Preprint version; Final version at https://ieeexplore.ieee.org/document/10611478
2024 IEEE International Conference on Robotics and Automation (ICRA)
DOI: 10.1109/ICRA57147.2024.10611478

Model Predictive Control with Graph Dynamics for Garment Opening Insertion during Robot-Assisted Dressing

Stelios Kotsovolis and Yiannis Demiris

***Abstract*— Robots have a great potential to help people with movement limitations in activities of daily living, such as dressing. A common problem in almost all dressing tasks is the insertion of a garment's opening around a part of the human body. The rich contact environment and the deformations of the garment make the task a challenging problem for robots. In this paper, we propose a bi-manual control method for garment opening insertion during robot-assisted dressing. Specifically, we propose a model predictive controller that uses an Attention-based Relational Graph Convolutional Network (ARGCN) for modeling the dynamics of the opening in the presence of the body. We train the model entirely in simulation and validate our method in four real-world dressing scenarios of a medical training manikin. We show that our method generalizes well in the real-world opening insertion tasks achieving an overall success rate of 97.5%, even though the dynamics and the shapes vastly differ from the simulation setup.**

I. Introduction

Assistive technologies are becoming increasingly important as the population ages and the number of people with disabilities increases. Specifically, dressing is reported as one of the most onerous activities for healthcare workers [1]. Assistive robots could provide assistance with dressing, as well as provide privacy and agency to the users. However, dressing remains a very challenging task for robots, as a plethora of different problems needs to be solved, from fine manipulation of garments to tracking of human motions.

Recent studies have made great progress in the field towards tackling these challenges [2-10]. Most of the dressing-related approaches mainly focus on either the manipulation after the insertion of the opening or on the preparation for it, while the insertion stage has received less attention. Regarding the latter, although studied in some specific tasks [11-13], strong assumptions were made about the shape of the body or the dynamics of the garment, making the generalization to different garments or body parts hard.

The insertion of a garment's opening is a problem present across almost all dressing scenarios. Motivated by that fact, in this paper we study the insertion stage of dressing, to build towards generalization for different dressing scenarios. This stage of dressing is particularly complex because the garment is not yet constrained on the body. For this reason, we use a bi-manual manipulation method to have more control over the garment, similar to how humans usually perform this stage. A fundamental challenge in this scenario is the rich contact environment which results in garment-body collisions. During the insertion stage of dressing, contacts between the garment and the human body parts are especially important, as they can cause failure if the garment gets caught, or they can be beneficial and used to create hooks that can help complete the task.

Fig. 1: The proposed method enables the Baxter robot to successfully insert the openings of garments in four dressing scenarios

To enhance robots with such abilities, we propose a model predictive controller that uses a learned graph dynamics model of the opening of a garment in the presence of the body. In particular, we use an Attention-based Relational Graph Convolutional Network (ARGCN) [14, 15] to model the dynamics. To generalize for the shape of body parts, we use the point clouds of the body as input, while we use the point cloud of the opening as the garment's representation. Our model is trained entirely in simulation. More specifically, we use a ring-shaped piece of cloth to simulate the opening and learn its dynamics when in contact with general shapes (cylinders, spheres, and boxes). We show in our experiments that our method can effectively succeed in real-world dressing scenarios, despite the different shapes and dynamics, proving its generalization properties. In this paper, we focus on the insertion stage, up to the point where the opening is securely positioned around the targeted body part for dressing. However, our method can potentially be complemented by another dressing method to complete the

The authors are with the Personal Robotics Lab, Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2BT, U.K. (e-mail:s.kotsovolis21@imperial.ac.uk; y.demiris@imperial.ac.uk). This research is supported in part by UKRI Grant EP/V026682/1, and a Royal Academy of Engineering Chair in Emerging Technologies to YD. For the purpose of open access, the author(s) has applied a Creative Commons Attribution (CC BY) license to any Accepted Manuscript version arising.

task [6, 16, 17]. Overall, the contributions of this paper can be summarized as follows:

- A model predictive controller for insertion of garment openings during robot-assisted dressing that can generalize across different garment dynamics and body part shapes.
- A dynamics model to predict future states of the opening of a garment under external contacts.
- An experimental evaluation against state-of-the-art baselines of the dynamics model in simulation as well as the overall system in four real-world dressing tasks, depicted in Fig. 1: Putting a hospital gown's neck opening around the head, a scarf around the head, a hospital gown's sleeve opening around the hand and a sleeveless jacket's opening around the hand.

II. Related Work

A. Robot-assisted Dressing

Robot-assisted dressing studies can be classified as studies about perception and state representation, studies about user behaviour modelling and personalization, and studies about manipulation of the garments themselves. Regarding the first category, some prior works focus on perceiving important parts of the garments, mainly the openings, and building representation models [2, 3, 18-20]. Other studies focus on finding grasping points and/or unfolding the garment in preparation for a dressing task [7, 8, 21, 22]. Several papers study the case of a collaborative user and propose approaches to track the human's pose under occlusions [23-25], using personalization techniques according to the user's preferences or movement limitations [16, 26-32].

