$, which gives us the edge cost and therefore completes the construction of our search-space graph. Furthermore, to aid with finding the path of least cost and in the spirit of dynamic programming, we add a pseudo source node connected to the first layer with zero cost edges. Similarly, a pseudo destination node is added at the n^{th} layer with zero cost

edges. Refer Fig. 5 for structure of the search space graph.

D. Grasp Plan Generation

Once the search space graph G is constructed, the computation of shortest-time path becomes a shortest-path search problem. We compute the shortest path from the source node to destination node using Dijkstra's algorithm. The computed path is our solution Ω which represents the optimal grasp plan for the two manipulators. The overall grasp planning process is depicted by Fig. 6.

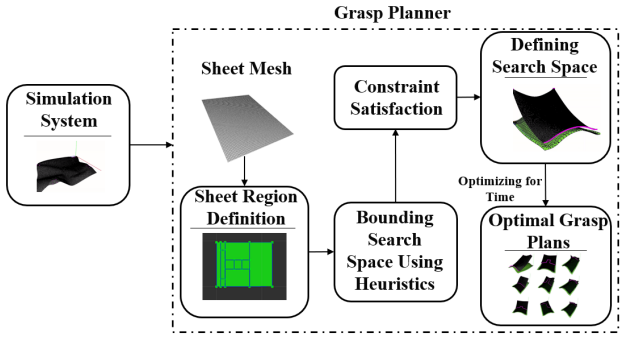


Fig. 6: Overall Process Flow for Grasp Planning.

E. Results on Representative Examples

Our planner was successful in computing feasible grasp plans for the molds with a wide variety of complexities. Fig. 8 shows three representative examples. We conducted physical experiments on the mold Part A (see Fig. 7). Table I displays the overall time analysis of the search. The time taken for search is directly proportional to the number of vertices and edges of the prepreg mesh, and the number of discrete states across which we conduct the search.

TABLE I: Grasp Planner Results

Type of Mold	No. of Draping Regions (n)	Plan Generation Time (secs)	Total Grasp Plan Process Time (secs)
Part A	9	945	154
Part B	6	641	122
Part C	8	236	135

V. Intervention Controller

A. Overview

The viscoelasticity of prepreg materials lead to a change in its material properties over time, which introduces errors in the material parameter model. The plans generated from our proposed methodology in section IV exhibit an inherent dependency on the material parameter model. A potential error in the estimation of the material parameters can lead to inaccuracies in the grasping locations. An online closed-loop system is needed to check the integrity of the prepreg draping process, and raise alert in case of any deviations from the ideal planned scenario. We introduce an intervention controller system that acts as an online monitoring and verification system for the grasp plans generated by our planning algorithm.

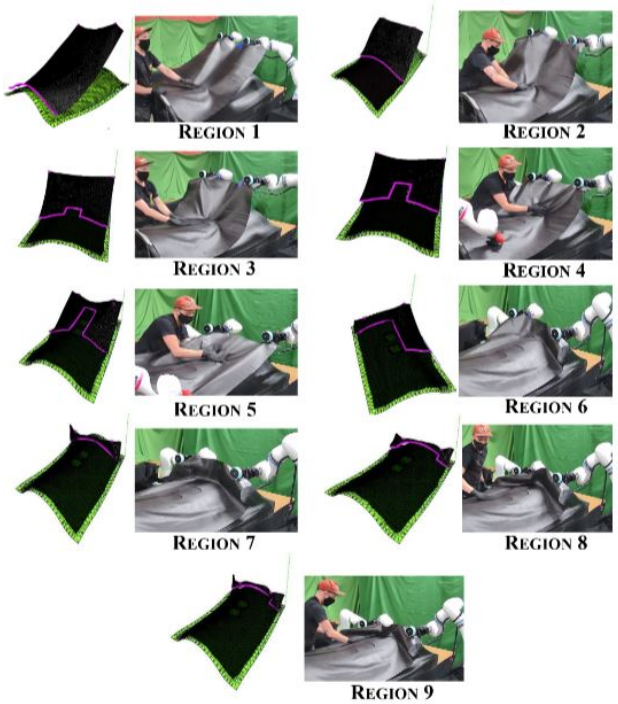


Fig. 7: Grasping positions for the 9 draping zones in simulation and physical setup for Part A.