On the other hand, many prior studies assume, as we also do, users without the ability to move. The objective of these studies is to achieve a desired state of the garment around the body. Some works study the problem of dressing a sleeve of a garment on an arm [4, 5, 17]. Recently, a study on that problem [6] used directly the point cloud to generalize for different poses. These methods focus more on the manipulation process after the insertion stage specifically for arm dressing and use single-arm manipulators. A few prior studies involved the insertion of the opening, for instance, when putting a T-shirt's neck around the head [11, 12] or during upper body dressing, with a T-shirt or a sleeveless jacket [13, 33]. However, they assume a specific body shape and use approximate representations, such as the centroid points. Instead, we propose a method for the initial insertion of garments by using the point clouds of the opening and the body, to generalize for different body parts and openings.

B. External Contacts in Deformable Objects Manipulation

Deformable object manipulation remains a challenging task for robots, mainly because of the high space state and the difficulty in dynamics modeling, especially when external contacts are present. Some approaches handle the contacts without directly modeling the dynamics [34, 35]. Others, propose primitive contact types [36], adaptation of the dynamics [37], or task-specific dynamics [38]. Following the promising results of Graph Neural Networks in modeling the dynamics of deformable objects [39-42], the authors of [43] proposed a Graph Convolutional Network for linear deformable object modeling with external contacts, with different kinds of graph edges for different types of interactions. In our work, we borrow the idea of the relational structure of the network. However, our approach differs in the dynamic model's architecture, as well as the complexity of the task.

The problem of insertion of deformable objects has received less attention. A FEM-based optimization method was presented in [44] for placing a ring-shaped object around a pillar, but the method was validated only in simulation. In [45] a heuristic planner was presented for the same problem, while an approach for driving units assembly was proposed in [46]. In [47], in which reinforcement learning was applied in different simulation environments, the authors reported unsatisfying results in a task where a rubber band had to be placed around pillars. These studies focus specifically on insertions around circular objects, while for the ones validating the method in real-world experiments, key points are used to represent the deformable object. To our knowledge, there is currently no method for the insertion of deformable objects that has demonstrated generalization for the shape of the target object. Our method can generalize for different body parts, as it successfully completed garment insertions around a hand and a head, even though the dynamics are learnt from interactions with general geometric shapes.

III. Problem Formulation and Assumptions

In our formulation of the problem of opening insertion during robot-assisted dressing, we make the following assumptions. In each time step t of the dressing task, we assume a point cloud of the garment's opening P_o^t and a point cloud of the targeted body part for dressing P_b^t. We leave the case of partial observations for future work. Here, we study a static user's case and we assume that P_b^t remains the same during the process. Hence, in the rest of this paper, we refer to it simply as P_b. We also assume that the robot has grasped the opening in two diametrically opposed points with two generic grippers. Lastly, we use low velocities and accelerations to meet the quasi-static condition.

The general goal of our method is to select actions a^t of the robot end-effectors, to place P_o around P_b. As presented in Fig. 2, we define the action of the robot and the state of the opening and the body part as a graph G^t. An Attention-based Graph Convolutional Network (ARGCN) is used to process the graph and make predictions about the next state of the garment's opening $\hat{P}_o^{t+1}$. The prediction model is then utilized in a model predictive controller to take actions that minimize a cost function $C(\hat{P}_o^{t+1}, P_b)$. In the following sections, we describe our method in detail.

IV. Dynamics Model

A. Graph Construction

Using different types of edges to represent the relations between the deformable objects and the contacts has been shown beneficial for dynamics modeling in the presence of external contacts in previous research [43]. We borrow this

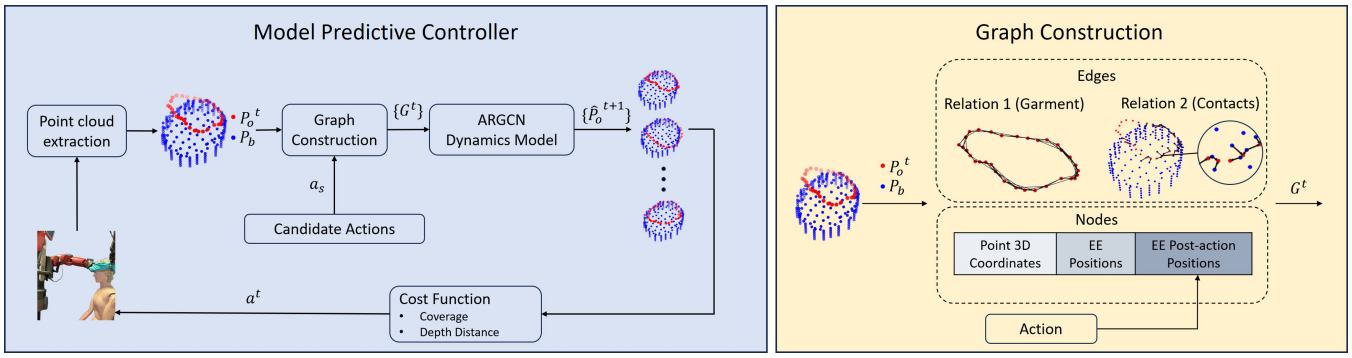

Fig. 2: Model predictive controller (MPC) architectural diagram. (Left): In each time step, the point clouds of the body and the opening of the garment (state) are converted into a graph. Candidate actions are sampled and a graph is generated for each state-action pair. For each candidate action a prediction of the next step's point cloud of the opening is made by the ARGCN model and the optimal action is selected according to a cost function. (Right): The point clouds are converted into a relational graph, by constructing bidirectional edges between the nodes of the opening (deformation dynamics) and directional edges from the body's nodes to the opening's nodes (contact dynamics). The actions are encoded in the graph by concatenation in the feature space.

idea and adjust it for our garment opening insertion task. Specifically, we structure our graph as $G^t = (V, E^r)$, where V are the nodes of the graph representing the 3D positions of the two point clouds' points and E^r are the edges of the graph under the relation r. In our work, we use 2 types of edges: bi-directional edges between the nodes of the opening of the garment P_o^t and directional edges from the body part's nodes P_b towards the nodes of the opening of the garment P_o^t. The first relation aims to model the dynamics of the garment's opening itself, while the second aims to model the contact dynamics with the body. The edges, represented by the adjacency matrices E^r, are constructed based on a connectivity radius. We encode the action by concatenating the end-effectors' positions P_e^t as well as the post-action end-effectors' positions of the next state P_e^{t+1}, in the feature space of the nodes of the graph, similar to [42, 48].

B. Network Architecture

The goal of our dynamics model is to provide an estimate of the opening's point cloud in the next time step $\hat{P}_o^{t+1}$, given the current state and an action, encoded by the graph G^t. We propose using an Attention-based Relational Graph Convolutional Network (ARGCN) [14, 15], which extends Relational Graph Convolutional Networks (RGCNs) [49], by using attention mechanisms, instead of fixed normalization constants. Relational GCNs are well-suited for processing relational graphs by using separate aggregation functions for each relation, different types of dynamics in our case. Moreover, the attention mechanism can be beneficial by focusing on the interactions, that are important for making predictions. We describe in this section our specific architecture.

We use a typical encoder-decoder architecture with linear weights as our encoder and decoder, and an ARGCN network as the core processor. Our formulation of the ARGCN follows the general formulation for each layer:

$$h_i^{'} = \sigma\Big(\sum_{r \in R} \sum_{j \in N_i^r} x_{ij,r} + W_0 h_i\Big) \tag{1}$$

where h_i and $h_i^{'}$ are the input and output representations of the node i in the layer, N_i^r is the set of neighbors of the node i under the relation r, R is the set of relations (2 in our case), W_0 is a trainable weight, σ is the activation function and $x_{ij,r}$ describes the aggregation between neighbors of each relation. We use different weights for each layer.

We further use a multi-head attention mechanism [50] as:

$$x_{ij,r} = W_{b_r}[x_{ij,r}^1 || x_{ij,r}^2 || ... || x_{ij,r}^K] \tag{2}$$

where K is the number of attention heads. The outputs of the attention heads are concatenated (denoted here by the symbol $||$) and passed through a trainable weight W_{b_r}, while for each attention head k, an attention mechanism is used:

$$x_{ij,r}^k = a_{ij,r}^k W_r^k h_j \tag{3}$$

where W_r^k is a trainable weight, different for every attention head k and relation r and

$$a_{ij,r}^k = \frac{\sigma(\alpha_{k,r}^T [W_r^k h_i || W_r^k h_j])}{\sum_{m \in N_i^r} \sigma(\alpha_{k,r}^T [W_r^k h_i || W_r^k h_m])} \tag{4}$$

where $\alpha_{k,r}$ is a trainable weight. In our implementation, we use different weights per relation and attention head. We use as a dimension of each attention head as $d_k = d_i/K$, where d_i is the dimension of the input's representation, as in [50], to keep the dimension fixed between the layers.

After L layers of the described message propagation function, the final layer's representation is decoded to acquire an estimate of the opening's point cloud in the next time step $\hat{P}_o^{t+1}$. We use $L = 8$ layers, $K = 4$ attention heads, a ReLU activation function and a latent space size of 64.