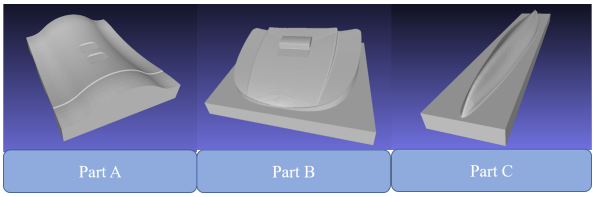


Fig. 8: The three molds on which the grasp planner was tested. These molds vary in terms of complexity of surface features and the draping strategy.

The sheet is tracked in real-time by employing a sheet tracking system which comprises of three RealSense D415 sensors. The primary function of this system is to generate a filtered point cloud of the undraped composite sheet at each grasping location. The raw point cloud data of the prepreg from each of the three Realsense D415 sensors is filtered and merged to create a unified tracked point cloud of the composite sheet. This data can then be utilized to draw a comparison between the simulated data and the observed data.

B. Constraint Violation Monitoring

In order to detect anomalous behavior, we employ a similar constraint satisfaction methodology as discussed in section IV. We record the prepreg's point-cloud $P(t)$ at time t using the RealSense sensors. We perform constraint satisfaction on $P(t)$ for the proximity, collision and alignment constraints by using a pre-recorded point-cloud of the mold. For monitoring the Elastic Energy, we measure the force and torque experienced by the manipulators grasping the sheet.

During the planning, we archive the simulation data of the prepreg at each of the feasible state in our optimal grasp plan Ω. This data consists of the constraint values for the prepreg's simulated model at every draping zone i. We

compare this data against the constraints evaluated from real-time sheet tracking data to monitor and detect any constraint violations. The overall process flow of constraint monitoring is depicted in Fig. 9. We compute the error between the corresponding values of constraints for simulated data and the point-cloud $P(t)$. This error is a measure of deviation of the sheet behavior at a draping zone i. If the value of error exceeds a certain threshold, we trigger appropriate actions to take corrective measures (see section V-C).

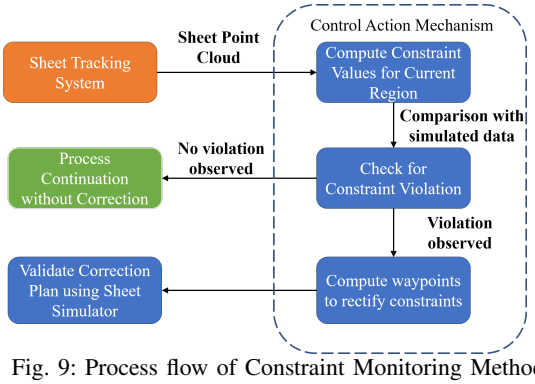


Fig. 9: Process flow of Constraint Monitoring Method.

C. Control Actions

The intervention controller executes certain control actions based on the magnitude of the error defined in section. V-B. As discussed earlier, variations in material parameter model of the prepreg impact the value of the error. We typically classify the required level of intervention into three cases, depending on the level of inaccuracy in the material parameter model (see Fig. 10).

Intervention scenarios include the following:

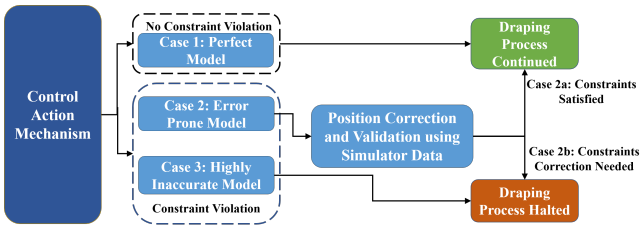


Fig. 10: Process flow of our Intervention Controller.

- *Case 1: Accurate Material Parameter Model* In this case, the generated plans observe minimal deviations. Our grasp planner will generate feasible plans without a need for intervention.
- *Case 2: Error-Prone Material Parameter Model:* This case occurs when there is a minor discrepancy in the material parameter model. Based on the errors encountered in each of the constraints described in Section. V-B we take appropriate actions. For e.g. when Droop Factor is violated, we move the robots towards the extermity of the prepreg along the grasped edge by a value proportional to the error.
- *Case 3: Highly Inaccurate Material Parameter Model:* In this case due major discrepancy in material model, we experience incorrigible deviations from the desirable locations. Consequently, a halt condition is triggered.

D. Results

We tested grasp planning methodology on the Part A (see Fig. 8). The composite layup cell shown in Fig. 1 was used for the experiments. In case 1, we generated grasp plans for Part A with an accurate material parameter model (see Fig. 11). In this case, the intervention controller reported no violations in the process constraints. In the second case, we artificially introduced an error of 10% in the material parameter model. Due to this inaccuracy, we observed violations in the constraints on comparing the prepreg point cloud data and the stored simulation values. As the error encountered was within an acceptable threshold, our intervention controller successfully computed new grasping locations. At the new grasping locations, we validate the constraints by simulating the new position and comparing it with the sheet point cloud. We observed that the error at this updated location was within the acceptable limit and hence the remaining plan was executed with intermittent interventions. These intervention occurred for all 9 draping stages. In the third case, we the material model error was set to 20% which resulted in triggering of the halting condition by the intervention controller.

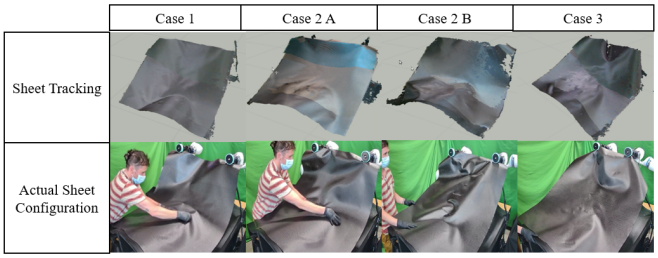


Fig. 11: Comparison between different cases for material parameter model. Row 1 depicts the simulated data for the four control action cases, and Row 2 depicts the Real Sheet Configuration of the corresponding data.

VI. Conclusions

We demonstrated how to automatically generate grasp plans for draping of large sheets over complex molds. We did this using a novel state-space search formulation that incorporates physically based sheet simulation to model realistic sheet deformations. We successfully executed the generated grasp plans in a real-world physical setup. We then showcased an intervention controller that can monitor the task execution using a sheet tracking system and prevent task failure.

The current grasp planning formulation is for two-robot setup. In the future, we plan to generalize it to handle n-robot setups. This will enable us to handle larger sheets. The current algorithm does not update the material model based on the observed performance. In the future, we plan to update material parameters in real-time, enabling us to modify grasp plans online and prevent process interruptions.

Acknowledgment: This work is supported in part by National Science Foundation Grant #1925084. Opinions expressed are those of the authors and do not necessarily reflect opinions of the sponsors.

REFERENCES

[1] R. K. Malhan, A. M. Kabir, A. V. Shembekar, B. Shah, T. Centea, and S. K. Gupta, "Hybrid cells for multi-layer prepreg composite sheet layup," in *IEEE International Conference on Automation Science and Engineering (CASE)*, Munich, Germany, Aug 2018.

[2] R. K. Malhan, A. M. Kabir, B. Shah, T. Centea, and S. K. Gupta, "Automated prepreg sheet placement using collaborative robotics," in *North America Society for the Advancement of Material and Process Engineering (SAMPE) Long beach conference*, CA, USA, 2018.

[3] R. K. Malhan, A. V. Shembekar, A. M. Kabir, P. M. Bhatt, B. Shah, S. Zanio, S. Nutt, and S. K. Gupta, "Automated planning for robotic layup of composite prepreg," *Robotics and Computer-Integrated Manufacturing*, vol. 67, p. 102020.

[4] R. K. Malhan, A. M. Kabir, B. Shah, and S. K. Gupta, "Identifying feasible workpiece placement with respect to redundant manipulator for complex manufacturing tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5585–5591.