V. Simulation and Training

To collect data for our model, we built a simulation environment in PyBullet, [51], a FEM-based simulator, as depicted in Figure 3. A ring-shaped cloth object of $1m$ radius with a small thickness was used to simulate a garment's opening. Three general geometric objects (a cylinder, a sphere, and a rectangular box) were used to get data from interactions of the opening with other objects. Their dimensions and orientation were randomized during the rollouts and the center of their position was used to center the scene

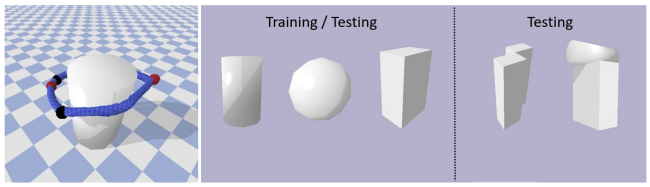

Fig. 3: We use a simulation setup in PyBullet with a cloth ring-shaped object to record interactions with objects of general geometric shapes. Two spheres (red), attached to the cloth, perform random actions, while force injection spheres (black) are used to simulate forces applied by the rest of the garment.

in the real-world dressing application. Similarly, the initial position and orientation of the opening were randomized, while the size of the opening remained fixed and used as a scaling factor for the real-world dressing application.

We recognize that in real-world dressing tasks, the rest of the garment influences the opening as it applies forces on it, because of its weight and friction with other parts of the body. Still, the opening remains the most important part of the garment for the insertion stage, while the rest of it affects the opening by applying forces on it. For this reason, instead of using the whole garment in simulation, for which complicated interactions with the body are difficult to simulate, we directly apply small random forces in random vertices of the opening's mesh. Under the assumption that forces remain roughly the same between small time steps, we maintain the forces fixed during rollouts. This method sometimes referred to as force injection in sim-to-real studies [52], randomizes the dynamics and is suitable in our case because the randomization is uneven across the opening.

The 2 end-effectors of the robot were simulated by 2 spheres anchored on random, but diametrically opposed vertices of the opening's mesh. We performed small random displacements of the end-effectors of a small fixed size $\delta = 0.3m$. We gathered 33,400 data tuples (state, action, and next state) for training. In simulation, we capture the point cloud directly from equally distributed vertices of the meshes and since we know the correspondence in time, we trained our model with a Mean Squared Error (MSE) loss.

VI. Model Predictive Controller

To generate actions, we propose a model predictive controller that uses the learned ARGCN model and a cost function that encourages the insertion of the opening of the garment around the body. In each time step, the controller generates random candidate actions of fixed step δ. From these actions, we discard the ones that result in the end-effector coming closer to the body than a set threshold. Consequently, the candidate actions, along with the state of the environment represented by the two point clouds, are utilized to construct a graph for each action, as described in section IV-A. These graphs are then processed by our dynamics model to generate a prediction $\hat{P}_o^{t+1}$ of the opening's point cloud for each candidate action. The controller then selects the action that minimizes the following cost function:

$$C(\hat{P}_o^{t+1}, P_b) = C_{prog}(\hat{P}_o^{t+1}) + w * C_{cov}(\hat{P}_o^{t+1}, P_b) \quad (5)$$

Fig. 4: Snapshots during four real-world dressing scenarios. Our controller successfully inserts the openings by taking advantage of the contacts between the garment and the body.

The cost function is constructed by two terms and a tunable weight w. The first term encourages the progression of the task by bringing the opening deeper around the body's axis z, using the average distance from the desired depth:

$$C_{prog}(\hat{P}_o^{t+1}) = \frac{\sum_{i=1,..,N_o} |\hat{P}_{o_i,z}^{t+1} - p_{d,z}|}{N_o} \quad (6)$$

where $\hat{P}_{o,z}^{t+1}$ is the predicted garment's opening, projected on a plane perpendicular to the axis of the body, N_o is the number of points in the opening's point cloud and $p_{d,z}$ is the desired depth. With the second term, we aim to maximize the coverage area of the opening around the body part:

$$C_{cov}(\hat{P}_o^{t+1}, P_b) = (1 - N_{b_{in}}/N_b) \quad (7)$$

where N_b is the number of points of the body's point cloud (with a uniform density) when projected on a plane perpendicular to the axis of the body part, and $N_{b_{in}}$ is the number of those points that are inside of the area shaped by $\hat{P}_o^{t+1}$, projected on the same plane. Finally, in each time step the controller selects the action a^t, that minimizes the cost:

$$a^t = \underset{a_s}{\operatorname{argmin}}\, C(\hat{P}_o^{t+1}, P_b) \quad (8)$$

We use one prediction step, as preliminary experiments did not show any significant improvement in the performance when a bigger horizon was used. On the contrary, it heavily increased the processing time between actions. As we show in our results, the proposed controller achieved high success rates, even with one prediction step.

VII. Evaluation

We conduct experiments to investigate the performance of our dynamics model and the overall system's performance.

A. Dynamics Model Evaluation

To evaluate the dynamics model we conduct experiments in simulation and compare our model against 3 baselines. We collect 8000 new data tuples for the three objects used during training and another 8000 data tuples with 2 objects unseen during training: an object constructed by 2 boxes connected diagonally and a T-like object, as shown in Fig.

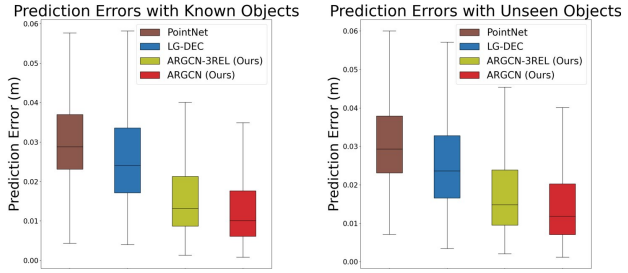

Fig. 5: Results of the error between the prediction and the actual point clouds of the opening, using the proposed method and baselines. Our model is more accurate and robust compared to the baselines, as it achieves smaller errors and variance.

3. We compare our proposed Attention-based Relational Graph Convolutional Network (ARGCN) against the following baselines:

- ARGCN-3REL: We use our dynamics model, but instead of a feature space action encoding, we encode the actions geometrically by using a third relation in the graph, similar to the action encoding method from [43].
- LG-DEC: We compare against the network architecture from [43], which was proposed for 2D deformable object dynamics modeling in the presence of external contacts.
- PointNet: Since we use the point clouds as inputs and outputs, we compare against a segmentation-type PointNet [53]. For a more fair comparison, we use two separate branches for processing the point clouds of the opening and the body, which are concatenated along with the action in the global feature, after the PointNet's pooling layer.

From Fig. 5 we can draw the following conclusions. Firstly, in our case, the feature action encoding resulted in slightly better performance, than when encoded as a third relationship. We believe this happens because, in the case of a geometrical action encoding, the information of the end-effectors' positions before the action is not preserved in the graph, if nodes of the opening not attached to the end-effector enter the connectivity radius of the corresponding relation. In our task, this seems to be often the case, probably because of the circular shape of the opening, leading to inferior performance of the geometric action encoding. From the comparison with the two last baselines, we prove the superior performance of our proposed dynamics model against the two state-of-the-art networks. Moreover, our network is more robust, since the variance of the errors is smaller.

B. Model Predictive Control Evaluation

To evaluate the overall control process for the opening insertion during robot-assisted dressing, we conduct real-world experiments with a Baxter humanoid robot equipped with two 2-finger Robotiq grippers. For safety reasons, we use a medical training manikin with an open-palm hand configuration. The manikin's head is designed with human-like hair, significantly increasing friction. We evaluate our method against 2 baselines in 4 different dressing scenarios that include 2 different body shapes and 4 garments of different sizes and deformation properties:

- Scenario 1: The robot has to insert the neck's opening of a hospital gown around the head of the manikin. The rest of the gown is placed around the body of the manikin.
- Scenario 2: The robot has to insert a circular scarf around the manikin's head.
- Scenario 3: The robot has to insert a short sleeve of a hospital gown around the manikin's hand.
- Scenario 4: The robot has to insert the arm's opening of a sleeveless jacket around the manikin's hand.

The manikin is fixated on a chair, while the hand is fixated horizontally for the hand dressing scenarios, although slight deviations are allowed. We compare the baselines on the same settings to secure a fair comparison. We place an L515 LiDAR on top of the head or front of the hand, depending on the task. We capture the point clouds of the body parts using depth segmentation before the beginning of the tasks and extend their contours in the z direction of the camera as an estimation of their full shape. For the opening's point cloud, we used depth and color segmentation and the inner contour of the mask to extract the opening, which is then used along with the depth image, to acquire the point cloud. The point clouds are then downsampled and passed through a Moving Least Squares (MLS) filter to be smoothed. We initialize the task by the two grippers holding the opening, above the head or in front of the hand, in various positions, same for all baselines. We consider the tasks completed when the opening is placed around the body and in some depth, where the insertion task can be assumed to be safely completed. For the head, we set the depth on the ears' height and for the hand on the wrist's height. We compare our method against the following two baselines:

- MPC-Centroid Cost: To investigate the effectiveness of the cost function of the MPC, we substitute the coverage term with a distance between the centroids of the opening and the body, which is a common concept in robot dressing.
- Visual Feedback Controller: To demonstrate the necessity of our controller, we compare it against a visual feedback controller, similar to the logic in [12]. Firstly, the controller aligns the opening with the targeted body part, by using as feedback the mean of the points of the body's point cloud that are out of the area of the opening's point cloud when projected on a plane, perpendicular to the body's axis. When a coverage threshold is reached, the robot follows a linear trajectory towards the goal's depth.

For each scenario and baseline, we perform 10 trials and measure the success rate (120 trials in total). The outcome of the trials is shown in Table I, while snapshots of our method for all scenarios are depicted in Fig. 4. Moreover, to qualitatively demonstrate the efficiency of our method we show in Fig. 6 snapshots during a sample trajectory of the opening for each of the baselines during the most competitive task according to the results: the scarf dressing scenario.