[5] R. K. Malhan, A. M. Kabir, B. Shah, T. Centea, and S. K. Gupta, "Determining feasible robot placements in robotic cells for composite prepreg sheet layup," in *ASMEs 14th Manufacturing Science and Engineering Conference*, Erie, PA, USA, June 2019.

[6] R. K. Malhan, R. Jomy Joseph, A. V. Shembekar, A. M. Kabir, P. M. Bhatt, and S. K. Gupta, "Online grasp plan refinement for reducing defects during robotic layup of composite prepreg sheets," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, May 2020.

[7] J. Barbič, F. S. Sin, and D. Schroeder, "Vega fem library," 2012, http://www.jernejbarbic.com/vega.

[8] S. G. Hancock and K. D. Potter, "The use of kinematic drape modelling to inform the hand lay-up of complex composite components using woven reinforcements," *Composites Part A: Applied Science and Manufacturing*, vol. 37, no. 3, pp. 413 – 422, 2006.

[9] G. Newell and K. Khodabandehloo, "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*, vol. 209, no. 6, pp. 423–432, 1995.

[10] C. Krogh, J. A. Glud, and J. Jakobsen, "Modeling the robotic manipulation of woven carbon fiber prepreg plies onto double curved molds: A path-dependent problem," *Journal of Composite Materials*, vol. 53, no. 15, pp. 2149–2164, 2019.

[11] M. Elkington, D. Bloom, C. Ward, A. Chatzimichali, and K. Potter, "Hand layup: understanding the manual process," *Advanced Manufacturing: Polymer & Composites Science*, vol. 1, no. 3, pp. 138–151, 2015.

[12] R. O. Buckingham and G. C. Newell, "Automating the manufacture of composite broadgoods," *Composites Part A: Applied Science and Manufacturing*, vol. 27, no. 3, pp. 191 – 200, 1996.

[13] R. Molfino, M. Zoppi, F. Cepolina, J. Yousef, and E. E. Cepolina, "Design of a hyper-flexible cell for handling 3d carbon fiber fabric," *Recent advances in mechanical engineering and mechanics*, vol. 165, 2014.

[14] M. Elkington, C. Ward, and K. D. Potter, "Automated layup of sheet prepregs on complex moulds," in *SAMPE Long Beach Conference*, 2016.

[15] G. Seliger, F. Szimmat, J. Niemeier, and J. Stephan, "Automated handling of non-rigid parts," *CIRP Annals*, vol. 52, no. 1, pp. 21–24, 2003.

[16] G. Fantoni, M. Santochi, G. Dini, K. Tracht, B. Scholz-Reiter, J. Fleischer, T. K. Lien, G. Seliger, G. Reinhart, J. Franke *et al.*, "Grasping devices and methods in automated production processes," *CIRP Annals*, vol. 63, no. 2, pp. 679–701, 2014.

[17] A. Björnsson, M. Jonsson, and K. Johansen, "Automated material handling in composite manufacturing using pick-and-place systems – a review," Jan 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584517301758

[18] M. Elkington., C. Ward, and A. Sarkytbayev, "Automated composite draping: a review," in *SAMPE*. SAMPE North America, May 2017.

[19] Z. Peng and L. Yuanchun, "Position/force control of two manipulators handling a flexible payload based on finite-element model," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Dec 2007, pp. 2178–2182.

[20] P. Zhang and Y. c. Li, "Simulations and trajectory tracking of two manipulators manipulating a flexible payload," in *IEEE Conference on Robotics, Automation and Mechatronics*, Sept 2008, pp. 72–77.

[21] D. Tzeranis, Y. Ishijima, and S. Dubowsky, "Manipulation of large flexible structural modules by space robots mounted on flexible structures," *Proc. Int. Sym. on Artificial Intelligence, Robotics and Automation in Space*, 2005.

[22] J. Das and N. Sarkar, "Autonomous shape control of a deformable object by multiple manipulators," *Journal of Intelligent & Robotic Systems*, vol. 62, no. 1, pp. 3–27, Apr 2011.

[23] D. Henrich and H. Wörn, *Robot Manipulation of Deformable Objects*. Springer Science & Business Media, 2012.

[24] F. Ding, J. Huang, Y. Wang, T. Matsuno, and T. Fukuda, "Vibration damping in manipulation of deformable linear objects using sliding mode control," *Advanced Robotics*, vol. 28, no. 3, pp. 157–172, 2014.

[25] J. Alonso-Mora, R. Knepper, R. Siegwart, and D. Rus, "Local motion planning for collaborative multi-robot manipulation of deformable objects," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5495–5502.

[26] D. Kruse, R. J. Radke, and J. T. Wen, "Human-robot collaborative handling of highly deformable materials," in *American Control Conference (ACC)*, May 2017, pp. 1511–1516.

[27] P. Tsarouchi, J. Spiliotopoulos, G. Michalos, S. Koukas, A. Athanasatos, S. Makris, and G. Chryssolouris, "A decision making framework for human robot collaborative workplace generation," *Procedia CIRP*, vol. 44, pp. 228–232, 2016.

[28] G. G. Gunnarsson, O. W. Nielsen, C. Schlette, and H. G. Petersen, "Fast and simple interacting models of drape tool and ply material for handling free hanging, pre-impregnated carbon fibre material," in *International Conference on Informatics in Control, Automation and Robotics*. Springer, 2018, pp. 1–25.

[29] F. F. Khalil and P. Payeur, "Dexterous robotic manipulation of deformable objects with multi-sensory feedback-a review," in *Robot Manipulators Trends and Development*. InTech, 2010.

[30] P. Jiménez, "Survey on model-based manipulation planning of deformable objects," *Robotics and computer-integrated manufacturing*, vol. 28, no. 2, pp. 154–163, 2012.

[31] M. Moll and L. E. Kavraki, "Path planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 625–636, 2006.

[32] M. Saha and P. Isto, "Manipulation planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1141–1150, Dec 2007.

[33] O. Roussel, A. Borum, M. Taix, and T. Bretl, "Manipulation planning with contacts for an extensible elastic rod by sampling on the submanifold of static equilibrium configurations," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3116–3121.

[34] A. Papacharalampopoulos, S. Makris, A. Bitzios, and G. Chryssolouris, "Prediction of cabling shape during robotic manipulation," *The International Journal of Advanced Manufacturing Technology*, vol. 82, no. 1-4, pp. 123–132, 2016.

[35] M. Saha and P. Isto, "Motion planning for robotic manipulation of deformable linear objects," in *IEEE International Conference on Robotics and Automation*, May 2006, pp. 2478–2484.

[36] H. Wakamatsu, E. Arai, and S. Hirai, "Knotting/unknotting manipulation of deformable linear objects," *The International Journal of Robotics Research*, vol. 25, no. 4, pp. 371–395, 2006.

[37] A. M. Ladd and L. E. Kavraki, "Using motion planning for knot untangling," *The International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 797–808, 2004.

[38] T. Matsuno, T. Fukuda, and F. Arai, "Flexible rope manipulation by dual manipulator system using vision sensor," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings (Cat. No.01TH8556)*, vol. 2, 2001, pp. 677–682 vol.2.

[39] D. Berenson, "Manipulation of deformable objects without modeling and simulating deformation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 4525–4532.

[40] Y. Li, Y. Yue, D. Xu, E. Grinspun, and P. K. Allen, "Folding deformable objects using predictive simulation and trajectory optimization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2015, pp. 6000–6006.

[41] A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrík, A. Kargakos, L. Wagner, V. Hlaváč, T. K. Kim, and S. Malassiotis, "Folding clothes autonomously: A complete pipeline," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1461–1478, Dec 2016.