As shown in Table I, our method significantly outperformed the two baselines. Specifically, the visual feedback controller achieved a high performance only in the sleeveless jacket's scenario, in which the opening is wide and stiff. For

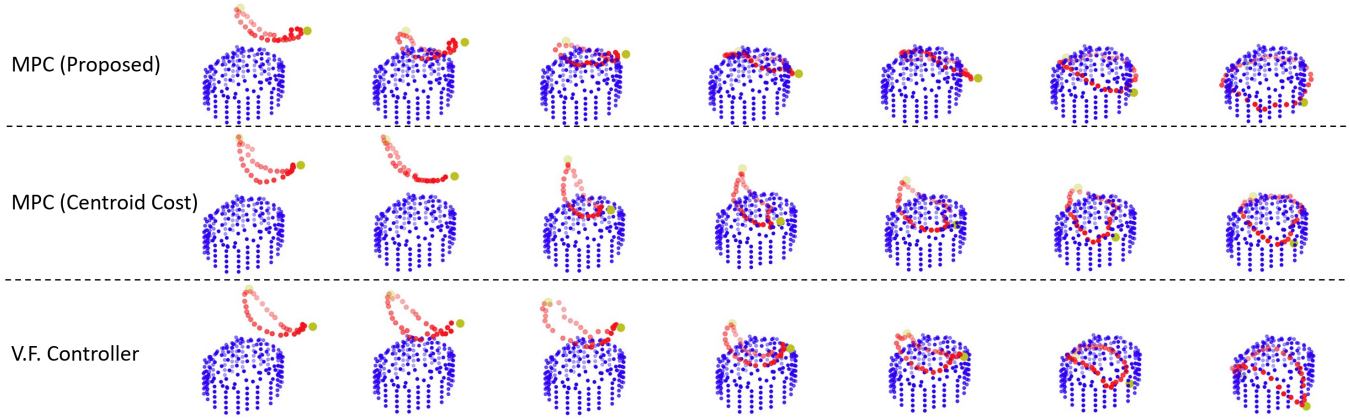

Fig. 6: Trajectories of the point cloud of the opening of the scarf (red) and end-effectors (yellow) around the head (blue) during the real-world scarf dressing task for our method (top) and the two baselines (middle and bottom). The two baselines fail to insert the scarf, which ends up getting caught on the head, because of the smaller size of the scarf compared to the head and the large deformations. On the contrary, our method successfully inserts the scarf around the head by making use of the contacts.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
MPC (Proposed)	**10/10**	**9/10**	**10/10**	**10/10**
MPC (Centroid Cost)	5/10	2/10	3/10	9/10
V.F. Controller	4/10	0/10	3/10	9/10

TABLE I. Success rate of our method and the baselines, during the 4 opening insertion dressing scenarios.

the other cases, this controller resulted in the opening getting caught on the body in most of the trials. In fact, for this baseline, we had to tune the coverage threshold specifically for each task. If the threshold was set too low, the controller would have worse performance in tasks with wide openings, like the jacket, while when it was set too high, it would often get stuck in trying to fully fit the opening and never initiate the linear trajectory, in tasks with narrow openings. Overall, we observed that this controller highly depended on the initial state and the deformation of the opening, while it succeeded consistently only when the opening was stiff and initially wide open. A slightly better performance was achieved by the MPC baseline with a centroid distance cost function. With this baseline, the robot arms could move independently, allowing better control of the garment and the controller did not seem to depend a lot on the initial state. Although the dynamics model provided accurate predictions, the controller led the opening towards the center due to the centroid term of the cost function, which was often caught around the edges of the head, or in the fingers of the hand, mainly the thumb and the pinky finger. To sum up, the two baselines failed in scenarios with garments of high deformations and/or narrow openings.

On the contrary, we observed that our full MPC system encouraged the use of contacts and often created hooks, managing to insert openings even with high deformations and small openings, such as the scarf and the sleeve. Such a case is depicted in Fig. 6 (top) for the scarf dressing scenario. We give the following explanation for this behavior through that example. The robot first tries to achieve a high coverage area and lead the opening towards the center of the head, while also lowering it to achieve lower progression cost. After some point, no more area can be covered and the coverage cost has similar costs for all candidate action. The progression cost, which has now a more important role, leads the opening lower and towards some edge to place a part of the opening lower down the edge, creating a hook. Then, the dynamics model starts predicting states with very high coverage when the actions lead towards the opposite edges, since now the coverage area can grow, as the opening is hooked and not in the air. Finally, after maximizing the area towards the other side, the progression function has again more impact, leading the opening towards the desired depth and completing the task. Similar behavior was observed in the hand scenarios too, which allowed the robot to avoid the opening getting caught in the fingers. The only failure, noticed in the scarf dressing scenario, happened because when placing part of the opening around one side of the head, the scarf slid down creating a very deep hook. Then, when the controller attempted to place the other side of the scarf around the head, the length and elasticity of the scarf were not enough to achieve it. Methods to recover from such cases might be a line of future work.

VIII. Conclusions and Future Work

In this paper, we presented a method for placing an opening of a garment around a targeted body part for dressing. We proposed a model predictive controller with a graph dynamics model for the garment's opening, which achieved higher performance against state-of-the-art methods. Our approach achieved high success rates in four real-world dressing scenarios, showing that our model can be transferred in a zero-shot manner to the real world and generalize for different garments and body parts. In future work, we aim to tackle the problem of partial observations, which is the main assumption made in this study. This will allow the insertion of garments in cases where part of the opening is occluded either from self-occlusion or the body, such as when dressing garments with long sleeves, gloves, or socks.