[42] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection based on multiple-view geometric cues with

application to robotic towel folding," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 2308–2315.
[43] J. Van Den Berg, S. Miller, K. Goldberg, and P. Abbeel, "Gravity-based robotic cloth folding," in *Algorithmic Foundations of Robotics IX*. Springer, 2010, pp. 409–424.
[44] P. N. "Koustoumpardis, K. I. Chatzilygeroudis, A. I. Synodinos, and N. A. Aspragathos, ""human robot collaboration for folding fabrics based on force/rgb-d feedback"," in *"Advances in Robot Design and Intelligent Control"*. "Springer International Publishing", "2016", pp. "235–243".
[45] S. Miller, M. Fritz, T. Darrell, and P. Abbeel, "Parametrized shape models for clothing," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4861–4868.
[46] S. Miller, J. van den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 249–267, 2012.
[47] Y. Bai, W. Yu, and C. K. Liu, "Dexterous manipulation of cloth," *Computer Graphics Forum*, vol. 35, no. 2, pp. 523–532, 2016.
[48] J. Schulman, J. Ho, C. Lee, and P. Abbeel, *Learning from Demonstrations Through the Use of Non-rigid Registration*. Cham: Springer International Publishing, 2016, pp. 339–354.
[49] D. McConachie and D. Berenson, "Bandit-based model selection for deformable object manipulation," *arXiv preprint arXiv:1703.10254*, 2017.
[50] D. McConachie, M. Ruan, and D. Berenson, "Interleaving planning and control for deformable object manipulation," in *International Symposium on Robotics Research (ISRR)*, 2017.
[51] N. Koganti, T. Tamei, K. Ikeda, and T. Shibata, "Bayesian nonparametric learning of cloth models for real-time state estimation," *IEEE Transactions on Robotics*, vol. 33, no. 4, pp. 916–931, Aug 2017.
[52] S. Flixeder, T. Glück, and A. Kugi, "Modeling and force control for the collaborative manipulation of deformable strip-like materials," *IFAC-PapersOnLine*, vol. 49, no. 21, pp. 95 – 102, 2016, 7th IFAC Symposium on Mechatronic Systems MECHATRONICS 2016.
[53] D. Kruse, R. J. Radke, and J. T. Wen, "Collaborative human-robot manipulation of highly deformable materials," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 3782–3787.
[54] Z. Hu, P. Sun, and J. Pan, "Three-dimensional deformable object manipulation using fast online gaussian process regression," *Robotics and Automation Letters*, vol. 3, no. 2, pp. 979–986, 2018.
[55] P.-C. Yang, K. Sasaki, K. Suzuki, K. Kase, S. Sugano, and T. Ogata, "Repeatable folding task by humanoid robot worker using deep learning," *Robotics and Automation Letters*, vol. 2, no. 2, pp. 397–403, 2017.
[56] O. C. Zienkiewicz, *The Finite Element Method*. Maidenhead, Berkshire, England: McGraw-Hill Book Company (UK) Limited, 1977.
[57] P. Volino, N. Magnenat-Thalmann, and F. Faure, "A simple approach to nonlinear tensile stiffness for accurate cloth simulation," 2009.
[58] H. Wang, J. F. O'Brien, and R. Ramamoorthi, "Data-driven elastic models for cloth: modeling and measurement," *ACM Transactions on Graphics (SIGGRAPH 2011)*, vol. 30, no. 4, pp. 71:1–71:12, 2011.
[59] A. X. Lee, S. H. Huang, D. Hadfield-Menell, E. Tzeng, and P. Abbeel, "Unifying scene registration and trajectory optimization for learning from demonstrations with application to manipulation of deformable objects," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2014, pp. 4402–4407.
[60] A. X. Lee, H. Lu, A. Gupta, S. Levine, and P. Abbeel, "Learning force-based manipulation of deformable objects from multiple demonstrations," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 177–184.
[61] S. H. Huang, J. Pan, G. Mulcaire, and P. Abbeel, "Leveraging appearance priors in non-rigid registration, with application to manipulation of deformable objects," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2015, pp. 878–885.
[62] T. Yoshikawa, "Manipulability of robotic mechanisms," *The International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985. [Online]. Available: https://doi.org/10.1177/027836498500400201
[63] J. Pan, S. Chitta, and D. Manocha, "Fcl: A general purpose library for collision and proximity queries," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 3859–3866.