REFERENCES

[1] T. L. Mitzner, T. L. Chen, C. C. Kemp, and W. A. Rogers, "Identifying the potential for robotics to assist older adults in different living environments," *International journal of social robotics*, vol. 6, no. 2, pp. 213–227, 2014.

[2] L. Twardon and H. Ritter, "Active boundary component models for robotic dressing assistance," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 2811–2818.

[3] 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, 2017.

[4] Z. Erickson, H. M. Clever, G. Turk, C. K. Liu, and C. C. Kemp, "Deep haptic model predictive control for robot-assisted dressing," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4437–4444.

[5] Y. Wang, D. Held, and Z. Erickson, "Visual haptic reasoning: Estimating contact forces by observing deformable object interactions," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 426–11 433, 2022.

[6] Y. Wang, Z. Sun, Z. Erickson, and D. Held, "One Policy to Dress Them All: Learning to Dress People with Diverse Poses and Garments," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[7] F. Zhang and Y. Demiris, "Visual-tactile learning of garment unfolding for robot-assisted dressing," *IEEE Robotics and Automation Letters*, 2023.

[8] ——, "Learning garment manipulation policies toward robot-assisted dressing," *Science robotics*, vol. 7, no. 65, p. eabm6010, 2022.

[9] S. Kotsovolis and Y. Demiris, "Bi-manual manipulation of multi-component garments towards robot-assisted dressing," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9865–9871.

[10] A. Colomé, A. Planells, and C. Torras, "A friction-model-based framework for reinforcement learning of robotic tasks in non-rigid environments," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 5649–5654.

[11] T. Tamei, T. Matsubara, A. Rai, and T. Shibata, "Reinforcement learning of clothing assistance with a dual-arm robot," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2011, pp. 733–738.

[12] I. Garcia-Camacho, M. Lippi, M. C. Welle, H. Yin, R. Antonova, A. Varava, J. Borras, C. Torras, A. Marino, G. Alenya *et al.*, "Benchmarking bimanual cloth manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1111–1118, 2020.

[13] R. P. Joshi, N. Koganti, and T. Shibata, "A framework for robotic clothing assistance by imitation learning," *Advanced Robotics*, vol. 33, no. 22, pp. 1156–1174, 2019.

[14] J. Jiang, A. Wang, and A. Aizawa, "Attention-based relational graph convolutional network for target-oriented opinion words extraction," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021, pp. 1986–1997.

[15] D. Junhua, H. Yucheng, Z. Yi-an, and Z. Dong, "Attention-based relational graph convolutional network for knowledge graph reasoning," in *2022 21st International Symposium on Communications and Information Technologies (ISCIT)*. IEEE, 2022, pp. 216–221.

[16] F. Zhang, A. Cully, and Y. Demiris, "Personalized robot-assisted dressing using user modeling in latent spaces," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3603–3610.

[17] J. Zhu, M. Gienger, G. Franzese, and J. Kober, "Do you need a hand?–an interactive robotic dressing assistance scheme," *arXiv preprint arXiv:2301.02749*, 2023.

[18] N. Koganti, T. Tamei, T. Matsubara, and T. Shibata, "Estimation of human cloth topological relationship using depth sensor for robotic clothing assistance," in *Proceedings of Conference on Advances In Robotics*, 2013, pp. 1–6.

[19] ——, "Real-time estimation of human-cloth topological relationship using depth sensor for robotic clothing assistance," in *The 23rd IEEE international symposium on robot and human interactive communication*. IEEE, 2014, pp. 124–129.

[20] F. Strazzeri and C. Torras, "Topological representation of cloth state for robot manipulation," *Autonomous Robots*, vol. 45, no. 5, pp. 737–754, 2021.

[21] F. Zhang and Y. Demiris, "Learning grasping points for garment manipulation in robot-assisted dressing," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9114–9120.

[22] J. Qie, Y. Gao, R. Feng, X. Wang, J. Yang, E. Dasgupta, H. J. Chang, and Y. Chang, "Cross-domain representation learning for clothes unfolding in robot-assisted dressing," in *European Conference on Computer Vision*. Springer, 2022, pp. 658–671.

[23] F. Zhang, A. Cully, and Y. Demiris, "Probabilistic real-time user posture tracking for personalized robot-assisted dressing," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 873–888, 2019.

[24] G. Chance, A. Jevtić, P. Caleb-Solly, G. Alenya, C. Torras, and S. Dogramadzi, ""elbows out"—predictive tracking of partially occluded pose for robot-assisted dressing," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3598–3605, 2018.

[25] Z. Erickson, M. Collier, A. Kapusta, and C. C. Kemp, "Tracking human pose during robot-assisted dressing using single-axis capacitive proximity sensing," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2245–2252, 2018.

[26] Y. Gao, H. J. Chang, and Y. Demiris, "User modelling for personalised dressing assistance by humanoid robots," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 1840–1845.

[27] ——, "Iterative path optimisation for personalised dressing assistance using vision and force information," in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2016, pp. 4398–4403.

[28] S. D. Klee, B. Q. Ferreira, R. Silva, J. P. Costeira, F. S. Melo, and M. Veloso, "Personalized assistance for dressing users," in *International Conference on Social Robotics*. Springer, 2015, pp. 359–369.

[29] E. Pignat and S. Calinon, "Learning adaptive dressing assistance from human demonstration," *Robotics and Autonomous Systems*, vol. 93, pp. 61–75, 2017.

[30] A. Kapusta, Z. Erickson, H. M. Clever, W. Yu, C. K. Liu, G. Turk, and C. C. Kemp, "Personalized collaborative plans for robot-assisted dressing via optimization and simulation," *Autonomous Robots*, vol. 43, pp. 2183–2207, 2019.

[31] G. Canal, G. Alenyà, and C. Torras, "Adapting robot task planning to user preferences: an assistive shoe dressing example," *Autonomous Robots*, vol. 43, no. 6, pp. 1343–1356, 2019.

[32] A. Clegg, Z. Erickson, P. Grady, G. Turk, C. C. Kemp, and C. K. Liu, "Learning to collaborate from simulation for robot-assisted dressing," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2746–2753, 2020.

[33] T. Matsubara, D. Shinohara, and M. Kidode, "Reinforcement learning of a motor skill for wearing a t-shirt using topology coordinates," *Advanced Robotics*, vol. 27, no. 7, pp. 513–524, 2013.

[34] S. Huo, A. Duan, C. Li, P. Zhou, W. Ma, H. Wang, and D. Navarro-Alarcon, "Keypoint-based planar bimanual shaping of deformable linear objects under environmental constraints with hierarchical action framework," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5222–5229, 2022.

[35] J. Zhu, B. Navarro, R. Passama, P. Fraisse, A. Crosnier, and A. Cherubini, "Robotic manipulation planning for shaping deformable linear objects withenvironmental contacts," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 16–23, 2019.

[36] F. Süberkrüb, R. Laezza, and Y. Karayiannidis, "Feel the tension: Manipulation of deformable linear objects in environments with fixtures using force information," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 216–11 222.

[37] P. Mitrano, A. LaGrassa, O. Kroemer, and D. Berenson, "Focused adaptation of dynamics models for deformable object manipulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5931–5937.

[38] Z. Weng, F. Paus, A. Varava, H. Yin, T. Asfour, and D. Kragic, "Graph-based task-specific prediction models for interactions between deformable and rigid objects," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 5741–5748.

[39] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. W. Battaglia, "Learning mesh-based simulation with graph networks," *arXiv preprint arXiv:2010.03409*, 2020.

[40] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia, "Learning to simulate complex physics with graph

networks," in *International conference on machine learning*. PMLR, 2020, pp. 8459–8468.

[41] A. Longhini, M. Moletta, A. Reichlin, M. C. Welle, D. Held, Z. Erickson, and D. Kragic, "Edo-net: Learning elastic properties of deformable objects from graph dynamics," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3875–3881.

[42] X. Ma, D. Hsu, and W. S. Lee, "Learning latent graph dynamics for visual manipulation of deformable objects," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 8266–8273.

[43] Y. Huang, C. Xia, X. Wang, and B. Liang, "Learning graph dynamics with external contact for deformable linear objects shape control," *IEEE Robotics and Automation Letters*, 2023.

[44] Y. Kim and C. Sloth, "Assembly strategy for deformable ring-shaped objects," in *IROS*, 2020, pp. 357–358.

[45] I. G. Ramirez-Alpizar, K. Harada, and E. Yoshida, "A simple assembly planner for the insertion of ring-shaped deformable objects," *Assembly Automation*, vol. 38, no. 2, pp. 182–194, 2018.

[46] S. Jin, D. Romeres, A. Ragunathan, D. K. Jha, and M. Tomizuka, "Trajectory optimization for manipulation of deformable objects: Assembly of belt drive units," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 10 002–10 008.

[47] R. Laezza, R. Gieselmann, F. T. Pokorny, and Y. Karayiannidis, "Reform: A robot learning sandbox for deformable linear object manipulation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4717–4723.

[48] C. Wang, Y. Zhang, X. Zhang, Z. Wu, X. Zhu, S. Jin, T. Tang, and M. Tomizuka, "Offline-online learning of deformation model for cable manipulation with graph neural networks," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5544–5551, 2022.

[49] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*. Springer, 2018, pp. 593–607.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[51] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning," 2016.

[52] E. Valassakis, Z. Ding, and E. Johns, "Crossing the gap: A deep dive into zero-shot sim-to-real transfer for dynamics. in 2020 ieee," in *RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5372–5379.

[53] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.